Atlas-based Segmentation
Medical Image Registration and Applications

Md. Kamrul Hasan, Basel Alyafi, Fakrul Islam Tushar

December 9, 2018

Contents

1 Introduction

Image segmentation is the process of dividing the image into non-overlapping regions. It is considered as a hard job for specialists who usually spend long times specifying pixels to which region they belong. Automatic segmentation makes that job easier and has a wide range of techniques and a long list of applications. One of the most promising techniques that has been widely used in image segmentation is Atlas-based image segmentation. In this report, atlas is going to be used in the medical domain using brain image of MICCAI challenge to segment the brain into the following non-overlapping regions:

1. Cerebrospinal Fluid (label 1).
2. White matter (label 2).
3. Gray matter (label 3).

2 Atlas Construction

To construct an Atlas, The main steps are as follows:

1. Register the dataset to one representative image (selecting this image is discussed later in 2.1).
2. Apply the transformation got from the previous registration on dataset labels.
3. Average the registered images to get the template image.
4. Extract different regions pixels from registered labels and average them.

To see the bird-eye view, Fig. 1 and Fig. 2 show the complete process of constructing the template image and labels.

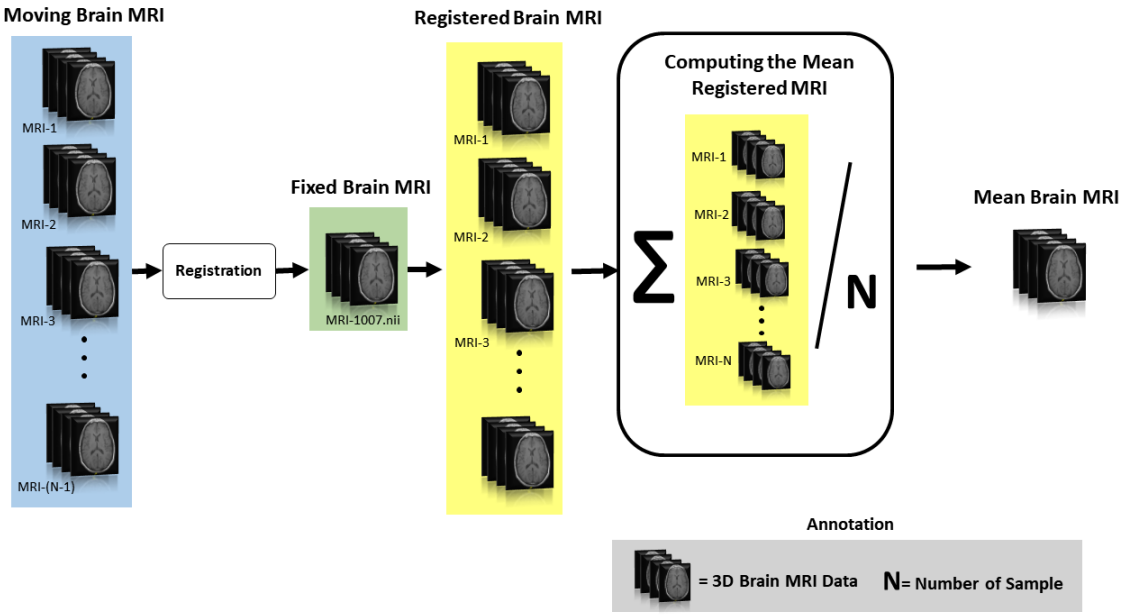


Figure 1: The pipeline of constructing Atlas template image

2.1 Database Registration

To register the database images, a good representative should be selected wisely. The way that was done in this report is as follows:

1. Use one image as fixed and all others as moving
2. Register all moving images (Affine Transform) with the fixed one and measure individual Normalized Cross Correlation.
3. Record metric values
4. Change the fixed image to the next one and repeat until all the images are considered as fixed once.
5. Pick the one with the most-stable, has short legs!, and accepted-median metric values, see Fig. 3

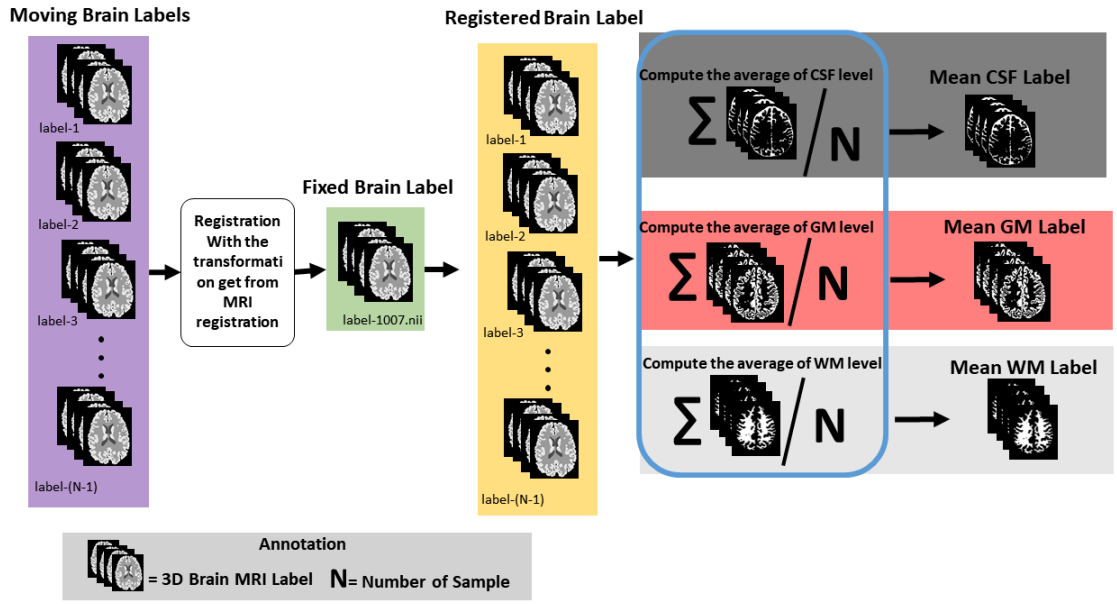


Figure 2: The pipeline of constructing Atlas template labels

After applying the previous algorithm, image 1007 was selected because it showed relatively robust results. Analysis the box-plot we can see NCC is varying a lot for dinned MRI as fixed image. Fig. 3 shown MRI data "1007.nii" and "1008.nii" giving the maximum median for the NCC as fixed image compared to others and there is no outliers in both cases.Now if we compared the outcomes for "1007.nii" and "1008.nii" it can be seen that "1007.nii" has small range for The upper and lower whiskers compared to the "1008.nii" which represents less score outside the middle 50%. considering all this factor "1007.nii" is selected as fixed image for the registration process.

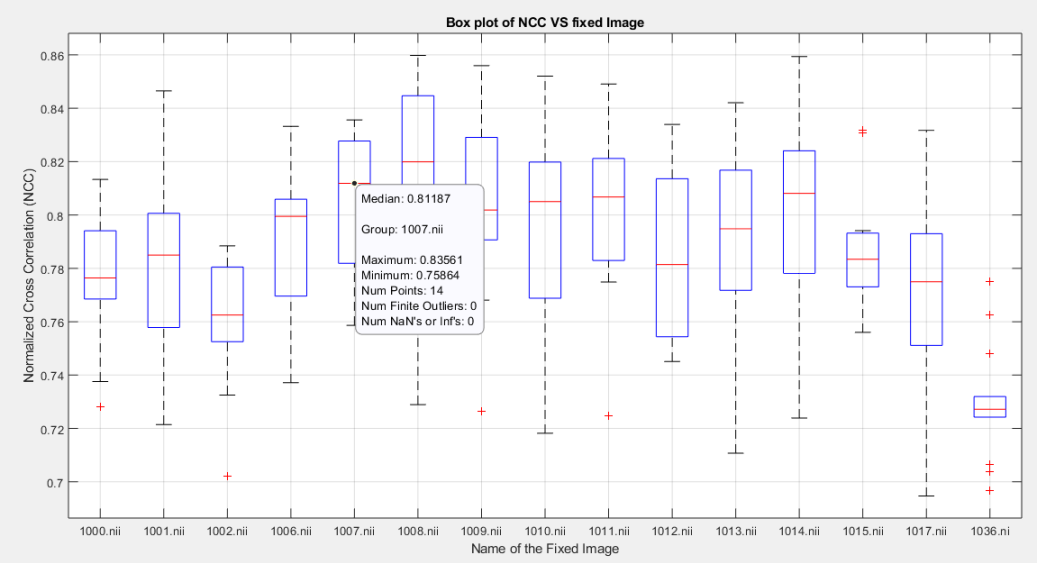


Figure 3: The boxplot showing the statistics for different fixed images and corresponding normalized cross correlation

2.2 Elastix Parametric file Parameters Selection

In this section we will discuss about the registration methods and parameter selection. The desired registration was performed in two steps,first an Affine (rigid registration) registration was applied for initial alignment, afterwards, a non-rigid registration namely B-SPLINE REGISTRATION was applied as a final registration. Below, first, we discuss parameters selection for the initial registration and we elaborate to it to final registration.

2.3 Initial Registration

The main characteristic of this parameter file is the `AffineTransform`. The idea of doing so is to match the large picture (scale, translation, rotation, etc..) without going into details. A picture is worth a thousand words. Fig. 4 shows the 141^{th} axial slice from the following:

- Fixed is 1007 always.
- Moving is 1000.
- Result is the registration result.

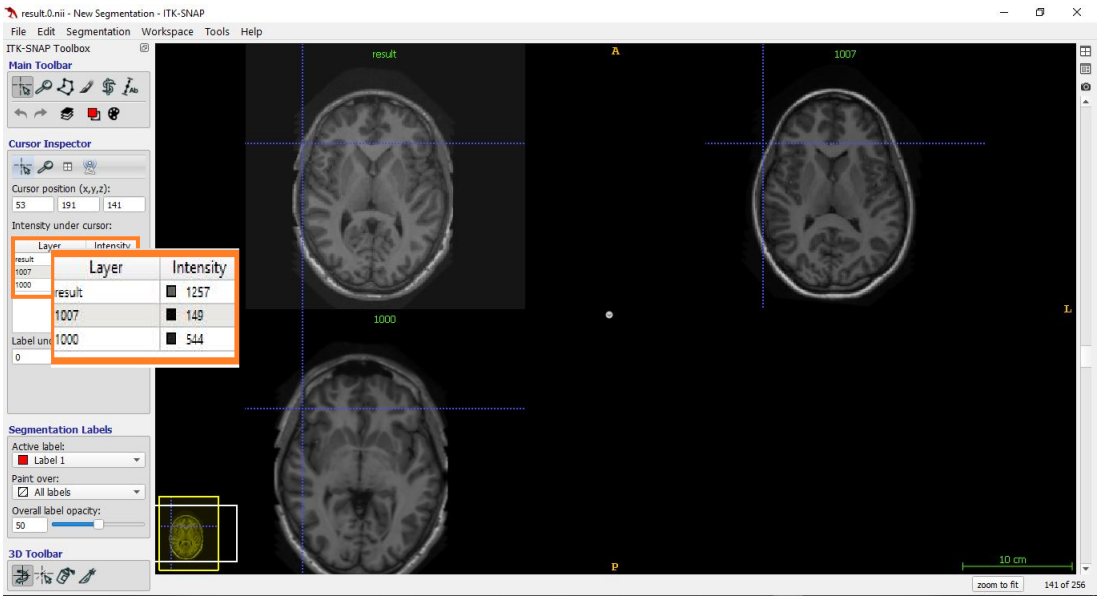


Figure 4: Affine Registration, the 141^{th} axial slice

From the figure, it can be read that, (53, 191) point is on the skull of the registered image while it is background for the fixed image due to unfeasible (using affine transformation) deformations.

2.3.1 Multi-resolution Framework

Our experience from the previous lab works proved that multi-resolution framework increases the chance of successful registration. So, here we have used multi-resolution frame. We used Gaussian pyramid for our registration. In Elastix parametric file its defines as,

```
(FixedImagePyramid "FixedRecursiveImagePyramid")
(MovingImagePyramid "MovingRecursiveImagePyramid")
```

In Elastix, Gaussian pyramid applies smoothing and down-sampling. Next, we defined the number of resolution or level of pyramid we want to use. In general 3 resolution is good starting point but as recommended in [1] for 3D data is better to use up-to 5. we use 5 resolution. The default scheduler was used smoothing and down-sampling which smooth the fixed image by a factor of 2 in each dimension.

2.3.2 Image Sampler

In general, during registration looping over all the voxels of the of the fixed image is not necessary a subset is enough for the registration. In Elastix Image Sampler defines this sampling strategies with different options as random, on a grid, etc. We used A random coordinate sampler as it states in [1] to be performed well in conjunction with the AdaptiveStochasticGradientDescent optimizers which is been used a optimizer for the registration.

```
(ImageSampler "RandomCoordinate")
```

Then we defined the amount of samples, randomly selected in every iteration and enforce the selection of new samples in every iteration by the 2 lines below respectively.

```
(NewSamplesEveryIteration "true")
```

2.3.3 Interpolation

During Optimization for evaluating the non-voxels position the interpolation is used. Elastix provides a range of interpolation option such as Nearest neighbor, Linear and n-th order B-spline. In our registration Linear interpolation is used as it fast and a good trade-off between quality and speed.

```
(Interpolator "LinearInterpolator")
```

For generating the final result is Elastix we need to define ResampleInterpolator. Which is shown below,

```
(ResampleInterpolator "FinalBSplineInterpolator")
        (Resampler "DefaultResampler")
```

2.3.4 Transformation

We have performed a rigid registration using affine transformation. Affine transformation allows the translation, rotation,scaling and sharing. The parameter vector for the affine transformation is a vector of 12 parameters for 3D volume.Elastix using following line in parameter file we defined affine transformation,

```
(Transform "AffineTransform")
```

Affine transform require a centre of rotation, by default the geometric centre of the fixed image is taken, which is recommended [1]. In affine a important parameter is scale that we need to set. It is recommended in [1] to let the elastix compute it automatically. the line below was used to perform this.

```
(AutomaticScalesEstimation "true")
```

2.3.5 Metric

To optimize the error and perform better registration elastix provides numbers of similarity measures such as, Mean Square Differences (MSD), Normalized Correlation Coefficient (NCC), Mutual Information (MI) etc. In our registration we used NCC as metric which is defined by this equation.

$$NCC = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2(y_i - \overline{y})^2}} \tag{1}$$

where x_i=fixed image, $\overline{x} = mean\ \ of\ \ fixed\ \ image$, y_i=Moving Image, $\overline{y}$ =mean of moving image.

In Elastix it's defined as below,

```
(Metric "AdvancedNormalizedCorrelation")
```

NCC assumes a linear relation between the intensity values of the fixed and moving image [1].

2.3.6 Optimizer

In our registration as a optimizer we have used AdaptiveStochasticGradientDescent which is advanced version of StandardGradientDescent but avoids the gain factor initialization problem of StandardGradientDescent. AdaptiveStochasticGradientDescent it estimate proper initial value automatically. It is defined in the Elastix parameter file as below,

```
(Optimizer "AdaptiveStochasticGradientDescent")
```

Reasonable values for the parameters was estimated using displacement distribution using following line.

```
(ASGDParameterEstimationMethod "DisplacementDistribution")
```

2.4 Final Registration (B-spline)

For the final registration we applied Bspline non-rigid registration. For the final registration all the parameters discussed in the initial parameter Section 2.3 except for final registration the Transformation used it's `BSplineTransform`

In this case, the major difference is that the transformation here is `BSplineTransformation`. Fig. 5 shows a comparison with Fig.4.

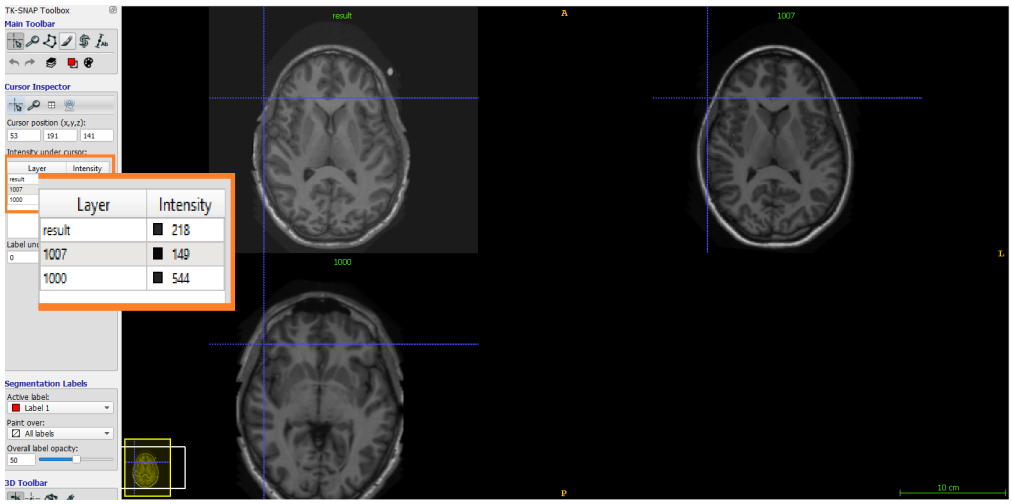


Figure 5: Using BSpline Transformation, the axial slice 141 becomes more similar to the fixed one

From Fig.5, it is obvious that point (53,191) in both the registered and fixed image is on the skull with closer intensities that using the affine transformation.

2.4.1 Transformation:BSplineTransform

The B-spline nonrigid transformation is defined by a uniform grid of control points. This grid is defined by the spacing between the grid nodes[1].Following line dine the transformation

```
(ResampleInterpolator "FinalBSplineInterpolator")
```

according to [1] most of the literature used cubic B-spline which ic is the 3rd order B-spline. following line was used to define the 3d order B-spline.

(FinalBSplineInterpolationOrder 3)

Initial and Final registration elastix's parameter files were shown below.

Elastix Parameter File

Initial Registration (Affine)
(Registration "MultiResolutionRegistration")
(Metric "AdvancedNormalizedCorrelation")
(ImageSampler "RandomCoordinate")
(Interpolator "LinearInterpolator")
(ResampleInterpolator "FinalBSplineInterpolator")
(Resampler "DefaultResampler")
(Transform "AffineTransform")
(Optimizer "AdaptiveStochasticGradientDescent")
(FixedImagePyramid "FixedSmoothingImagePyramid")
(MovingImagePyramid "MovingSmoothingImagePyramid")
(NewSamplesEveryIteration "true")
(NumberOfResolutions 5)
(FinalBSplineInterpolationOrder 3)
(AutomaticScalesEstimation "true")
(ASGDParameterEstimationMethod "DisplacementDistribution")
(FixedInternalImagePixelType "float")
(MovingInternalImagePixelType "float")
(HowToCombineTransforms "Compose")
(ResultImageFormat "nii")

2.5 Atlas Template and Labels

After applying the algorithm described in the previous two sections, the representative template and transformed (three) labels are available to play with. Fig. 7, Fig. 8, and Fig. 9 show a few slices of the template image. Figures 10, 11, 12, 13, 14, 15, 16 show different slices from average labels.

Final Registration (B-spline)
(Registration "MultiResolutionRegistration")
(Metric "AdvancedNormalizedCorrelation")
(ImageSampler "RandomCoordinate")
(Interpolator "LinearInterpolator")
(ResampleInterpolator "FinalBSplineInterpolator")
(Resampler "DefaultResampler")
(Transform "BSplineTransform")
(Optimizer "AdaptiveStochasticGradientDescent")
(FixedImagePyramid "FixedSmoothingImagePyramid")
(MovingImagePyramid "MovingSmoothingImagePyramid")
(NewSamplesEveryIteration "true")
(NumberOfResolutions 5)
(FinalBSplineInterpolationOrder 3)
(AutomaticScalesEstimation "true")
(ASGDParameterEstimationMethod "DisplacementDistribution")
(FixedInternalImagePixelType "float")
(MovingInternalImagePixelType "float")
(HowToCombineTransforms "Compose")
(ResultImageFormat "nii")

3 Results and Discussion

To evaluate the results, two ways were used: quantitative (Normalized Cross Correlation) and qualitative (by looking at results on ITK-snap). First, let's look at the NCC values for registering the dataset to image 1007, in other words, 1007 is fixed image and all others are moving. Table 1 shows the values with the average and standard deviation at the end. Those results were taken from `elastix` log files for the final registration (`BSplaineTransformation`) and last iteration and resolution. By comparing values for the boxplot shown in Fig.3 for 1007 as fixed and the median shown in Table 1, the benefit of using `BSplineTransformation` becomes clear.

3.1 Registered Tissue Models

To have a look at the distributions of pixels after registration, alg. 1 was followed.

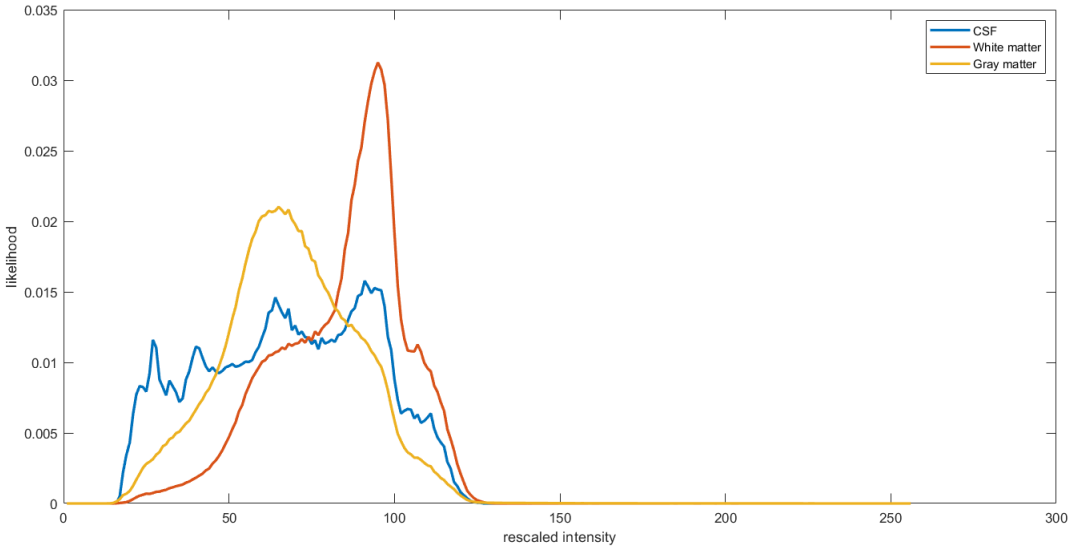
Algorithm 1 Tissue model calculation

1: rescale the image to have a predefined range [0,N]
2: extract regions pixels $(R_{CSF}, R_{WM}, R_{GM})$
3: $H_1 = Normalized\_Histogram(R_{CSF}, N+1)$; //N+1 bins
4: $H_2 = Normalized\_Histogram(R_{WM}, N+1)$
5: $H_3 = Normalized\_Histogram(R_{GM}, N+1)$
6: $H_j(i) = H_j(i) / \sum_{j=1}^{3} H_j(i)$, **for all** $0 \leq i \leq N$ **do**

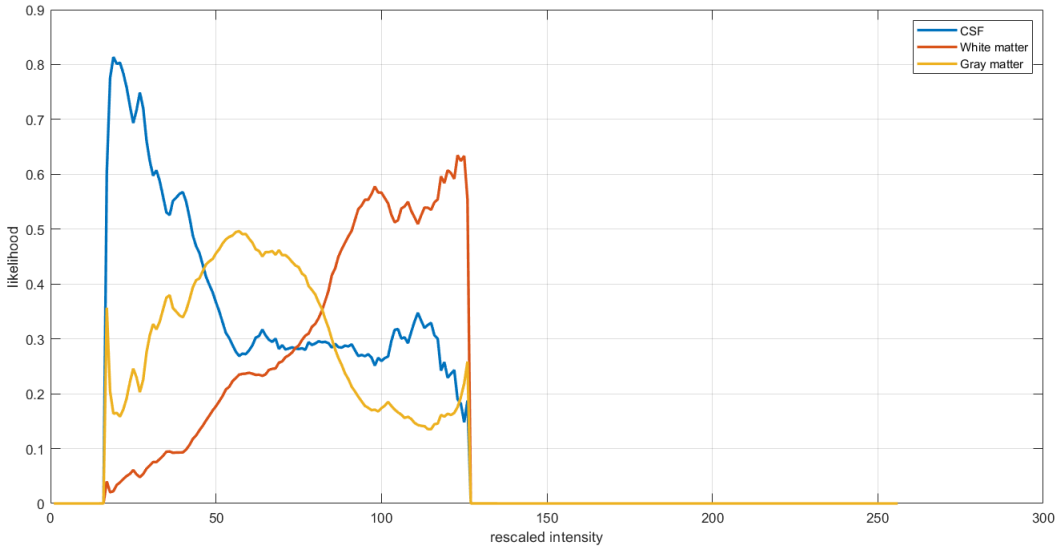
Moving Image name	**-NCC value**
1000	-0.957
1001	-0.962
1002	-0.917
1006	-0.960
1008	-0.971
1009	-0.966
1010	-0.968
1011	-0.967
1012	-0.965
1013	-0.953
1014	-0.962
1015	-0.959
1017	-0.965
1036	-0.949
Median	**-0.962**
Average	**-0.958**
Std	**0.0129**

Table 1: Metric values for registering the dataset to 1007

The idea of making the tissue model is to have probabilities that can be used for segmenting the tissues. At any vertical line, inside available intensities, the some of probabilities is one, see Fig.6.



(a) normalized histogram by tissue



(b) tissue model probabilities

Figure 6: Comparison between the normalized histogram and tissues models

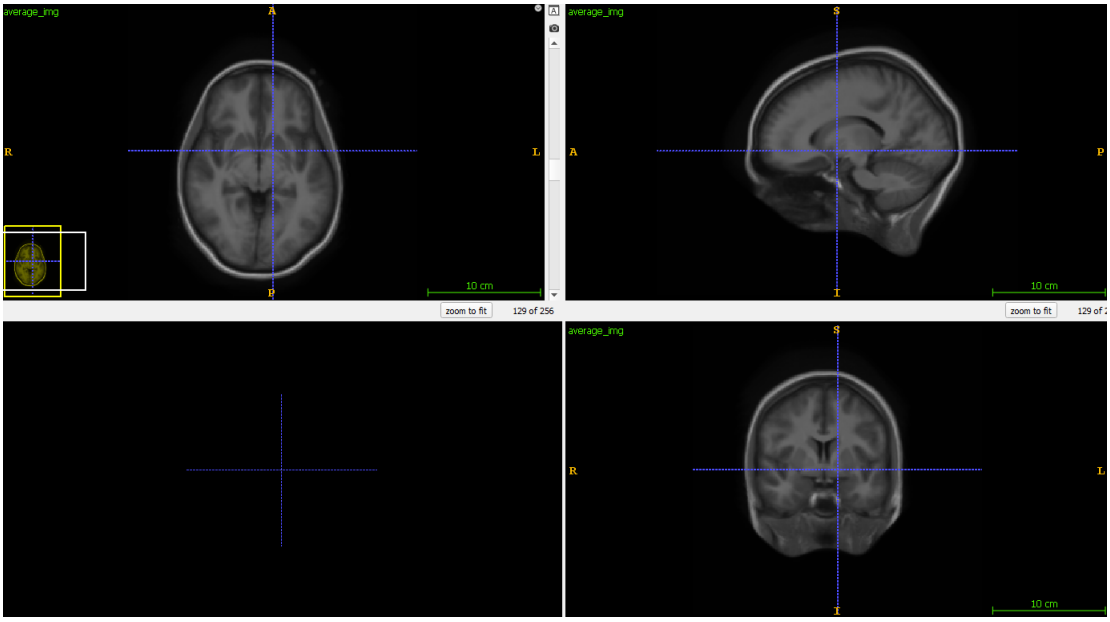


Figure 7: Axial slice = 129, sagittal, and coronal slices in the template image

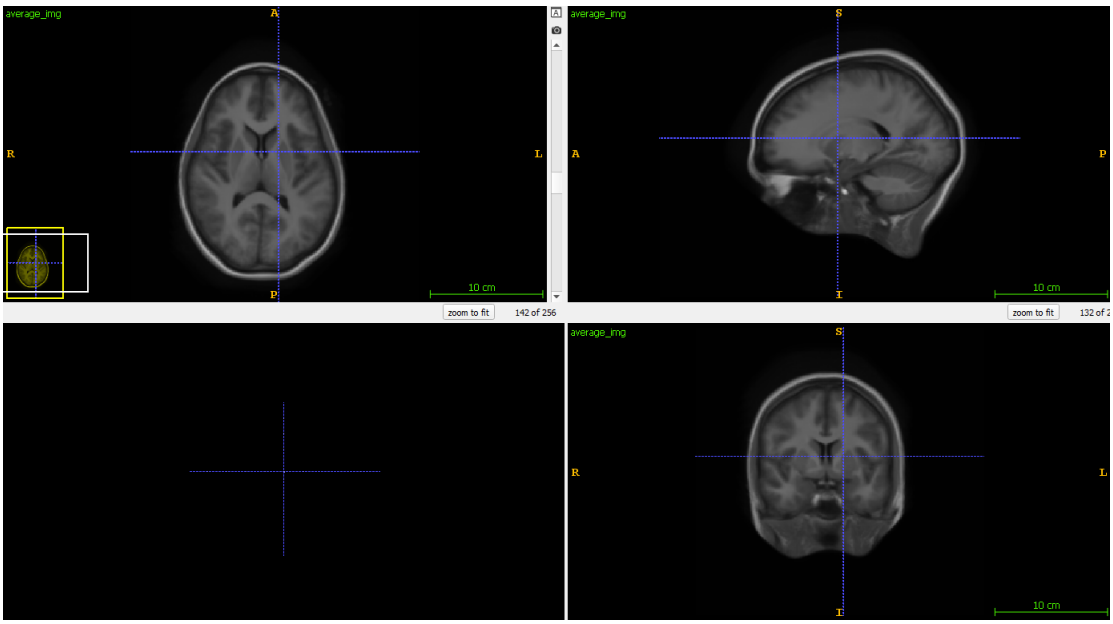


Figure 8: Axial slice = 142, sagittal, and coronal slices in the template image

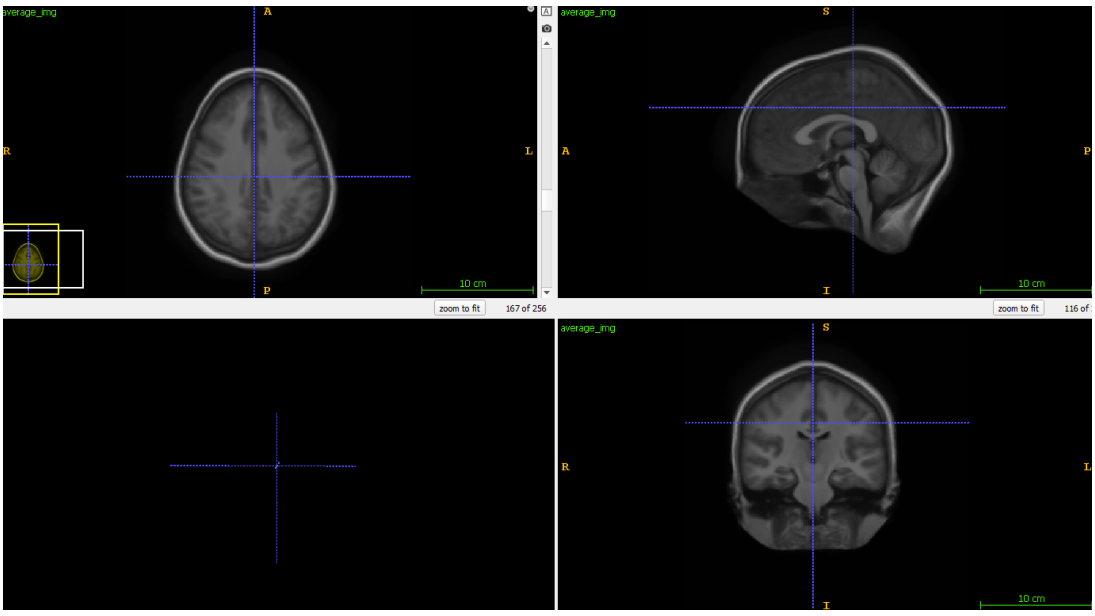


Figure 9: Axial slice = 167, sagittal, and coronal slices in the template image

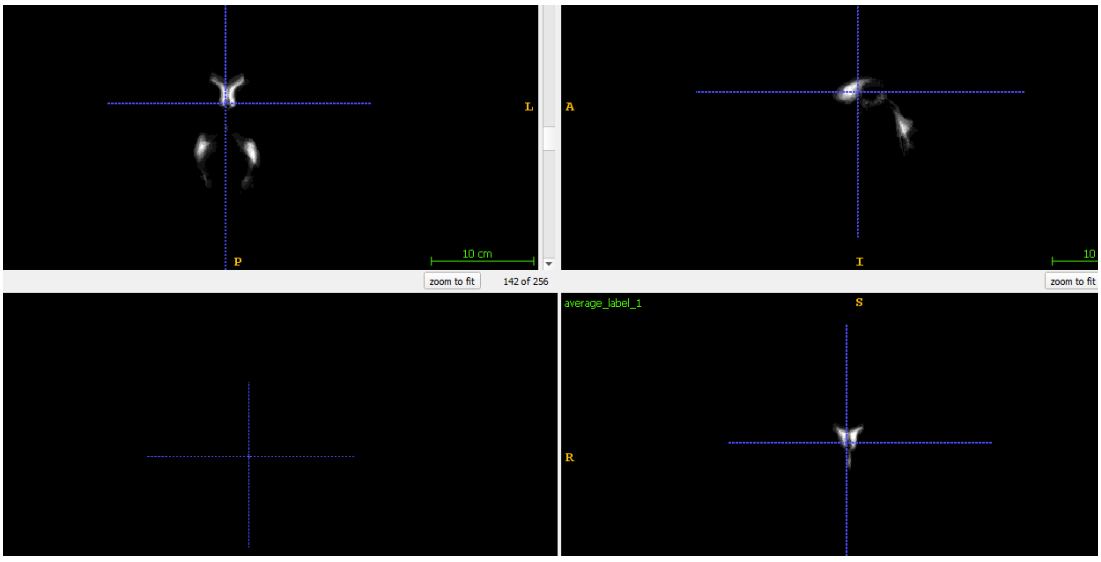


Figure 10: Axial=115, sagittal, and slices from the template label of CSF region

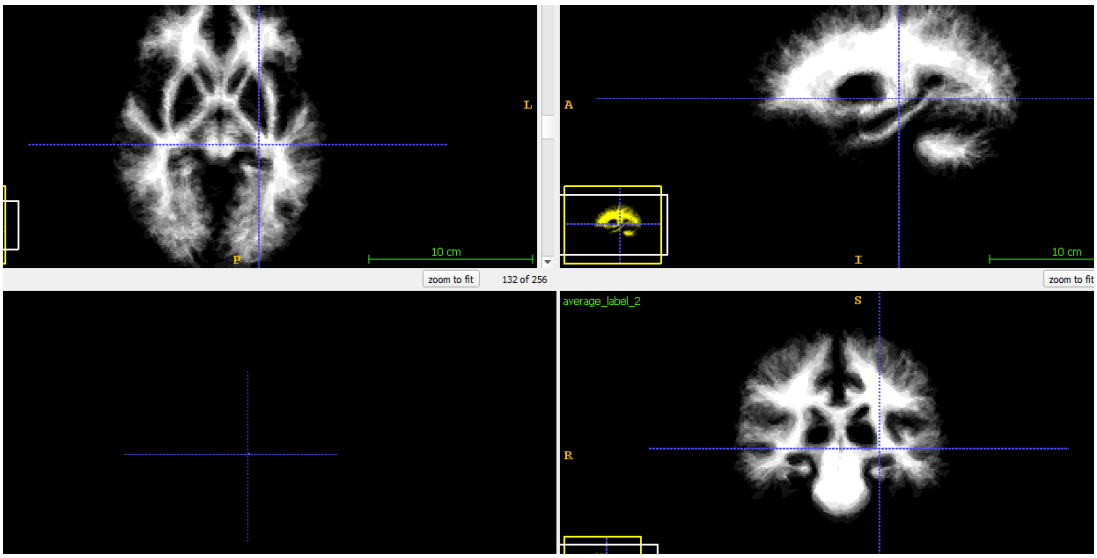


Figure 11: Axial slice = 132, from the template label of White matter (WM) region

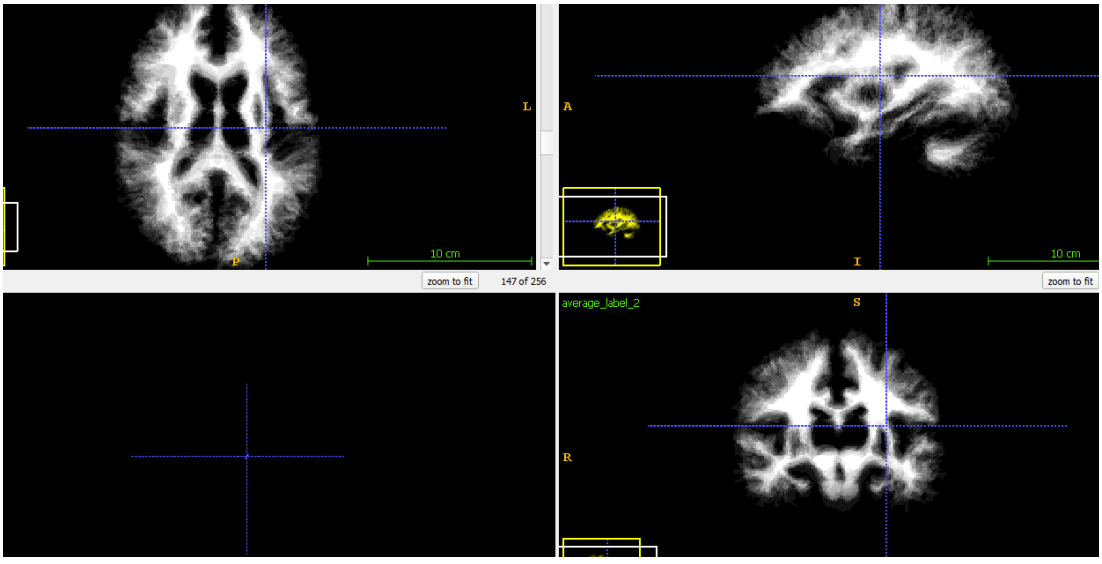


Figure 12: Axial = 147, sagittal, and coronal slices from the template label of White matter (WM) region

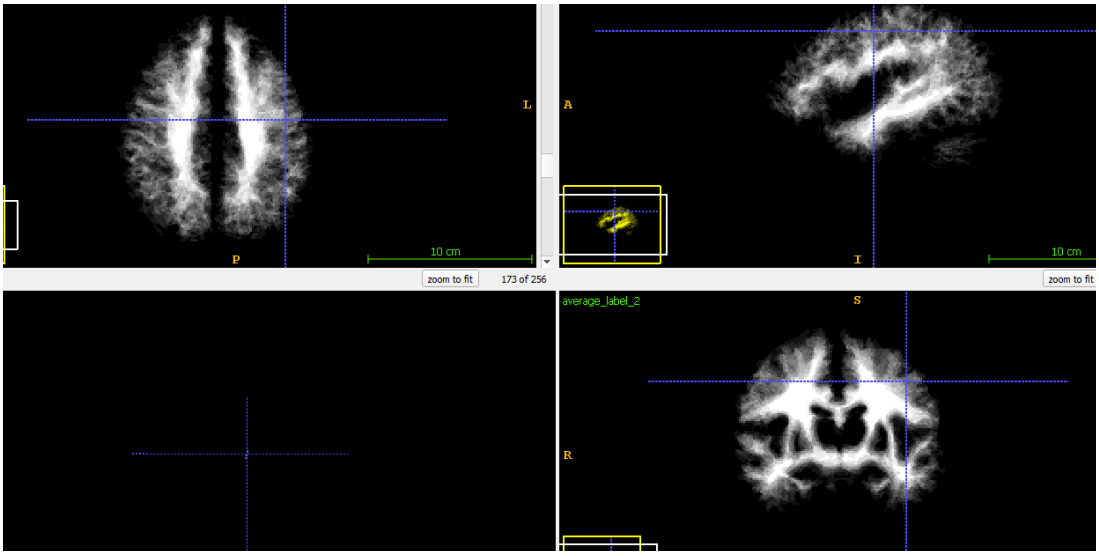


Figure 13: Axial = 173, sagittal, and coronal slices from the template label of White matter (WM) region

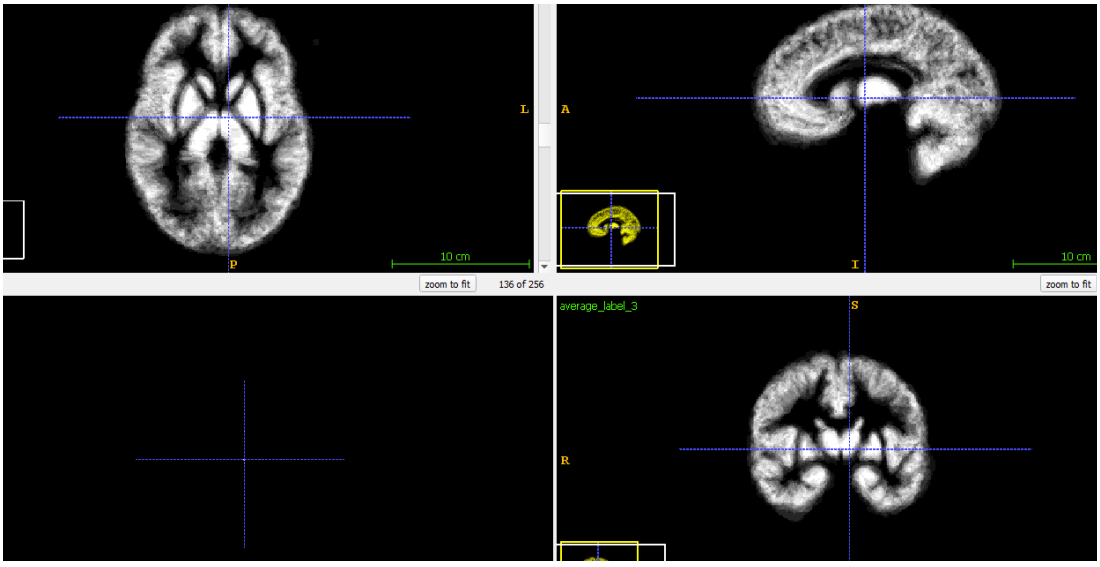


Figure 14: Axial slice = 136 of average label 3 (Gray matter)

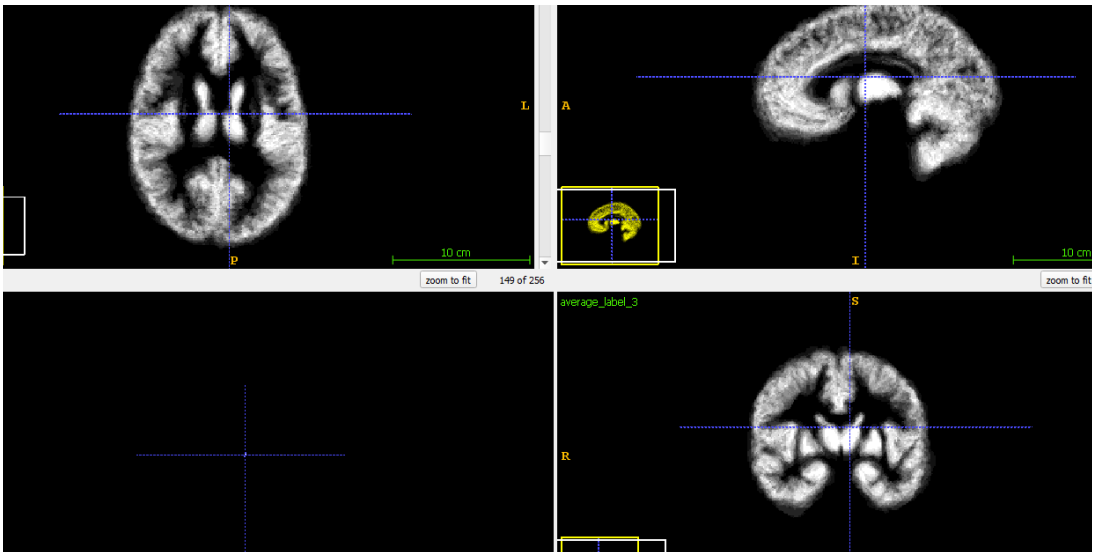


Figure 15: Axial slice = 149 of average label 3 (Gray matter)

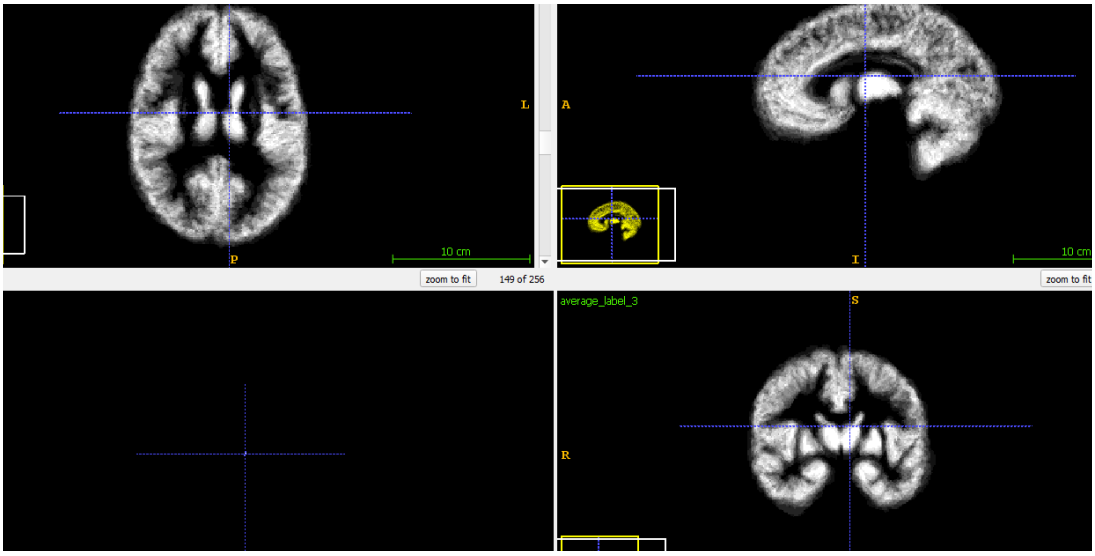


Figure 16: Axial slice = 171 of average label 3 (Gray matter)

References

[1] K. Stefen, and S. Marius, "elastix the manual," September 4, 2015.

Em + Atlas Brain Tissue Segmentation
Medical Image Segmentation and Applications

Md. Kamrul Hasan, Basel Alyafi, Fakrul Islam Tushar

December 16, 2018

Contents

1 Introduction and Problem Definition

Brain tissue segmentation is one of the primary tools in analyzing brain tissues. In previous labs, Expectation Maximization (EM) and Atlas Construction strategies ,which are widely used approaches for brain tissue segmentation, were implemented from scratch. This coursework is a continuation of our Registration Lab Where we built probabilistic atlas for brain tissue segmentation. We are going to combine EM with Atlas for segmenting CSF (Cerebrospinal fluid) with label 1, WM (White Matter) with label 2, and GM (Gray Matter) with label 3. In this report, we propose a solution of 5 complexity levels, namely:

1. Applying atlas based segmentation and extracting the probabilistic atlas.
2. Finding the distribution of the training images for each region and using it to assign labels.
3. Implementing the Bayesian framework by joining the upper two results.
4. Applying EM and extracting probabilities for each voxel belonging to CSF, WM or GM, starting from Bayesian labels.
5. The joint framework, Bayesian EM: combining EM and Bayesian framework.

At the end of this report, analysis of the outcome of the joint framework compared to single frameworks is provided.

2 Algorithm Design

All the phases of the algorithm are discussed here with the equations needed. The order of the sections is similar to the one used in the algorithm.

2.1 Atlas-based Segmentation

To construct an atlas, the main steps are as follows:

1. Register the dataset to one representative image (selecting this image is discussed broadly in part-1 of this Coursework (MIRA part)).
2. Apply the transformation got from the previous registration on dataset labels.
3. Average the registered images to get the template image.
4. Extract different regions' pixels from registered labels and average them.

To see the bird-eye view, Fig.1 and Fig. 2 show the complete process of constructing the template image and labels.

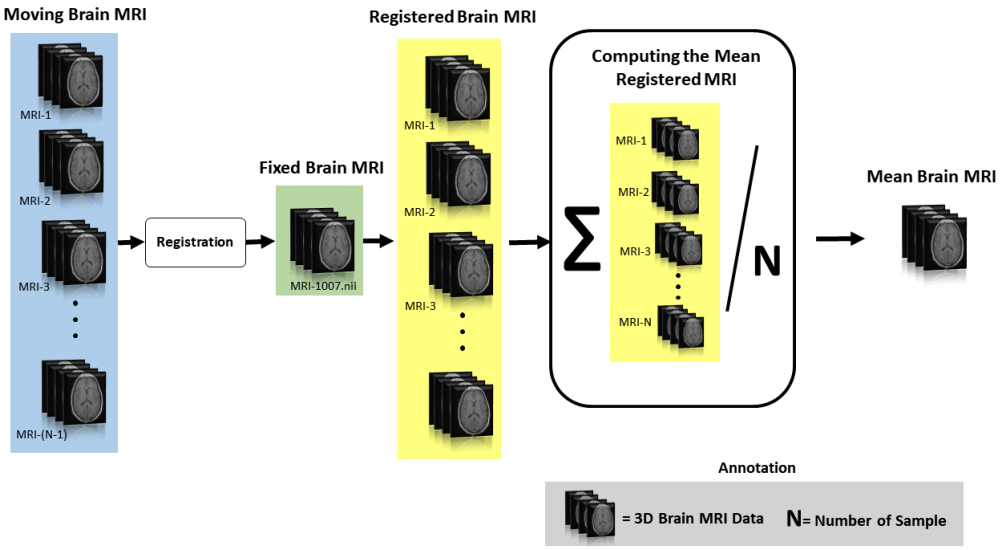


Figure 1: The pipeline of constructing Atlas template image

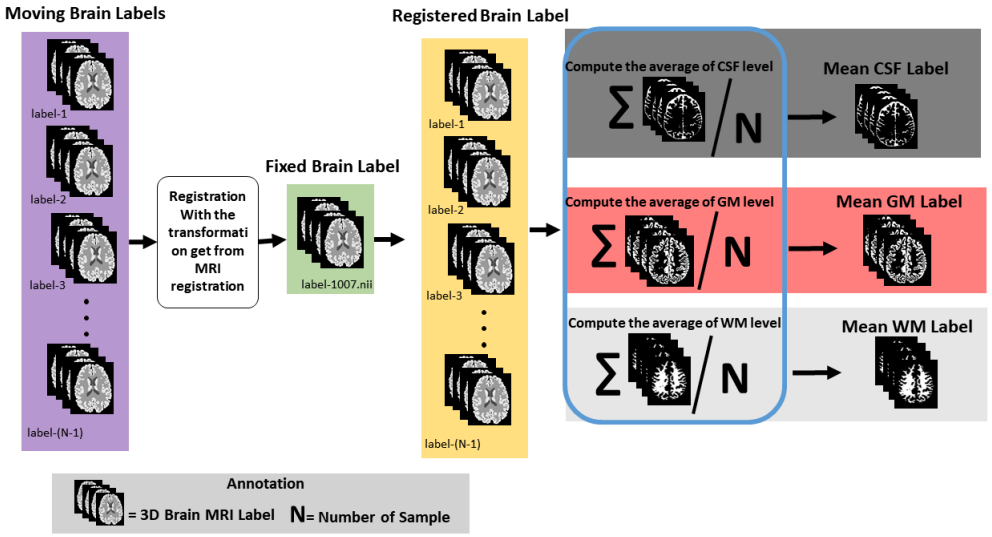


Figure 2: The pipeline of constructing Atlas template labels.

After constructing the atlas template image and template labels, the ultimate goal is to assign label to the test image. To do that, the template has to be registered (as moving) with every test image (as fixed). The resulting transformation is applied thereafter on template labels (for three regions separately) and the index of the maximum is taken lastly. Algorithm 1 shows the routine briefly.

Algorithm 1 Atlas-based Segmentation

1: $T = Register(fixed = template\_image, moving = test\_image)$
2: Using T, Transform template labels $(CSF\_prob, WM\_prob, GM\_prob)$ to get $CSF\_prob^T, WM\_prob^T, GM\_prob^T$
3: predicted_labels $= argmax(CSF\_prob^T, WM\_prob^T, GM\_prob^T)$

2.2 Tissue Models

Tissue models means that each region (CSF, WM, GM) has a distribution which can be depicted by analyzing a large number of images of the same machine. Unfortunately, only 15 images are there to analyze, though, they are good enough. Every training image is loaded along with its labels. Regions are extracted and the histogram is calculated (not normalized yet). The previous process is repeated for all training images and histograms are accumulated accordingly (one histogram accumulator for each region). Histograms are normalized at the end by dividing by the total corresponding-region pixels number. Algorithm 2 shows the process. Where, 4096 bins were used for the histograms of the three regions, that is because images had maximum intensities around this value and compressing the distributions might have caused more overlapping among regions.

Algorithm 2 Region-wise distribution

1: initialize all variables with zeros $(C_1^1 C_2^1 \dots C_{4096}^1, C_1^2 C_2^2 \dots C_{4096}^2, C_1^3 C_2^3 \dots C_{4096}^3)$
2: **for** all training images **do**
3: rescale the image to have integer values (0 to 4095)
4: extract regions pixels
5: accumulate N^1, N^2, N^3 with corresponding number of pixels.
6: compute the histogram region wise (H^1, H^2, H^3).
7: accumulate with old ones $C^1 += H^1, C^2 += H^2, C^3 += H^3$.
8: **end for**
9: normalize histograms

$$\frac{C^1}{N^1}, \frac{C^2}{N^2}, \frac{C^3}{N^3}$$

10: apply moving-average window of width 45 centered.
11: for each intensity, scale the three histograms by the sum of all histograms at that intensity.

$$C_j^i = \frac{C_j^i}{\sum_{i=1}^{3} C_j^i}, \forall i = 1, 2, 3; j = 1, 2, \dots 4096$$

Where, the normalizing step was mainly for using the distribution as a proper probability distribution (the sum at any intensity should be one). While, the filtering step was to fill the holes in the distribution (due to some intensities not available in images) and reduce perturbations, see Fig.4. After running the algorithm specified in Algorithm 2, on all 15 training images, the result was as shown in Fig.3

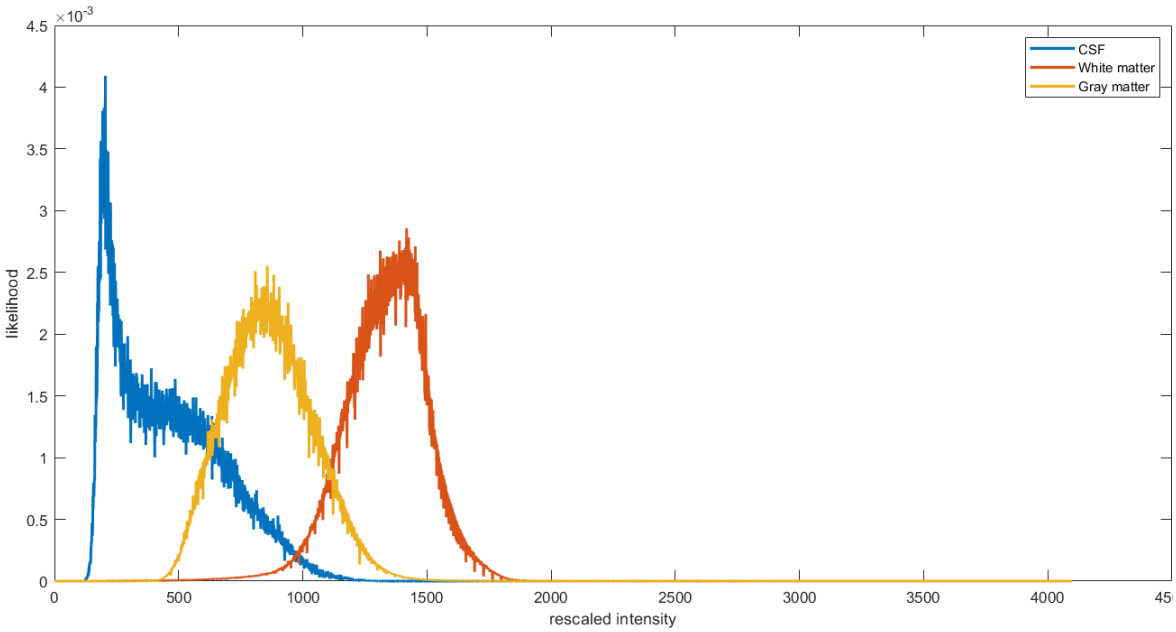


Figure 3: Tissue model before scaling and filtering

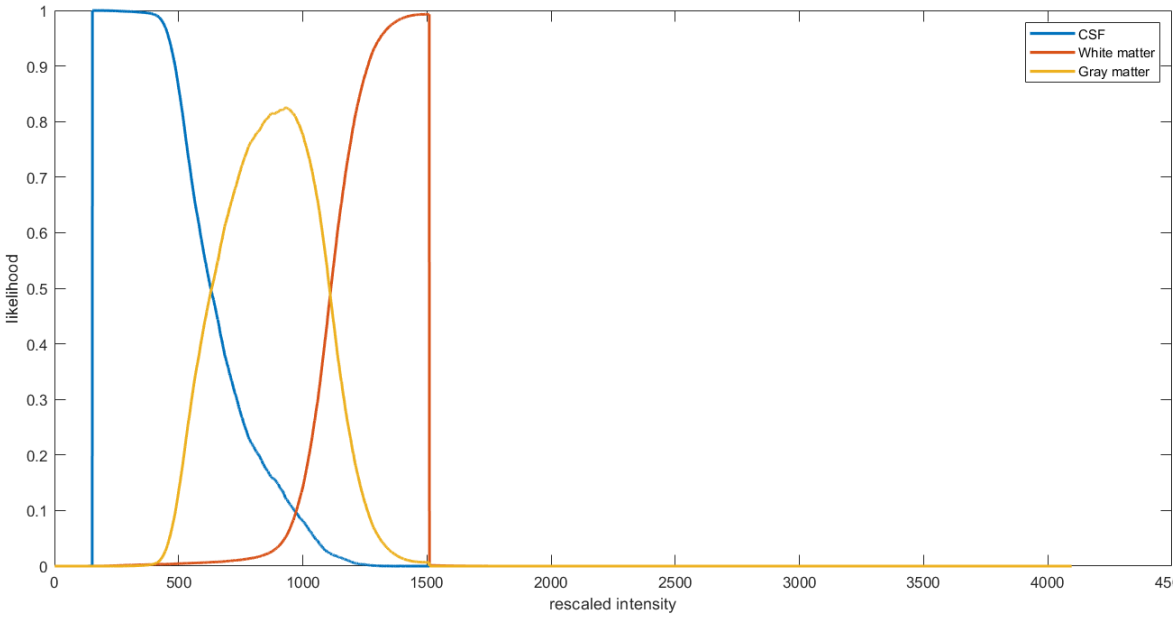


Figure 4: Tissue model after scaling and filtering

So, the probability for one pixel with coordinates (x, y) with intensity $I(x)$ (after rescaling to integer value between 0 and 4095) to belong to CSF, for example, is:

$$p^1(I(x,y)) = C^1_{I(x,y)}$$

It might be needed to add one to the intensity value when dealing with systems of 1 indexing like Matlab, and similarly for other regions. To find the label of that pixel using tissue models

$$label(x,y) = \underset{i}{argmax}(C^i_{I(x,y)})$$

2.3 Bayesian Framework

In Bayesian Framework, tissue models probabilities were multiplied by probabilistic atlas ones. Alg. 3 shows that procedure.

Algorithm 3 Bayesian Framework

Bayesian probabilities for pixel with coordinates (x,y) and intensity I(x,y):

$$bayes\_CSF(x,y) = C^1_{I(x,y)} * CSF\_prob^T(x,y)$$

$$bayes\_WM(x,y) = C^2_{I(x,y)} * WM\_prob^T(x,y)$$

$$bayes\_GM(x,y) = C^3_{I(x,y)} * GM\_prob^T(x,y)$$

To find the predicted label using the Bayesian framework:

$$label(x,y) = argmax(bayes\_CSF(x,y), bayes\_WM(x,y), bayes\_GM(x,y))$$

2.4 Expectation Maximization (EM)

Expectation Maximization (EM) is one of the most popular approaches to maximize the likelihood. The basic idea of the EM algorithm is to, estimate the distribution of the variable given data and current value of the parameters which called E-step and Maximizing the joint distribution of the data and the hidden variable. Starting from initialization, expectation, maximization, and stopping criteria. To make all the symbols used clear, a brief definition is given in Table 1. Three Gaussian Mixture Models are assumed. Each pixel is processed and the probability of belonging to each mixture model is calculated. The maximum probability of every observation is found and the corresponding index is considered the predicted label what is called soft assignment of pixel label.

Symbol	**meaning**
N	number of observations (relative to the context)
K	number of distinct classes
μ_k	the k^{th} class mean
Σ_k	the k^{th} class covariance matrix
α_k	the k^{th} mixture weight
$w_i k$	the probability that x^i belongs to class k
$N - k$	the effective number of observation belonging to class k

Table 1: Symbols definition

2.4.1 Initialization

Usually, EM is initialized with either Kmeans or randomly. However, in this coursework, a valuable information is available and gives a very good starting point for EM. This information is the registered Atlas (after step 3 in Algorithm 1). Clusters are distinguished by having different labels. Another important point is that the randomness of Kmeans (which comes from the random initialization of kmeans) is avoided by a robust initialization. To be familiar with notations used here, have a look at Table 1.

2.4.2 Expectation

In this step, the weights for every pixel being belonged to each class are calculated. Those weights are computed using the Gaussian mixture model multiplied (eq(3)) by alpha (which is in the first iteration, the proportion of the corresponding class :number of class elements divided by the total number of elements). To use the mixture model, the mean and covariance matrix for each class are computed using equations (1, 2), respectively. The new weights are computed via eq(4).

$$\mu_k^{new} = \frac{1}{N_k} \sum_{i=1}^{N} w_{ik} \ . \ x^i \tag{1}$$

$$\sigma_k^{new} = \frac{1}{Nk} \sum_{i=1}^{N} w_{ik}.(x^i - \mu_k^{new})(x^i - \mu_k^{new})'; 1 \leq k \leq K \tag{2}$$

$$p(x|\theta_k) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \ e^{\ -\frac{1}{2}(x-\mu_k)' \ \Sigma_k^{-1} \ (x-\mu_k)}; \ \theta_k := \mu_k, \Sigma_k \tag{3}$$

$$w_{ik} = \frac{p(x_i|\theta_k).\alpha_k}{\sum_{m=1}^{K} p(x_i|\theta_m).\alpha_m} \tag{4}$$

2.4.3 Maximization

In this step, the new mixture weights are computed via the following set of equations. Where, N_k is the effective number of observations belonging to the k^{th} class eq(5), and alpha is updated using eq (6).

$$N_k = \frac{1}{N} \sum_{i=1}^{N} w_{i,k} \tag{5}$$

$$\alpha_k = \frac{N_k}{N} \tag{6}$$

2.4.4 Stopping Criteria

To recognize convergence, a small change (if any) of the order of 10^{-3} in the log likelihood is used, while, a maximum number of iterations is used to assure the stability. The log likelihood is defined in eq (7).

$$\log l(\Theta) = \sum_{i=1}^{N} \log \sum_{k=1}^{K} \alpha_k.p(x_i|\theta_k) \tag{7}$$

2.5 Bayesian EM

To implement EM in conjunction with the bayesian fraemwork, Fig. 5 demonstrates a pictorial presentation.

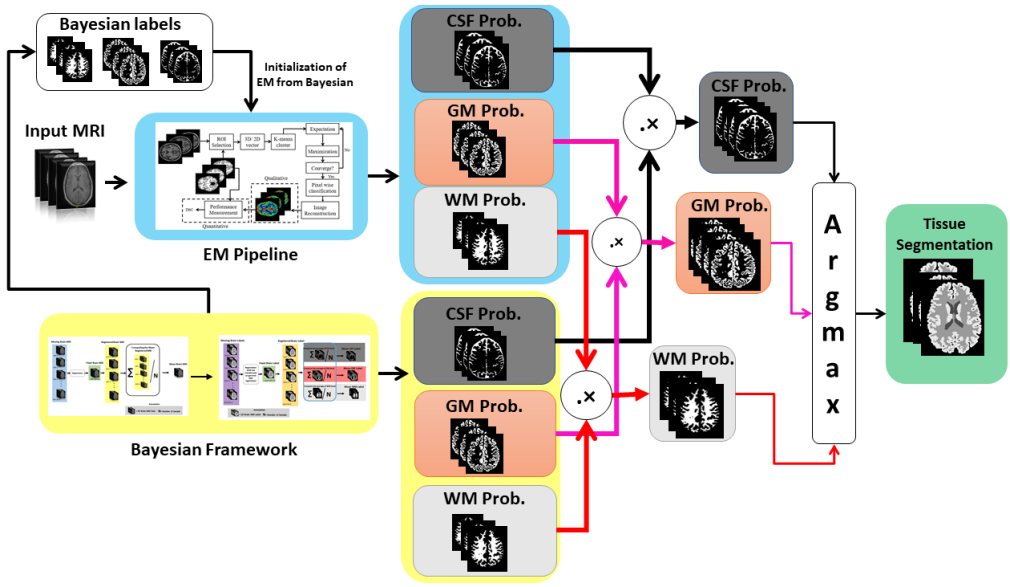


Figure 5: The pipeline for EM+Atlas based brain tissue segmentation.

In the ***EM_Probability.m*** function, for getting probability for each pixels either belong to CSF or GM or WM, test image has been passed along with the Bayesian labels (to cluster pixels). The way of implementing Expectation Maximization algorithm was described in a previous lab report (MISA lab report-2). This function will return three probabilistic volumes for CSF, WM, and GM as shown in Fig. 5. The pipeline for creating probabilistic atlas for CSF, GM, and WM has been described in Session-1 of this course work (MIRA report) with qualitative and quantitative performances.

Now, we have got three 3D probabilistic volumes (CSF Prob., GM Prob, WM Prob.) from EM pipeline and three 3D probabilistic volumes(CSF Prob., GM Prob, WM Prob.) from the bayesian framework pipeline as shown in Fig. 5 which were multiplied element-wise to get a single volumes for each of CSF, GM, and WM. Afterward, "**Argmax**" operator has been applied on the 4D volume for each pixel to assign the tissue label either CSF or GM or WM.

3 Results and Discussion

In this section, we show three main cases, namely:

- Atlas with the complete dataset.
- MNI atlas.
- Atlas with the small training dataset.

For each case, the results are discussed region-wise using box plots considering all the five complexity levels mentioned in Section 1. Additionally, image 1004 was selected with the axial slice 150 to show some qualitative results.

3.1 Atlas with the complete Training Dataset (15 images)

In this case, 15 training images were available to train the system, i.e., to learn tissues distributions and prior region locations. For probabilistic atlas, image 1007 was used as fixed and all others as moving, the registered moving labels were averaged and template

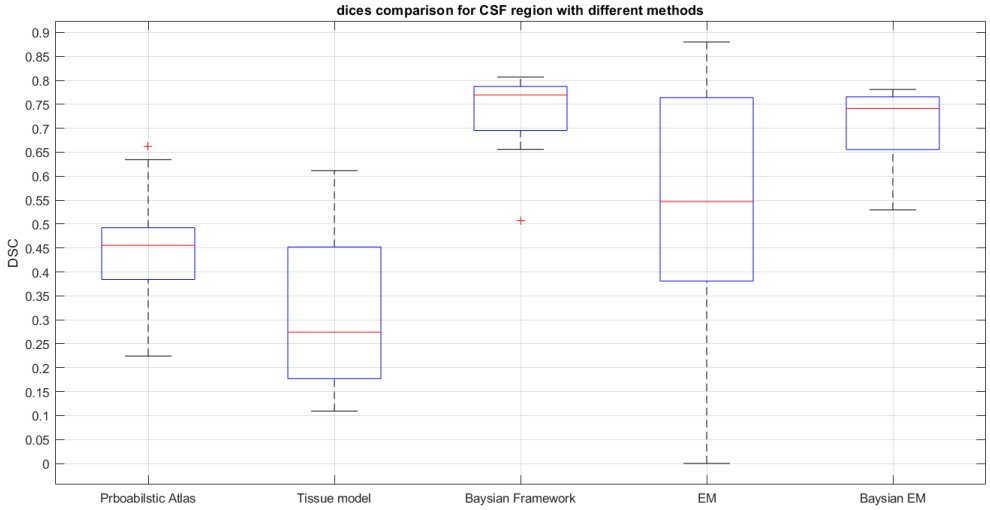
labels probabilities for three regions were used thereafter for complexity level 1 (atlas only). Template labels probabilities were registered to all test images individually, see Fig.1, Fig.2, Algorithm 1. Bayesian then was done by multiplying the previous results by the tissue models and passed as initialization to EM, the final EM probabilities were again multiplied by the bayesian probabilities and argmax was done finally. Fig.6 (a) (b) (c) show the box plots for the three regions with the five complexity levels.
To have a few words about the box plots, First, for CSF, it is clear that the Bayesian framework is outperforming other methods (median is higher than others), additionally, it is more stable due to the small box it has with short whiskers, see Fig.6(a). However, for the White matter, the fifth level of complexity (EM multiplied by Bayesian) has the highest median and maximum among others with acceptable stability (whiskers lengths), see Fig.6(b). Finally, for the Gray matter, again, the last method has outperformed other methods (highest median, highest maximum, best minimum). From the three previous figures, it is clear that the last algorithm is behaving relatively good.
Qualitatively speaking, Fig.10 shows five different segmentation results using the five methods. The continuous improvement can be seen by looking at the decreasing color area (green area comes from the groundtruth, while, purple area comes from the predicted labels) in the overlay images.

3.2 MNI Atlas Results

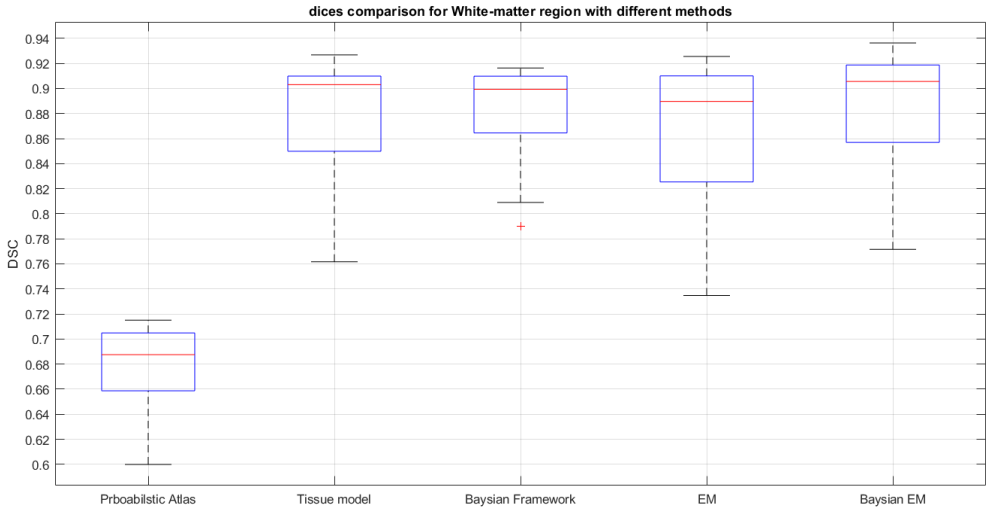
With MNI, one 4D object was given containing the mask, CSF probabilities, White matter probabilities, Gray matter probabilities. Those probabilities were extracted to be used as a probabilistic atlas (outcome of Fig. 2). A template image was given as well (the outcome of Fig.1). For tissue models, the avaialble training set (15 images) was used. To show the results of this MNI atlas, Fig.7 demonstrates the box plots. Firstly, for CSF segmentation, Bayesian framework showed good results, while, on the other hand, EM showed high maximum but unstable DSC values. Moreover, after multiplying EM with Bayesian, the DSC drops by a little amount. So, the chief here is the Bayesian framework, see Fig.7(a). Secondly, for White matter, here, EM wins with the highest median and maximum, with acceptable length of the whiskers, see Fig.7(b). Thirdly and finally, for gray matter results, Fig.7(c), EM is still the boss providing good median and stability. To sum up for MNI results, EM is giving really good and stable results for WM and GM but CSF result is not acceptable, so, it would be wiser to use Bayesian EM.
Qualitatively speaking, Fig.11 shows five different segmentation results using the five methods. The continuous improvement can be seen by looking at the decreasing color area (green area comes from the groundtruth, while, purple area comes from the predicted labels) in the overlay images.

3.3 Atlas With the Small Training Dataset

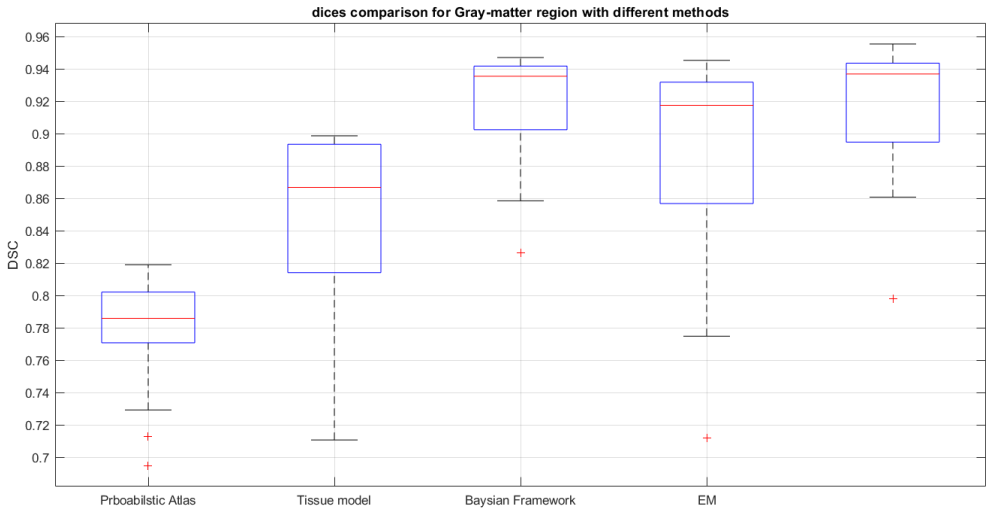
In this case, only four images were available in the dataset, 1008 was found to be the best option to use as fixed and others as moving. See Fig.8 for explaining the reason beyond that. All three images were registered to 1008.



(a) CSF boxplot with the complete training set

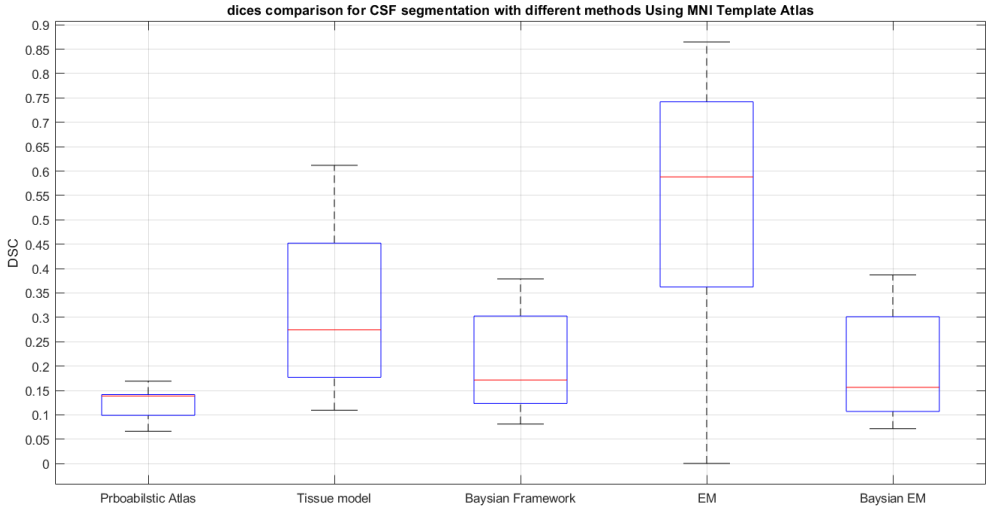


(b) WM boxplot with the complete training set

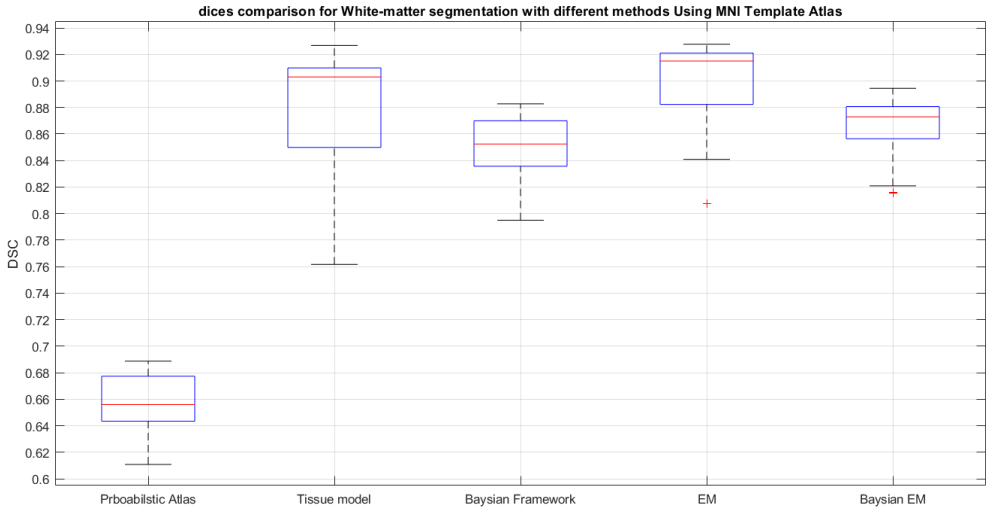


(c) GM boxplot with the complete training set

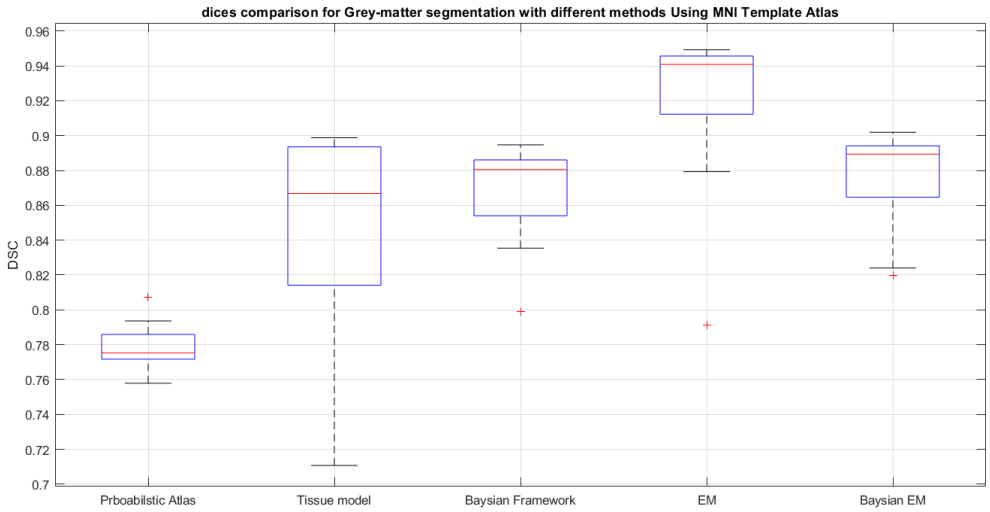
Figure 6: DSC comparison between different methods and regions using Atlas with the complete dataset



(a) CSF boxplot using MNI atlas



(b) WM boxplot using MNI atlas



(c) GM boxplot using MNI atlas

Figure 7: DSC comparison between different methods and regions using MNI atlas

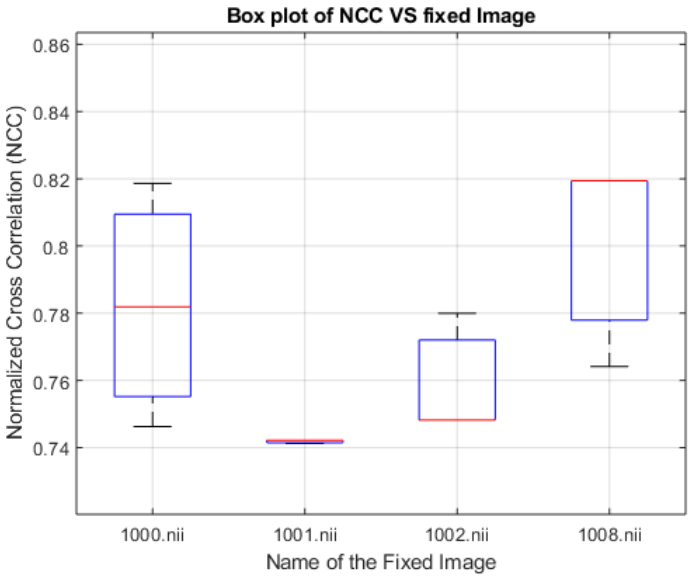
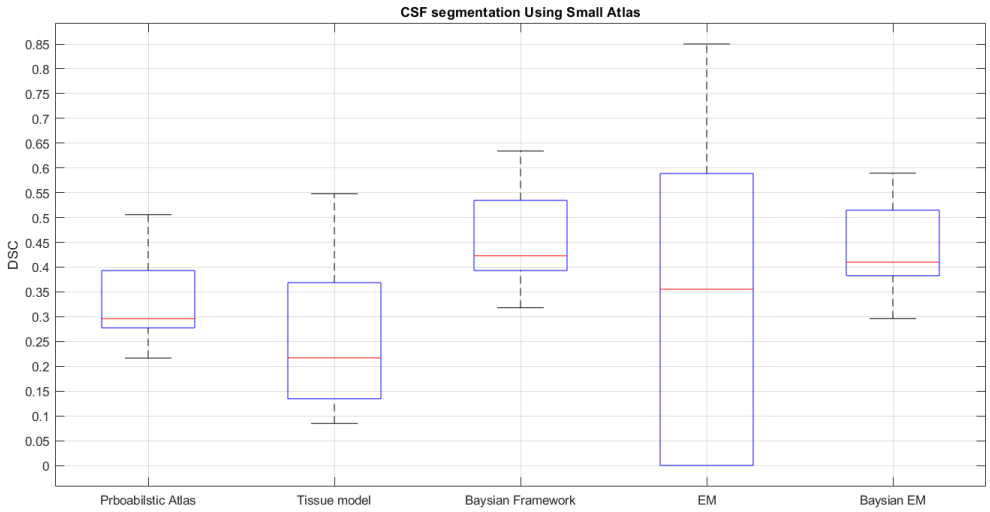


Figure 8: NCC comparison of different registrations

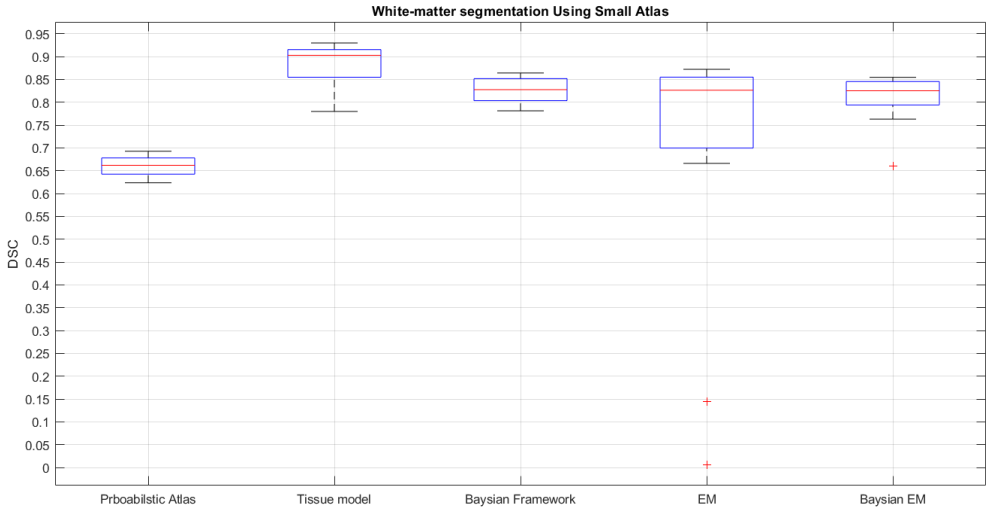
After that, tissues models were computed using Algorithm 2 for the four images only. To show results of this case, Fig.9 does the job. As can be seen from Fig.9(a) for CSF, Bayesian and Bayesian EM are giving close results with similar distributions. While, for White matter, Fig.9(b) surprisingly, tissue models only were able to give challenging results. For Gray matter, Fig.9(c) Bayesian and Bayesian EM are both highly acceptable. Overall, Bayesian EM is again and for the third time is selected as best choice.
Qualitatively speaking, Fig.12 shows five different segmentation results using the five methods. The continuous improvement can be seen by looking at the decreasing color area (green area comes from the groundtruth, while, purple area comes from the predicted labels) in the overlay images.

3.4 Discussion

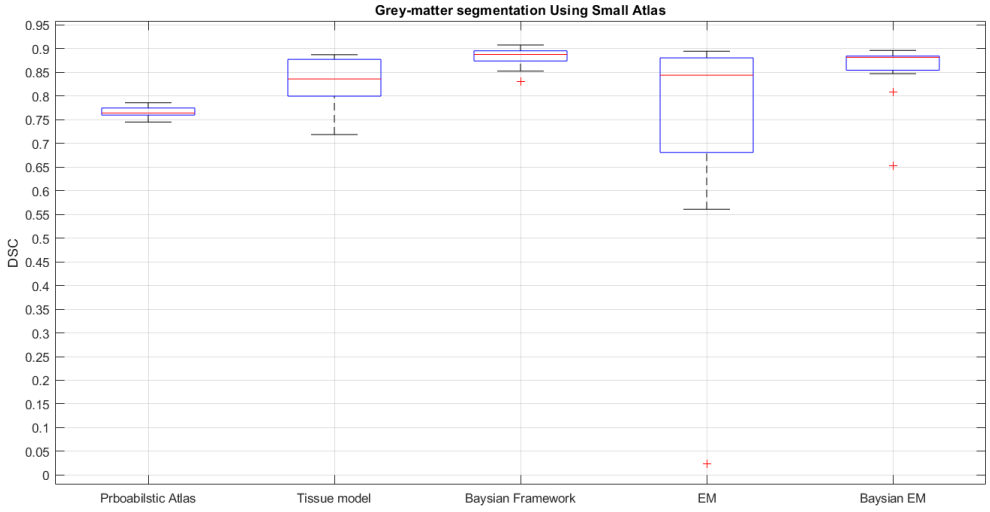
Quantitatively and qualitatively, Bayesian EM proved more accurate than other methods using either the complete dataset, the small dataset, or the MNI template. Comparing Fig.10(e), Fig.11(e), and Fig.12(e), it is clear that the complete dataset did better than the small one due to the fact that the larger the dataset, the more accurate the tissue models will be, which will enhance the whole chain. MNI was performing a bit less than the complete atlas owing to coming probably from a different modality brand (different distributions). So far, five different methods were exhaustively tested on different scales of the problem, Bayesian EM showed exceptional robustness among others and is proposed as a solution for the problem of brain tissue segmentation.



(a) CSF boxplot using small atlas

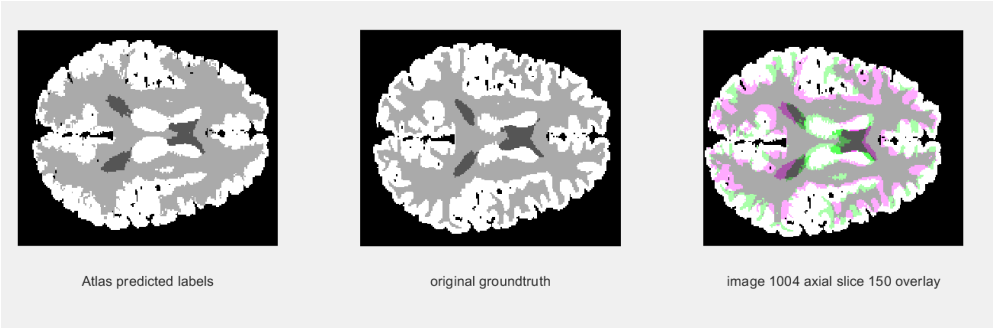


(b) WM boxplot using small atlas

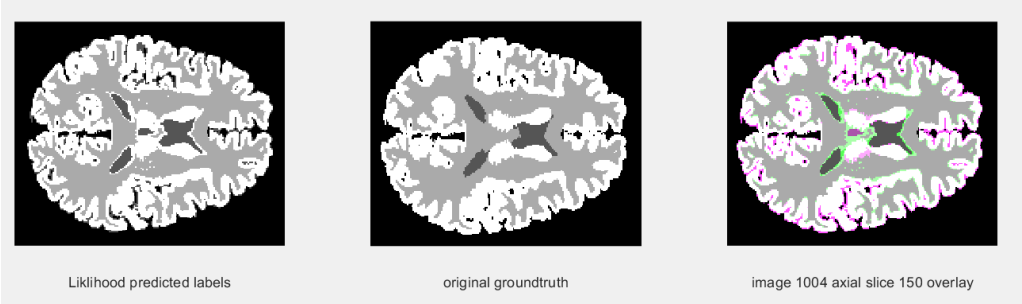


(c) GM boxplot using small atlas

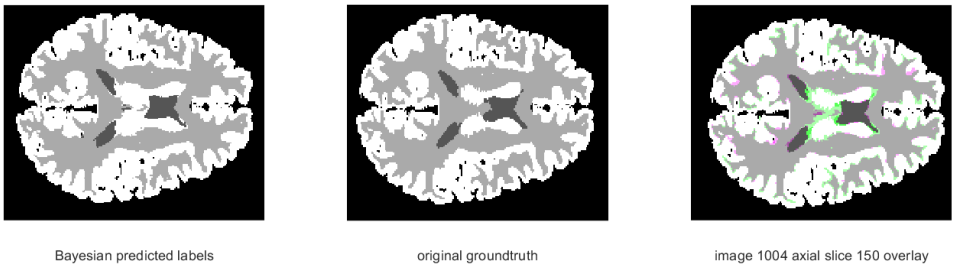
Figure 9: DSC comparison between different methods and regions using small atlas



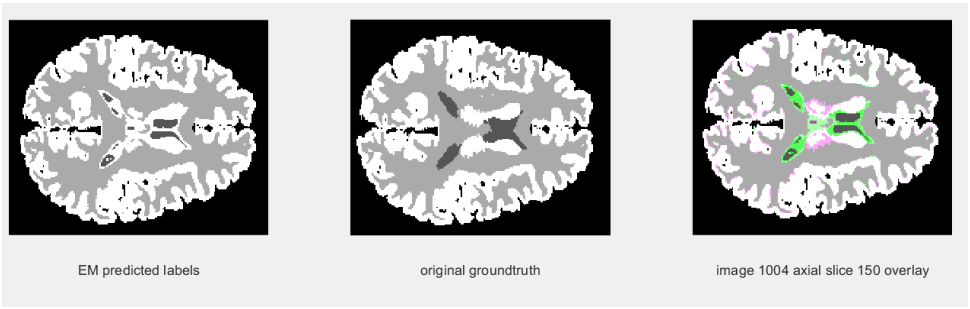
(a) atlas segmentation, dices (CSF, WM, GM) = 44, 71, 80



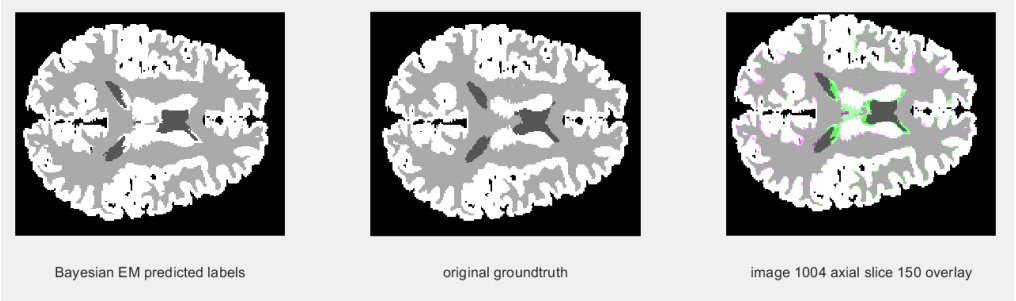
(b) tissue model segmentation, dices = 18, 90, 86



(c) Bayesian framework segmentation, dices= 67, 90, 94

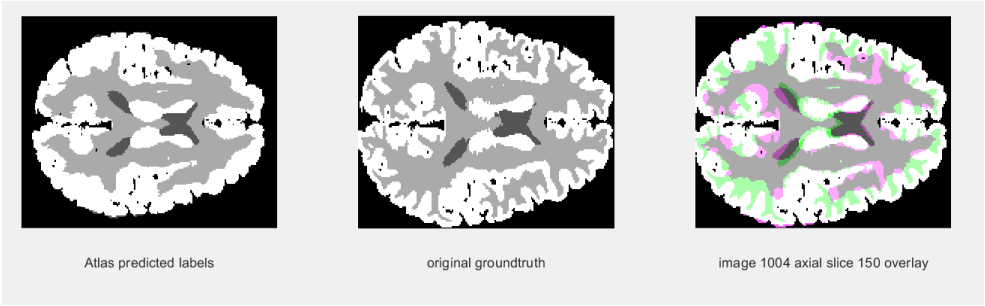


(d) EM segmentation, dices= 54, 92, 94

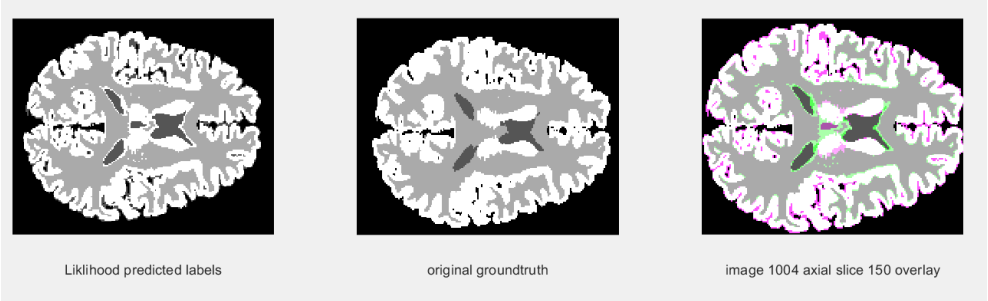


(e) Bayesian EM segmentation, dices= 64, 93, 95

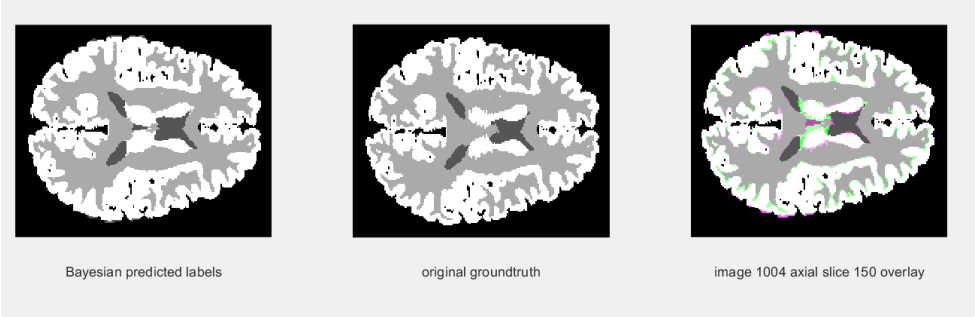
Figure 10: Image 1004, axial slice 150, qualitative evaluation of the five methods using Atlas with the complete dataset



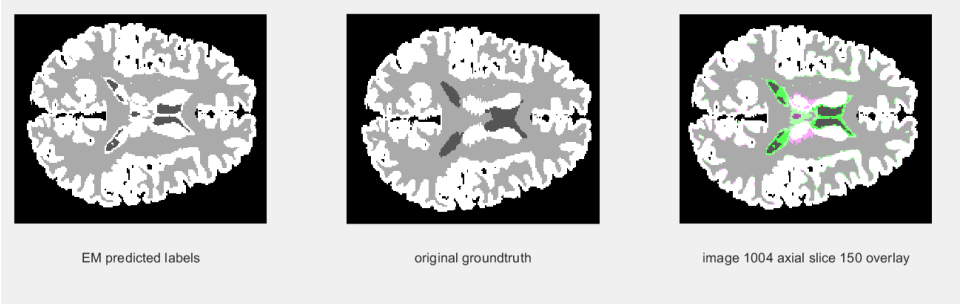
(a) atlas segmentation, dices (CSF, WM, GM) = 10, 64, 77



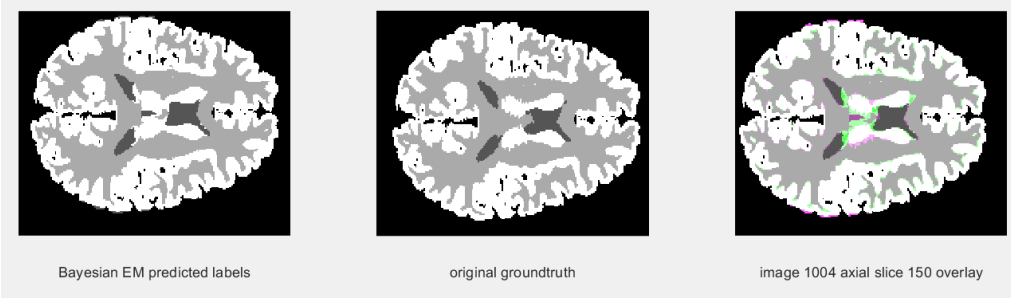
(b) tissue model segmentation, dices = 18, 90, 86



(c) Bayesian framework segmentation, dices= 13, 84, 87

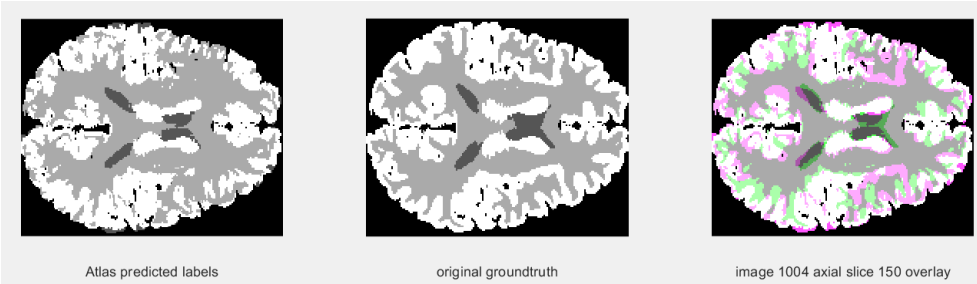


(d) EM segmentation, dices= 60, 92, 94

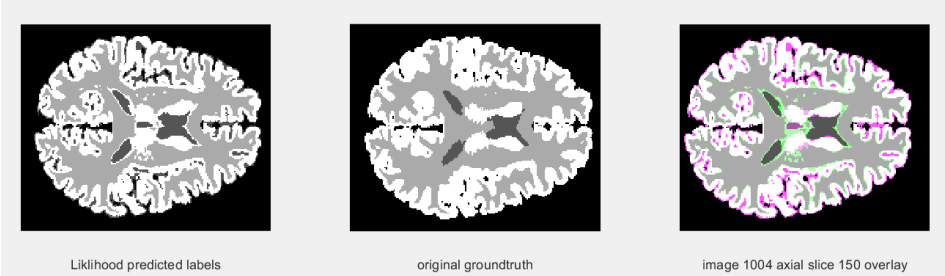


(e) Bayesian EM segmentation, dices= 12, 87, 89

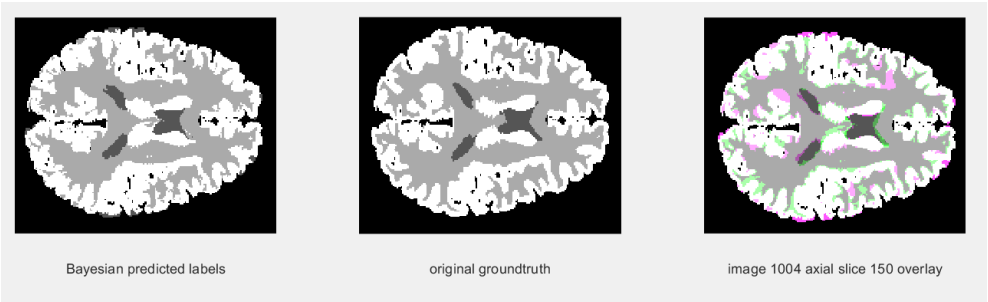
Figure 11: Image 1004, axial slice 150, qualitative evaluation of the five methods using MNI Atlas



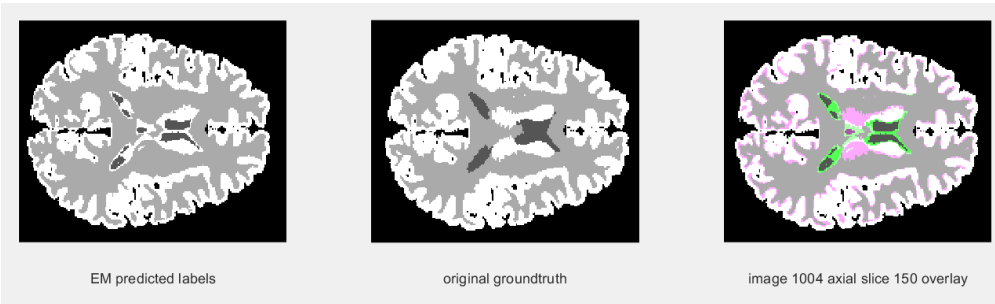
(a) atlas segmentation, dices (CSF, WM, GM) = 28, 67, 76



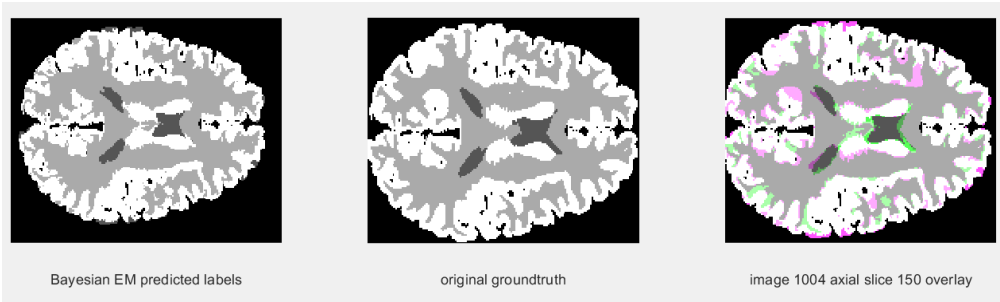
(b) tissue model segmentation, dices = 13, 87, 81



(c) Bayesian framework segmentation, dices= 34, 82, 87



(d) EM segmentation, dices= 62, 87, 89



(e) Bayesian EM segmentation, dices= 33, 85, 88

Figure 12: Image 1004, axial slice 150, qualitative evaluation of the five methods using small Atlas

References

[1] MAIA program, MIRA & MISA courses, Coursework 3. Atlas based segmentation