# New Expectation Maximization segmentation pipeline in Slicer 3

From April 6th to August 31th
At
Surgical Planning Laboratory (SPL)

Brigham & Women's Hospital
Harvard Medical School
75 Francis St.
Boston, MA 02115

Supervisors:  Ron KIKINIS - SPL, Harvard Medical School - kikinis@bwh.harvard.edu
Sylvain JAUME - CSAIL, Massashusetts Insitute of Technology - sylvain@csail.mit.edu

Referring Teachers:  Christine CAVARO-MENARD - M2 SIBM - christine.menard@univ-angers.fr
Dominique MARATRAY - ISEN Brest - dominique.maratray@isen.fr

by Nicolas RANNOU

# Acknowledgment

First of all, I would like to thank Ron KIKINIS who gave me the opportunity to carry out this internship at the Surgical Planning Laboratory of Harvard Medical School. He has always took the time discuss with me and to give me precious advices in my work. I would also like to thank Katie MASTROGIACOMO for her help on the administrative side, especially to obtain the visa.

Of course, my thoughts also go to Sylvain JAUME. We worked in close collaboration. He was very patient and comprehensive. Regarding my student status, he was very pedagogic to improve my knowledge in a lot of different ways and I am very grateful for that. He spent a lot of time to guide me and to help to achieve my internship.

I wish to express my gratitude to Steve PIEPER, Andriy FEDOROV and Daniel HAEHN for their assistance . They helped a lot me to get familiar with the whole working environment.

I must also acknowledge Christine CAVARO-MENARD who has accepted to supervise this internship and Dominique MARATRAY for her assistance.

Finally, I would like to thank all the people working in the laboratory for their reception and their friendliness. Thanks to them, my time in the Surgical Planning Laboratory was a very enriching experience and it was a real pleasure to work with them.

# Abstract

Many neuroanatomy studies rely on brain tissues segmentation of magnetic resonance (MR) images. Expectation-maximization (EM) algorithm are very popular framework for this task. The Surgical Planning Laboratory (SPL) of Harvard Medical School developed its own segmentation algorithm. Nevertheless, the segmentation is not accurate if the image exhibits intensity inhomogeneity. Moreover the optimum parameters are challenging to estimate.

In this context, this document first proposes a full description of the EM algorithms. Then solutions are presented to enhance the segmentation. Contributions are very various, from the algorithm to the graphic user interface (GUI). We add a bias field correction to correct the intensity inhomogeinities in the MR image, as a preprocessing step in the algorithm. We also propose new procedures in the graphic user interface to define the distribution of the tissues to be segmented. Some tools to make the segmentation process easier and more accurate are also presented. Finally, the results obtain with the new segmentation pipeline are evaluated by an expert.

Keywords: segmentation, expectation, maximization, normal distribution, Gaussian, bias correction, intensity inhomogeinity

# Résumé

Nombre d'études neuroanatomiques reposent sur la segmentation des tissus du cerveau à partir d'imagerie par résonnance magnétique (RM). Les algorithmes d'expectation-maximisation (EM) sont très populaires pour cette application. Le Surgical Planning Laboratory de Harvard Medical School a développé son propre algorithme de segmentation. Néamoins, la segmentation n'est pas précise si l'image présente des inhomogéinités d'intensité. De plus, les paramètres optimums ne sont pas évidents à estimer.

Dans ce contexte, ce document propose tout d'abord une description complète des algorithmes d'EM segmentation. Ensuite, des solutions sont apportées pour améliorer la segmentation. Les contributions sont très diverses, de l'algorithme jusqu'à l'interface graphique. Nous avons ajouté une correction du biais pour corriger les inhomogéinités d'intensité dans l'image RM, en tant que pré-traitement. De nouvelles méthodes sont proposées pour définir la distribution des tissus à segmenter, via l'interface graphique utilisateur. Des outils pour faciliter la segmentation sont également presentés. Finalement, les résultats obtenus en utilisant les nouvelles contributions sont evalués par un expert.

Mots clés : segmentation, expectation, maximisation, distribution normale, gaussienne, correction du bias, intensité inhomogène

# Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1    Context and motivation

Nowadays, medical image processing is becoming a major field of research in most of the laboratories. Indeed, because of the increasing complexity of the data they have to deal with, physicists need something to help. Help must be provided in many different ways. Before the surgery, to etablish a fast and accurate diagnosis. During the surgery to prevent physicians from errors and to help them to proceed to more precise moves. After the surgery, to see if it succeeded, or to follow the pathology of a patient. The informations brought to the physicians by the tools must be accurate, robust and provide a fast feedback.

In this context of pre and post operation, plenty of work has already been done. Nevertheless, there is still a lot of work to achieve. Regarding data storage and exchange, the increasing among of information leads us to find more approriate methods for the same purpose. Another interesting contribution of computer science to medicine is image segmentation. New methods have to be developed for a better diagnosis, or to detect new pathologies. A lot of segmentation techniques appeared like level-set segmentation, region growing or texture based segmentation. Each one is adapted to a specific problem like vessel segmentation or tumor detection. Another remarkable contribution is the segmentation based on expectation maximization (EM). It is very well suited for brain tissue segmentation.

For segmentation of brain MR images, the Surgical Planning Laboratory (SPL) at the Brigham and Women's Hospital, affiliated to Harvard Medical School, has developed an EM algorithm for segmentation. This model assumes that the MR volumes does not exhibit an intensity inhomogeinity. Moreover, the implementation is not widely used so far, because the complexity of the segmentation process. Finally, some steps have to be enhanced. In this report we will present an approach to enhance the segmentation for inhomogeneous MR images, correcting the intensity bias field. We will also provide tools to the end-user for an easier and more accurate segmentation process.

## 1.2    Contents

The main body of this report is divided as follows.

Chapter 2 focuses on the EM segmentation. Background is reminded and the algorithm used in the SPL is fully described. We also present in this chapter the segmentation pipeline developped in the SPL and the limitations of the current implementation. Chapter 3 describes our contributions. It explains the solutions we proposed to enhance the segmentation and to improve the usability of the current framework. Chapter4 shows the results obtained. It aslo discuss about the results obtained. An expert radiologist evaluates the segmentation. Finally, some perspectives about the future work to be done are presented.

# Chapter 2

# Expectation-maximization applied to brain segmentation

Here we get going with theory of the expectation-maximization (EM) applied to brain segmentation. We show first a simple approach of the problem, in the particular case of Gaussian mixture models followed by a more generalistic approach. The simple approach will give the reader an intuitive understanding of the problem then the general approach will formalize it in order to adapt it to most of the segmentation problems. Finally, there will be a presentation of the algorithm implemented in the software suit Slicer 3 [22].

## 2.1   Presentation of the EM segmentation

The EM algorithm was originally described in 1977 by Arthur Dempster, Nan Laird, and Donald Rubin[1]. They generalized and developped a method used previously by authors, for specific applications. It is widely used to solve problems where data are "missing". The EM algorithm is an iterative algorithm which works in two steps: Expectation and Maximization. It can be use to solve a lot of image processing's problems like classification, restoration[3] and motion estimation[2]. The EM segmentation has been applied to segmentation

Nowadays, EM algorithms are become a popular tool for classification problems. It is particulary well suited for brain MR images segmentation. A lot of algorithms already exist. They present complex frameworks using spatial information, neighborhood or intensity inhomogeneities to enhance the classification.

In the SPL, the algorithm developped uses spatial, tree structure arborescence and information about intensity inhomogeneity to segment the brain.

## 2.2   Background

We start with a presentation of all the fundamentals the reader needs to have a good understanding of EM segmentation. We begin with a description of the statistical model used for the brain. Then we describe briefly the widely used GM model. Finally we will present what the mamixum likelihood function is. This part is mainly based on [4], [5] and [6]

### 2.2.1   Statistical model used for the brain

We define the voxel intensities of a volume as $Y = \{y_1, ..., y_n\}$ when the image counts $n$ voxels. $Y$ is called observed data because this is the the data we see when we observe the image. Each $y$ is a realization of the random variable $Y$. The actual labelling of the image is $Z = \{z_1, ..., z_n\}$. $Z$ is called hidden data because we do not know the value of each label. This is precisely the purpose

of the segmentation: estimating the hidden data from the observed data. We assume that the observed data is generated from the hidden data and a parameter $\Phi$. The parameter $\Phi$ can either be a probability density function, noise,or bias field, among others, depending on the model.

$Y$ and $Z$ can be viewed as $n$-dimensional random variables $Y = \{Y_1, ..., Y_n\}$ and $Z = \{Z_1, ..., Z_n\}$ then each $y_i$ is a realization of $Y_i$ and each $z_i$ is a realization of $Z_i$. The conditional probability function describing $Y_i$ is $p(Y_i|Z_i, \Phi)$.

The easiest model assumes that all intensities in one class are the same, but these intensities are corrupted by factors like noise modelled by a Gaussian distribution. We can describe the relationship as below:

$$y_i = \mu_k + n_i$$

where $\mu_k$ is the mean intensity of the $k^{th}$ tissue and $n_i$ a random sample generated by the corrupting factor(s). Let's say that $n_i$ is generated by a Gaussian probability distribution function $G(., 0, \sigma)$, with 0 mean and $\sigma$ variance. That means that $y_i$ is a random sample generated by a Gaussian probability density function $G(., \mu_k, \sigma)$. Let's assume that each class has a different variance, $G(., \mu_k, \sigma)$ becomes $G(., \mu_k, \sigma_k)$ and it leads to:

$$p(y_i|Z_i = k, \Phi) = G(y_i, \mu_k, \sigma_k) \tag{2.1}$$

As the labelling is not known, it is usefull to express the probability density function (PDF) of $Y_i$ only depending on parameter $\Phi$ with the total probability theorem:

$$p(y_i|\Phi) = \sum_{k=1}^{K} p(y_i|Z_i = k, \Phi)p(Z_i = k|\Phi) \tag{2.2}$$

$p(Z_i = k|\Phi)$ is the prior probability. It expresses the probability that a voxel $i$ belongs to a class $k$. $p(y_i|Z_i = k, \Phi)$ is the likelihood. In our case, we will assume that the prior probability is unique for each each label. The new model we obtain for the labelling is a widely used one: the gaussian mixture model.

### 2.2.2   Gaussian mixture model

Let's remind the first hypotesis: the conditional probability function for each tissue to segment is defined as in Equation 2.1. Moreover, we will assume that prior probability is a constant $c_k$ for each class $k$. $c_k$ is the weight of the class $k$.

$$p(Z_i = k|\Phi) = c_k \tag{2.3}$$

The last assumption will be that $\Phi$ contains unknown means, variances and weights for each tissue. Then we can express $\Phi$ as a set of parameters such as $\Phi = (\mu_1, \sigma_1, c_1, ..., \mu_K, \sigma_K, c_K)$.

Using Equations 2.1 and 2.3, Equation 2.2 becomes:

$$p(y_i|\Phi) = \sum_{k=1}^{K} G(y_i, \mu_k, \sigma_k)c_k \tag{2.4}$$

In the case of GM model, each voxel is considered to be independent. That means that each voxel will have its own probability density function. Consequently, the normalized histogram of the whole volume can be interpreted as an approximation of the sum of all the probability density functions.

### 2.2.3   Maximum likelihood

In our case, we know the intensity of each observed voxel $y_i$. $\Phi$ are to be found. The best estimation of $\Phi$ will be obtain using the maximum likelihood principle. $p(y_i|\Phi)$ is called likelihood function. For each value of $\Phi$ it returns the value of the likelihood of $y_i$, given $\Phi$.

The voxels are considered to be independent. Through the whole volume, it leads to:

$$p(Y|\Phi) = \prod_{i=1}^{n} p(y_i|\Phi) \tag{2.5}$$

The objective is to find the parameter $\Phi$ which will maximize the likelihood of the observed volume. We can note this parameter:

$$\hat{\Phi} = \arg\max_{\Phi} p(Y|\Phi) \tag{2.6}$$

Therefore, it is more convenient to work with logarithm because the product from Equation 2.5 will be converted into a summation. Equation 2.6 becomes:

$$\hat{\Phi} = \arg\max_{\Phi} \log p(Y|\Phi).$$

With Equation 2.2, let us denote:

$$L(\Phi) \triangleq \log p(Y|\Phi)$$

$$= \sum_{i=1}^{n} \log \sum_{k=1}^{K} p(y_i|Z_i = k, \Phi) p(Z_i = k|\Phi)$$

Finally, in case of GM model, with Equation 2.4, $L(\Phi)$ becomes:

$$L(\Phi) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} G(y, \mu_k, \sigma_k) c_k$$

The *log* likelihood can be maximized by finding partial derivatives for each parameter. When the partial derivative of $L(\Phi)$ is 0 for a parameter, we have the maximum likelihood for the parameter $\Phi$.

For example, to find the maximum likelihood regarding $\mu_k$ we first have to find when:

$$\frac{\partial}{\partial \mu_k}(L(\Phi)) = 0$$

Then we compute the partial derivative of $L(\Phi)$ over $\mu_k$:

$$\frac{\partial}{\partial \mu_k}(L(\Phi)) = \frac{\partial}{\partial \mu_k}(\sum_{i=1}^{n} \log \sum_{k=1}^{K} G(y_i, \mu_k, \sigma_k) c_k)$$

$$= \sum_{i=1}^{n} \frac{G(y_i, \mu_k, \sigma_k) c_k}{\sum_{j=1}^{K} G(y_i, \mu_j, \sigma_j) c_j} \frac{\partial}{\partial \mu_k}(-\frac{(y_i - \mu_k)^2}{2\sigma_k^2})$$

$$= \sum_{i=1}^{n} \frac{G(y_i, \mu_k, \sigma_k) c_k}{\sum_{j=1}^{K} G(y_i, \mu_j, \sigma_j) c_j}(\frac{(y_i - \mu_k)}{\sigma_k^2})$$

$$= \sum_{i=1}^{n} \frac{p(y_i|Z_i = k, \Phi) p(Z_i = k|\Phi)}{\sum_{j=1}^{K} p(y_i|Z_i = j, \Phi) p(Z_i = j|\Phi)}(\frac{(y_i - \mu_k)}{\sigma_k^2}) \tag{2.7}$$

Using Bayes' theorem (see App. A, Sec. A.2), we notice that:

$$p(Z_i = k | y_i, \Phi) = \frac{p(y_i | Z_i = k, \Phi) p(Z_i = k | \Phi)}{\sum_{j=1} p(y_i | Z_i = j, \Phi) p(Z_i = j | \Phi)} \qquad (2.8)$$

Thus, setting the denominator to 0 in Equation 2.7 and using Equation 2.8 yields:

$$\sum_{i=1}^{n} p(Z_i = k | y_i, \Phi)(y - \mu_k) = 0 \qquad (2.9)$$

Let us denote

$$p_{ik} = p(Z_i = k | y_i, \Phi) \qquad (2.10)$$

Equation 2.9 leads to:

$$\mu_k = \frac{\sum_{i=1}^{n} y_i p_{ik}}{\sum_{i=1}^{n} p_{ik}} \qquad (2.11)$$

Proceeding the same way as we did for Equation 2.9, we can get similar equations for variance $\sigma_k$ and weight $c_k$.

We find that:

$$\sigma_k^2 = \frac{\sum_{i=1}^{n} (y_i - \mu_k)^2 p_{ik}}{\sum_{i=1}^{n} p_{ik}} \qquad (2.12)$$

$$c_k = \frac{1}{n} \sum_{i=1}^{n} p_{ik} \qquad (2.13)$$

Equations 2.11,2.12 and 2.13 provides us

$$p_{ik} = \frac{G(y_i, \mu_k, \sigma_k) c_k}{\sum_{j=1}^{K} G(y_i, \mu_j, \sigma_j) c_j} \qquad (2.14)$$

This equation expresses that a voxel $i$ belongs to the class $k$. It is formulation for soft segmentation. $p_{ik}$ is called soft assignment. $p_{ik}$ is used to fill a "map" of soft segmentation. At the end of the segmentation process, this map contains the probability that the voxel $i$ belongs to class $1, 2, ..., K$. We determine the class of the voxel $i$ looking at the class which has the highest probability in the map for this given voxel $i$.

The segmentation can now be calculated following an iterative process called expectation maximization.

## 2.3   Expectation maximization algorithm

The EM algorithm is a method to find the maximum likelihood for a given set of parameter ($\Phi$ in our case). Here we start with an intuitive description of the algorithm in the particular case of Gaussian mixture model then we will present a more general definition.

**Algorithm in case of Gaussian mixture data model**

Let's assume that we can find the maximum likelihood of the hidden data by a direct differentiation (because of the Gaussian mixture model). The EM algorithm is an iterative process of two steps: the expectation step (E-Step) and the maximization step (M-Step). At each iteration, the maximum likelihood will be increased until convergence is reached.

- **E-step**

  In this step, we calculate an estimation of soft segmentation $p^{(m+1)}$ with Equation 2.14 as below. We know all the variables needed for the calculation from the observed data and the current parameter estimate $\Phi^{(m)}$. Note that an initialization is necessary for the first iteration.

$$p_{ik}^{(m+1)} = \frac{G(y_i, \mu_k^{(m)}, \sigma_k^{(m)})c_k^{(m)}}{\sum_{j=1}^{K} G(y_i, \mu_j^{(m)}, \sigma_j^{(m)})c_j^{(m)}}$$

- **M-step**

  In this step, we estimate the maximum likelihood for parameter $\Phi^{(m+1)}$. We do it with Equations 2.11,2.12 and 2.13 as below. We know all the variables needed for the calculation from the observed data and the current estimate $p^{(m+1)}$ of hidden data.

$$\mu_k^{(m+1)} = \frac{\sum_{i=1}^{n} y_i p_{ik}^{(m+1)}}{\sum_{i=1}^{n} p_{ik}^{(m+1)}}$$

$$(\sigma_k^{(m+1)})^2 = \frac{\sum_{i=1}^{n} (y_i - \mu_k^{(m+1)})^2 p_{ik}^{(m+1)}}{\sum_{i=1}^{n} p_{ik}^{(m+1)}}$$

$$c_k^{(m+1)} = \frac{1}{n} \sum_{i=1}^{n} p_{ik}^{(m+1)}$$

The problem is simple as long as we are working with Gaussian mixture model. In the other case, the log-likelihood can not be maximized by direct differentiation and a generalized approach must be used.

**Generalized algorithm**

Now we assume that we are no longer working with Gaussian mixture model. Thus, we must use a more general algorithm. To explain the general algorithm, we will start from the log-likelihood $L(\Phi)$. As presented in the previous subsection, we wish to find $\Phi$ such as $p(Y|\Phi)$ is a maximum. We have:

$$L(\Phi) = \log p(Y|\Phi)$$

Since $log$ is a strictly increasing function, the value of $\Phi$ which maximizes $p(Y|\Phi)$ also maximizes $L(\Phi)$. Assume that after the $m^{th}$ iteration, the current estimate of $\Phi$ is given by $\Phi^{(m)}$. Since the objective is to maximize $L(\Phi)$ we want an updated estimate $\Phi$ such that

$$L(\Phi) > L(\Phi^{(m)})$$

In other words, we want to maximize the difference $L(\Phi) - L(\Phi^{(m)})$. For convenience, we introduce a new variable $z_{ik}$ which means that $Z_i = k$. Using the new notation, we can transform this difference as below:

$$L(\Phi) - L(\Phi^{(m)}) = \log p(Y|\Phi) - \log p(Y|\Phi^{(m)})$$

$$= \sum_{i=1}^{n} \{\log \sum_{k=1}^{K} p(y_i|z_{ik}, \Phi)p(z_{ik}|\Phi) - \log p(y_i|\Phi^{(m)})\}$$

$$= \sum_{i=1}^{n} \{\log \sum_{k=1}^{K} p(y_i|z_{ik}, \Phi)p(z_{ik}|\Phi). \frac{p(z_{ik}|y_i, \Phi(m))}{p(z_{ik}|y_i, \Phi^{(m)})} - \log p(y_i|\Phi(m))\}$$

$$= \sum_{i=1}^{n} \{\log \sum_{k=1}^{K} p(z_{ik}|y_i, \Phi^{(m)}). \frac{p(y_i|z_{ik}, \Phi)p(z_{ik}|\Phi)}{p(z_{ik}|y_i, \Phi_n)} - \log p(y_i|\Phi_n)\}$$

$$\geq \sum_{i=1}^{n} \{\sum_{k=1}^{K} p(z_{ik}|y_i, \Phi^{(m)}) \log \frac{p(y_i|z_{ik}, \Phi)p(z_{ik}|\Phi)}{p(z_{ik}|y_i, \Phi^{(m)})} - \log p(y_i|\Phi^{(m)})\}$$

We can deduce this inequality from Jensen's inequality (see App. A, Sec. A.3) since $p(z_{ik}|y_i, \Phi^{(m)})$ is a probability mesure and *log* a concave function ([5]).

We will then use the fact that $\sum_k p(z_{ik}|y_i, \Phi^{(m)}) = 1$. In this case, it leads to $\log p(Y|\Phi^{(m)}) = \sum_k p(z_{ik}|y_i, \Phi^{(m)}) \log p(Y|\Phi^{(m)})$. This allows us to bring $\log p(Y|\Phi^{(m)})$ into the summation.

We will also use a new variable $e_k$ to express the difference through the whole volume without a summation. $e_k$ is defined as $e_k = \{z_{1k}, ..., z_{nk}\}$ when the image consists of $n$ voxels. For example, $z_{ik} = e_k = \{0, ..., 0, 1, 0, ..., 0\}$ means that voxel $i$ belongs to class $k$.

$$L(\Phi) - L(\Phi^{(m)}) \geq \sum_{i=1}^{n} \{\sum_{k=1}^{K} p(z_{ik}|y_i, \Phi^{(m)}) \log \frac{p(y_i|z_{ik}, \Phi)p(z_{ik}\Phi)}{p(z_{ik}|y_i, \Phi^{(m)})} - \log p(Y|\Phi^{(m)})\}$$

$$= \sum_{i=1}^{n} \{\sum_{k=1}^{K} p(z_{ik}|y_i, \Phi^{(m)}) \log \frac{p(y_i|z_{ik}, \Phi)p(z_{ik}|\Phi)}{p(z_{ik}|y_i, \Phi^{(m)})p(y_i|\Phi^{(m)})}\}$$

$$= \sum_{k=1}^{K} p(e_k|Y, \Phi^{(m)}) \log \frac{p(Y|e_k, \Phi)p(e_k|\Phi)}{p(e_k|Y, \Phi^{(m)})p(Y|\Phi^{(m)})}$$

$$\triangleq \Delta(\Phi|\Phi^{(m)})$$

We can then conclude that:

$$L(\Phi) \geq L(\Phi^{(m)}) + \Delta(\Phi|\Phi^{(m)})$$
$$\geq l(\Phi|\Phi^{(m)})$$

where $l(\Phi|\Phi^{(m)}) \triangleq L(\Phi(m)) + \Delta(\Phi|\Phi^{(m)})$.

We have now a function $l(\Phi|\Phi^{(m)})$ which is bounded above by $L(\Phi)$. Additionnaly, we observe that $(l(\Phi^{(m)}, \Phi^{(m)})) = L(\Phi^{(m)})$.

Our objective is to choose values of $\Phi$ which will maximize $L(\Phi)$. We have shown that the function $l(\Phi, \Phi^{(m)})$ is bounded above by the likelihooh function $L(\Phi)$. Moreover, the value of $\Phi$ for which the function $(l(\Phi, \Phi^{(m)}))$ and $L(\Phi)$ are egals is $\Phi = \Phi^{(m)}$. Therefore, any $\Phi$ which will increase $l(\Phi, \Phi^{(m)})$ will increase $L(\Phi)$. In order to maximize $L(\Phi)$ as much as possible, we maximize $L(\Phi)$ such as: $L(\Phi(m)) = l(\Phi^{(m)}, \Phi^{(m)}) < l(\Phi^{(m+1)}, \Phi^{(m)}) < L(\Phi^{(m+1)})$.... As soon as $L(\Phi(m+1)) < L(\Phi(m))$, the convergence has been reached and the maximum likelihood found. We can formalize this search. $\Phi^{(m+1)}$ is the updated value which is found after maximization of $\Phi$ using $\Phi^{(m)}$.

$$\Phi^{(m+1)} = \arg\max_{\Phi}\{l(\Phi|\Phi^{(m)})\}$$

$$= \arg\max_{\Phi}\{L(\Phi^{(m)}) + \sum_{k=1}^{K} p(e_k|Y,\Phi^{(m)})\log\frac{p(Y|e_k,\Phi)p(e_k|\Phi)}{p(e_k|Y,\Phi^{(m)})p(Y|\Phi^{(m)})}\}$$

As $p(e_k|Y,\Phi^{(m)})$ and $p(Y|\Phi^{(m)})$ do not depend on $\Phi$

$$= \arg\max_{\Phi}\{\sum_{k=1}^{K} p(e_k|y_i,\Phi^{(m)})\log p(Y|e_k,\Phi)p(e_k|\Phi)\}$$

$$= \arg\max_{\Phi}\{\sum_{k=1}^{K} p(e_k|Y,\Phi^{(m)})\log\frac{p(Y,e_k,\Phi)p(e_k,\Phi)}{p(e_k,\Phi)p(\Phi)}\}$$

$$= \arg\max_{\Phi}\{\sum_{k=1}^{K} p(e_k|Y,\Phi^{(m)})\log p(Y,e_k|\Phi)\} \tag{2.15}$$

$$= \arg\max_{\Phi}\{E_{Z|Y,\Phi^{(m)}}\{\log p(Y,Z|\Phi)\}\}$$

We notice from Equation 2.15, that for a given voxel $i$:

$$\sum_{k=1}^{K} p(e_k|y_i,\Phi^{(m)})\log p(y_i,e_k|\Phi) = \sum_{k=1}^{K} p_{ik}^{(m+1)}\log p(y_i,e_k|\Phi) \tag{2.16}$$

Now, both expectation and maximization steps are apparents.

- **E-step**

  This is the expectation step. During this step, we estimate the probability that the pixel $i$ belongs to class $k$ regarding $\Phi_{(m)}$. This equation is obtained from Equation 2.16 with Bayes formula.

  $$p_{ik}^{(m+1)} = \frac{p(y_i|e_k,\Phi^{(m)})p(e_k|\Phi^{(m)})}{\sum_{j=1}^{K} p(y_i|e_j,\Phi^{(m)})p(e_j|\Phi^{(m)})} \tag{2.17}$$

  Using this probability, $E_{Z|Y,\Phi^{(m)}}$ returns the expected value of the parameter $\Phi$ regarding $\Phi^{(m)}$.

- **M-step**

  This is the maximization step. During this step $\arg\max_{\Phi}$ maximizes $E_{Z|Y,\Phi^{(m)}}$ for the parameter $\Phi$. It returns $\Phi^{(m+1)}$

  $$\arg\max_{\Phi}\{E_{Z|Y,\Phi^{(m)}}\{\log p(Y,Z|\Phi)\}\} \tag{2.18}$$

The EM algorithm iterates until convergence is reached. The condition of convergence can differ from an algorithm to another. A possibility is to fix the number of iterations, $C$, of the algorithm. Most of the time, the following approach is used; convergence is reached when the difference between the estimation of the parameter $\Phi$, at step $m$ and step $m+1$ is smaller than $\varepsilon$:

$$\Phi^{(m+1)} - \Phi^{(m)} < \varepsilon$$

If after the $C^{th}$ iteration, this condition is not satisfied, the EM algorithm is stopped.

We are now familiar with the theory behind the EM algorithm. The logic appears and we can summarize the basic algorithm with the Figure 2.1:
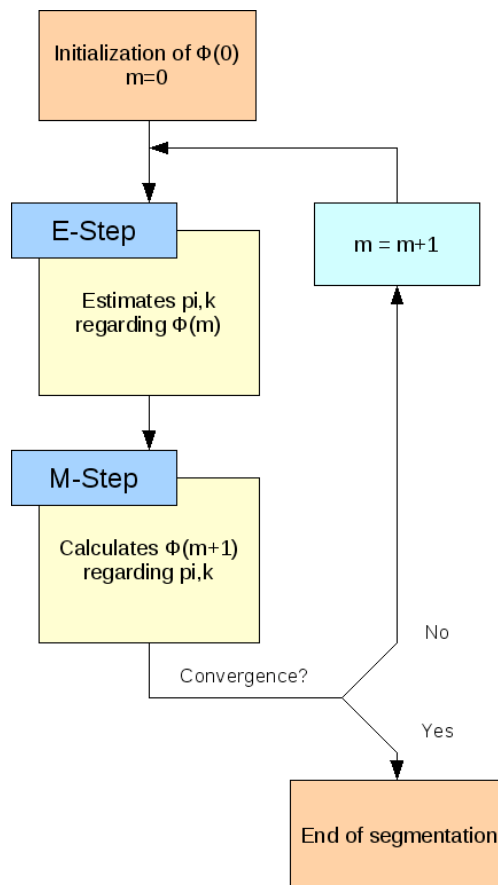
Figure 2.1: Illustration of the basic EM algorithm

## 2.4   Expectation maximization algorithm used in Slicer 3

In this part, we present the EM algorithm which has been developped in Slicer 3 and the pipeline in which it has been integrated as well. Finally, we briefly discuss the limitations of this one.

The EM algorithm which is used in the SPL is derived from the original one. It enhances the original algorithm by the addition of informations like a probalistic atlas, a multichannel segmentation, a bias correction and a structure information. In Slicer 3, the Gaussian mixture model is used to describe the tissues to segment. Thus it will simplify the problem and the notations.

### 2.4.1   Probabilistic atlas

The EM algorithm is very sensitive to initialization since it only finds local extremums during the maximization step. A solution to enhance the initialization is to use atlases. An atlas is needed for each tissue the user wants to segment. For each voxel $i$ of the volume, the atlas returns the probability that this voxel belongs to class $k$. This probability can be used as initialization.

$$p_{ik}^{(0)} = p_{ik}^{atlas}$$

From this value, it estimates $\Phi^{(1)}$ and the algorithm iterates until convergence is reached. The probabilistic atlases are not only used to inialize the process. It is also use to get a more robust algorithm. Indeed, we can use the spatial information given by the atlases. Voxels will be classified not only based on intensity but regarding spatial position too. Van Lemput *et al.* ([8] and [9]) used the spatial prior at each iteration. It is constant and we then have a spatial information.

The probability that a pixel $i$ belongs to class $k$, in the E-Step changes. From Equation 2.17, it becomes:

$$p_{ik}^{(m+1)} = \frac{p(y_i|e_k, \Phi^{(m)} p_{ik}^{atlas}}{\sum_{j=1}^{K} p(Y_i|e_j, \Phi^{(m)}) p_{ik}^{atlas}}$$

## 2.4.2   Multichannel segmentation

Most of the time, several sequences of a given modality are used to process brain segmentation. Indeed, the best suited sequence to use depends on the tissue you want to segment. For example, MR images provided by T1 sequences [12] are well suited to segment white matter (WM) but are not accurate for cerebrospinal fluid (CSF). On the contrary, T2 sequences [12] MR images which are well suited for CSF and not for WM. To formalize the utilization of different MR images sequence (i.e. different channels)s during the segmentation, we will change the definition of $y_i$, $\mu_k$ and $\sigma_k$ we did at the beginning. Let $y_i = \{y_{i1}, y_{i2}, ..., y_{iR}\}$, $\mu_k = \{\mu_{k1}, \mu_{k2}, ..., \mu_{kR}\}$ and $\sigma_k = \{\sigma_{k1}, \sigma_{k2}, ..., \sigma_{kR}\}$ when we use $R$ channels, from different modalities to do the segmentation. The equations for the E-Step and the M-Step will remain the same.

## 2.4.3   Bias field correction

A major issue in MR modality is that the images can be corrupted by a low field bias field. It is mainly due to equipment limitations and to patient induced electrodynamic interactions ([12]). We will now present how this bias field can be estimated and corrected in the EM algorithm.

### Principle

Let $I = (I_1, ..., I_n)$ the observed intensities in an image, $I^* = (I_1^*, ..., I_n^*)$ the ideal intensities and $F = (F_1, ..., F_n)$ the bias field. We use the assumption that the bias field is only multiplicative. Degradation of each voxel can then be expressed as:

$$I_i = I_i^* F_i$$

Let $Y = (Y_1, ..., Y_n)$ and $Y^* = (Y_1^*, ..., Y_n^*)$ be the log-transformed observed and ideal intensities.$B = (B_1, ..., B_n)$ the log-bias field. This transform makes the bias field become additive instead of multiplicative without the log-approach.

$$Y_i = Y_i^* + B_i$$

We can model the PDF of the voxel intensity with a Gaussian distribution

$$p(y_i|e_k, \Phi, B) = G(y_i - b_i, \mu_k, \sigma_k)$$

The low frequency characteristic of the bias field $B$ can be modeled by a linear combination of smooth basic functions $\Psi_l(x)$ ([13]). Let $b_i$ be the realization of the random variable $B_i$

$$b_i = \sum_{l=1}^{L} a_l \Psi_l(pos(i))$$

$pos(i)$ returns the 3D position $(x, y, z)$ of the voxel $i$. $a_i$ is the $i^{(th)}$ value of the vector $A = (a_1, ..., a_L)$. $A$ represents the bias field parameters.

In the Gaussian model, bias field can then be estimated using EM framework. The bias field parameter $A$ will be used during the E-Step , through $b_i$ to estimate the soft segmentation. $A$ will be re-estimated during the M-Step, after the maximization of the tissue class parameters (mean, variance and weight). Van Leemput formalised the two steps as below ([8] and [9]):

- **E-step**

$$p_{ik}^{(m+1)} = \frac{G(y_i - b_i, \mu_k^{(m)}, \sigma_k^{(m)}) p_{ik}^{atlas}}{\sum_{j=1}^{K} G(y_i - b_i, \mu_j^{(m)}, \sigma_j^{(m)}) p_{ij}^{atlas}}$$

- **M-step**

  - Gaussian distribution parameters estimation

$$\mu_k^{(m+1)} = \frac{\sum_{i=1}^{n} y_i p_{ik}^{(m+1)} - b_i}{\sum_{i=1}^{n} p_{ik}^{(m+1)}}$$

$$(\sigma_k^{(m+1)})^2 = \frac{\sum_{i=1}^{n} (y_i - \mu_k^{(m+1)} - b_i)^2 p_{ik}^{(m+1)}}{\sum_{i=1}^{n} p_{ik}^{(m+1)}}$$

  - Bias field correction

$$(A^{(m+1)})^T = (F^T W^{(m+1)} F)^{-1} F^T W^{(m+1)} R^{(m+1)} \tag{2.19}$$

with:

$$\mathbf{F} = \begin{pmatrix} \Psi_1(pos(1)) & \Psi_2(pos(1)) & \dots & \Psi_L(pos(1)) \\ \Psi_1(pos(2)) & \Psi_2(pos(2)) & \dots & \Psi_L(pos(2)) \\ \vdots & \vdots & \ddots & \\ \Psi_1(pos(N)) & \Psi_2(pos(N)) & \dots & \Psi_L(pos(N)) \end{pmatrix}$$

$$\mathbf{W^{(m+1)}} = \begin{pmatrix} \sum_{k=1}^{K} w_{1k}^{(m+1)} & 0 & \dots & & 0 \\ 0 & \sum_{k=1}^{K} w_{2k}^{(m+1)} & \dots & & 0 \\ \vdots & \vdots & \ddots & & \\ 0 & \dots & & 0 & \sum_{k=1}^{K} w_{Nk}^{(m+1)} \end{pmatrix}$$

$$w_{ik}^{(m+1)} = \frac{p_{ik}^{(m+1)}}{(\sigma_k^{(m+1)})^2}$$

$$\mathbf{R} = \begin{pmatrix} y_1 - \tilde{y}_1^{(m+1)} \\ \vdots \\ y_N - \tilde{y}_N^{(m+1)} \end{pmatrix}$$

$$\tilde{y}_i^{(m+1)} = \frac{\sum_{k=1}^{K} w_{ik}^{(m+1)} \mu_{ik}^{(m+1)}}{\sum_{k=1}^{K} w_{ik}^{(m+1)}}$$

The bias field correction can be interpreted as follows: the estimated soft segmentation $(p_{ik}^{(m)})$ and tissue class parameters are used to reconstruct the image $\tilde{Y} = (\tilde{y}_1, ..., \tilde{y}_n)$. This new image is supposed to be not corrupted by the bias field. We then substract the reconstructed image $\tilde{Y}$ from the observed image $Y$. We obtain the residual image $R$. From $R$, we estimate the bias field. $F$ represents the discretized geometry of the bias field. $W$ is an inverse covariance matrix. It returns information about the possible error for each voxel. The covariance matrix is described in details in [17].

The approach used in Slicer 3 is based on the same principle but differs regarding the maximization method and the parameter which is maximized.

**Variation used in Slicer 3**

In this method, we are working with a Gaussian mixture model. Moreover the parametres of this Gaussian distribution are assumed to be known. The idea of estimating the field in EM framework was originally proposed by Wells *et al.* ([10]). He proposed to only use maximization to re-estimate the bias field. The maximum a posteriori principle (MAP) instead of the maximum likelihood principle (Equation 2.6) is used to find the lower bound, during the maximization:

$$\hat{\Phi} = \arg\max_{\Phi} p(\Phi|Y)$$

As Bayes's theorem can be applied

$$= \arg\max_{\Phi} \frac{p(Y|\Phi)p(\Phi)}{pY}$$

As $p(Y)$ do not depend on $\Phi$

$$= \arg\max_{\Phi} p(Y|\Phi)p(\Phi)$$

Proceeding the same way as we did in section 2.3, the new E-Step becomes:

$$E_{MAP} = E_{Z|Y,\Phi^{(m)}}\{\log p(Y, Z|\Phi)\} + \ln p(\Phi) \tag{2.20}$$

In Wells' method, the only parameter to estimate is the bias field. We assume that the noise has a Gaussian distribution:

$$p(\Phi) = p(B) = G(B, 0, \Sigma_B)$$

The equation for the bias field will change. We add the smoothness constraint in the Gaussian distribution: $\Sigma_B^{-1}$. We also set $F$ to a matrix filled with ones as no parametric model for the bias field is assumed. Finally, we define the mean residual image $\bar{M}^{(m+1)}$

$$\bar{M}^{(m+1)} = W^{(m+1)}R^{(m+1)}$$

The Equation 2.19 for the bias field will then be remplaced by $(B^{(m+1)})^T$:

$$(B^{(m+1)})^T = (W^{(m+1)+\Sigma_B^{-1}})^{-1}\bar{M}^{(m+1)}$$

### 2.4.4   Hierarchical information

The last modification of the original EM algorithm is the addition of hierarchical information in the iterative segmentation process. The algorithm was described by Pohl *et al.* [11] The idea was to describe the structures we want to segment as a tree. It allows us to subdivide the segmentation process into subproblems, that are easier to solve.

Here we continue with an intuitive description of the process. It is a brief explanation of how the image would be segmented using the hierarchical information 2.2. At the first iteration, the MR image will be segmented into the background (BG) and the intracranial cavity (ICC) with the EM algorithm. At the second iteration, the BG will be segmented into the air and the skull. Finally, at the last iteration, the ICC will be segmented into white matter (WM), grey matter (GM) and cerebrospinal fluid (CSF).

To formalize it, we incorporate $H$, a set of structure-specific information in Equation 2.20. $H$ contains a lot of information like the structures of the tree which have to be segmented, an
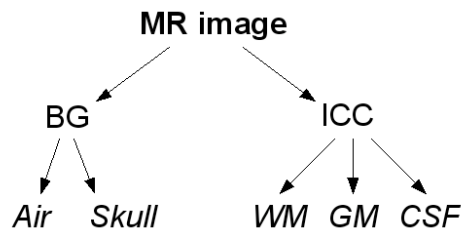
**MR image**

BG　　　　　　　ICC

Air　Skull　　　WM　GM　CSF

Figure 2.2: A simple tree structure of the brain
The brain is first divided into the background (BG) and the intracranial cavity (ICC). Second the BG is dived into Air and Skull. Finally, the ICC is divided into white matter (WM), grey matter (GM) and cerebrospinal fluid (CSF)

approximative size of the structure to be segmented (the Global Prior Weight) and information about which modality is the best suited to segment this structure.

$$\Phi^{(m+1)} = \arg\max_{\Phi}\{\sum_{k=1}^{K}(p(e_k|Y,\Phi^{(m)},H)\log p(Y,e_k|\Phi,H)) + \log p(\Phi,H)\}$$

### 2.4.5　Summary

We have shown that the EM algorithm is very flexible and can be transformed to solve a lot of segmentation problems. It is very well suited for segmentation of MR brain images and we can add a lot of informations through this algorithm to enhance the segmentation. The iterative general process is divided in two steps: the expectation (E-Step) and the maximization (the M-Step).

- **E-Step**
  Estimates a soft segmentation ($p_{ik}$), given parameter $\Phi^{(m)}$. The soft segmentation creates a map of probability. Each voxel contains the probabilities that it belongs to each class. It is used for the final segmentation.

- **M-Step**
  Estimates $\Phi^{(m+1)}$, using the soft segmentation done in the E-Step.

  - Estimates the intensity distribution for each tissue to be segmented.
  - Estimates the bias field, $\Phi^{(m+1)}$, using the soft segmentation and the intensity class distribution

The image will be segmented until the whole tree has been processed as described in section 2.4.4.

We can also describe the segmentation process in the Figure 2.3. The EM segmentation box represents the figure 2.1. To describe how it works, we use the tree structure presented in Figure 2.2. It first segments the node $n = 0$, i.e., BG and ICC. Once the EM Segmentation has converged, BG and ICC have been segmented and we move to the next node. Air and Skull will then be segmented. Following this process, all the structures of the tree will be segmented.

## 2.5　Workflow in Slicer 3

We will now present the whole segmentation workflow used in Slicer 3. It will describe all the initialisation steps done by the user, via the graphical user interface (GUI). It will also present how the whole algorithm works. We will explain why each initialisation step is important and where this information is used.
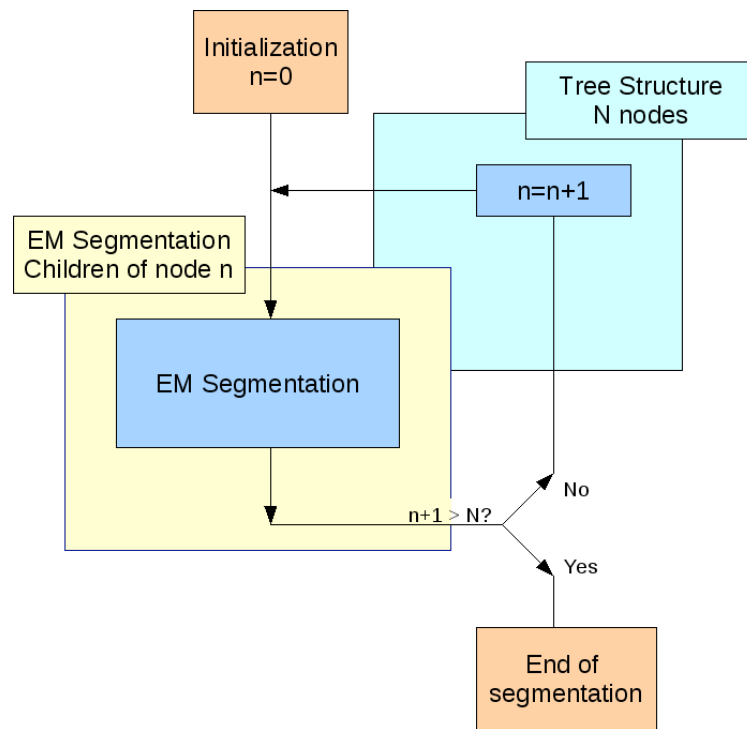
Figure 2.3: Illustration of the EM segment algorithm in Slicer

## 2.5.1   User interface

It consists of a manual initialization of the parameters which are required for the segmentation on Slicer 3. The user chooses the optimal values via the GUI.

- **Step 1: Tree structure creation**

    We first create our problem specific tree structure for the segmentation. It will be used to define $H$ (section 2.4.4).

- **Step 2: Atlas assignment**

    We assign to each node of the tree, i.e. each tissue to segment, the related atlas. It will be used for the spatial information (section (2.4.1)). It implies that an atlas for each structure to segment is needed.

- **Step 3: Mutlichannel segmentation**

    We choose the images we want to use for the segmentation. As discussed in section (2.4.2), it is usefull for the multichannel segmentation since some tissues are not visible in some sequence, in some modalities.

- **Step 4: Intensity normalization**

    We choose the value for the intensity normalization. We normalize the intensity of the images to be segmented, regarding the related atlas. The utility of this normalization step is presented in the next section.

- **Step 5: Class definition**

    We define mean value, variance and covariance for each class and modality. It is an accurate way to initialiaze the class tissues distribution for the algorithm. It it useful because the EM algorithm only estimates the bias field but still requires information about the tissues to be segmented. Using these values, the algorithm is initialized and estimates the bias field.

- **Step 6: Hierarchical parameters**

    Here we set some parameters for the hierachical segmentation. We define the utility of each target image, of the atlas for each tissue to segment and approximated the size of the tissue to be segmented. This will be stored in $H$ (section 2.4.4). For each tissue, $H$ knows now useful informations like which volume is the more relevant for the class and the size that the class is supposed to be.

    Let's take the example of the CSF. We assume a multichannel segmentation where T1 and T2 MR images are available. We give a weight of one (maximum) to the T2 target volume and zero (minimum) to the T1 target volume. It means that the only relevant information for CSF will be in the T2 volume. The algorithm will act accordingly and only use the information from T2 to perform the segmentation. It is the same for the atlas. If we set the weight of the atlas to one, the algorithm will use the spatial information. If we set it to zero, it will ignore the spatial information.

- **Step 7: Registration method**

    We choose the type of registration we want. Different kind of registrations are available. The default registration method is a non-rigid registration. The rationale behind selecting a specific registration method is presented in the next section.

### 2.5.2  Algorithm

After all the initialization steps done via the GUI (section 2.5.1), we will now present the segmentation pipeline.

- **Step 1: Intensity normalization**

    The intensity of the target volumes are normalized to the value that the user chose in the GUI. The normalization works in two steps: the background is first detected in the histogram, using its low intensity and the significant number of pixels of which it is created. Then it estimates the mean values of the pixels, background excluded. Then this mean value is normalized the normalization value defined by the user via the GUI. Thus, atlas and target images have the same mean intensity (background expected). This normalization is useful if we want to use the command-line interface (i.e. if we want to run a lot of segmentations without the GUI). We just choose one atlas and all the volumes to be segmented. By normalization, the mean intensity of the target images to be segmented will be the same as in the atlas. This step is usefull because now all the target images should have the same initial parameters.

- **Step 2: Image registration**

    In order for the atlas to guide the segmentation (spatial information), it has to be aligned to the target images. The transformation between the atlas and the first related target volume is evaluated during this step, using the registration method defined by the user in the GUI.

- **Step 3: Spatial prior alignement**

    The transformation computed during the image registration step (step 2) is applied to all the structure-specific atlases. We finally obtain new atlases which are aligned to the images to be segmented.

- **Step 4: EM Algorithm in the tree structure**

    Now, all the required initializations and preprocessing steps have been done. The whole segmentation workflow is then applied (section 2.4.5). A typical multichannel segmentation, using two $200x300x300$ targets volumes (T1 and T2 MR images) and segmenting five tissues, lasts around twenty minutes.

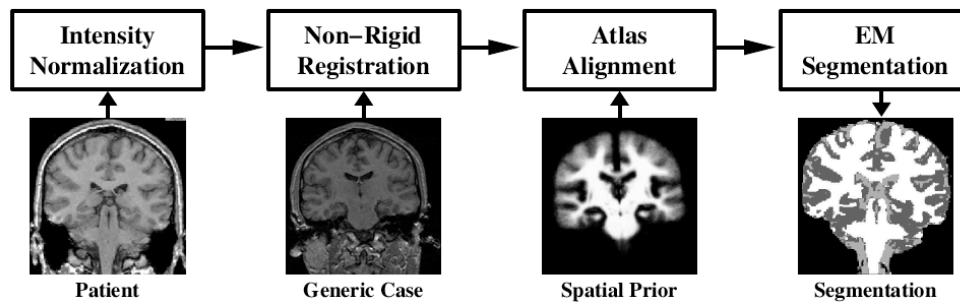    The whole algorithm pipeline in summarized in Figure (2.4).

Figure 2.4: Algorithm segmentation pipeline in Slicer 3
Courtesy Pohl et al. [18]. The first step of the segmentation process consists of intensity normalization. The second consists of an estimation of the transformation to apply for the registration. Third the atlases are aligned using the transformation. Finally, the EM segmentation algorithm is performed.

### 2.5.3   Summary

In Slicer 3, the whole segmentation pipeline can be described as in Figure 2.5. There is first an initialization step, done by user via the GUI. Then, some pre-processing steps are applied in order to enhance the segmentation. Finally runs the EM pipeline to segment the MR images.

## 2.6   Limitations

Even if the EM segmentation pipeline is robust in Slicer 3, some limitations appear.

The first problem the user has to face appears during the intensity normalization step. Because of the lack of vizual feedback, it is challenging for the user to find the good normalization value. This problem will be discussed in section 3.4.

The second problem is directly linked to the EM algorithm. As the maximization method is a local one, the class distribution, which are used for the initialization of the algorithm has to be well defined. So far, we have no possibilities to know how accurate our definition of the class is. This problem will be discussed in details in section 3.2.

Another problem appears at the same steps. The actual method for defining means and variances for each class appears not as efficient as we want it to be. This problem will be discussed in details in section 3.1.

Moreover, if the user is not familiar with the EM segment algorithm, defining good hierarchical parameters can be challenging. In section 3.5 we will provide a tool to help the user.

The last problem we encounter appears after the initialization, during the preprocessing. Only one preprocessing step (intensity normalization) is done on the target images before the atlases are registered to it. Since the EM algorithm is targeted to work on MR images, bias field is a recurrent problem. Performing the registration without bias correction might affect the results of the registration. This problem will be discussed in details in section 3.3.

In this chapter, we have described the EM segmnetation algorithm and its implementation in Slicer 3. We have concluded this the limitations of the current implementation. In the next chapter we will propose some solutions and describe them in details.
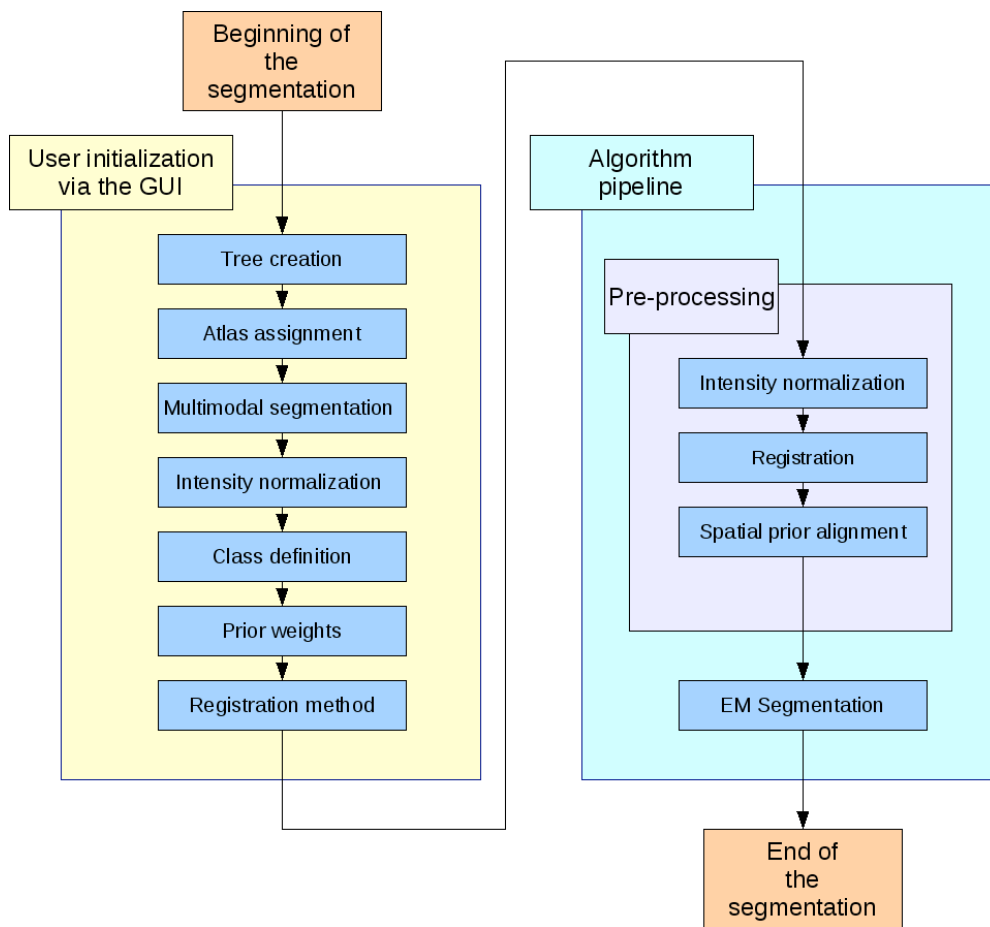
Figure 2.5: Flowchart of EM segmentation pipeline in Slicer 3

# Chapter 3

# Contributions

In this chapter we will present all the contribution realized to enhance the segmentation workflow in Slicer 3. We propose solutions to enhance the class selection method and to allow the user to evaluate the selection. We will address the registration problem in a third step. Finally, we present the tools that we have added to help the user to find the optimum intensity normalization value and to estimate the hierarchical parameters.

## 3.1    Class Distribution Selection

During parameters initialization, the user has to define each class distribution. The previous method of selection presents some limitations and we propose a new approach.

### 3.1.1    Motivation

So far, the user has two choices to define each class distribution.

The first possiblity consists of entering manually the intensities mean value and variance for each class, for each volume to be processed. This way, the user is precise to define each class. However it is challenging to find the optimum mean value and variance for each class for each volume. Morever, each time the user wants to process a new volumen, mean values and variances have to be redefined. It is not convenient and it can be time-consuming to find accurate values for the parameter initialization.

The next approach consists of defining a class model by manual sampling. For each class, the user clicks in the related part of the volume. The problem with this method is that the computation of the mean value and variance only uses a few samples. Because manual sampling is time consuming, the user will only select a reduced number of sample compared to the total number of voxels in the volume. Consequently mean values and variances are not accurate. Besides, reproducibility is an issue. Indeed it is challenging for an user to perform an identical sampling on two volumes.

Because of all these limitations, we proposed a new approach using a labelmap, to estimate each class model.

### 3.1.2    Solution

Our solution is to create a labelmap where each tissue class is given a label and a color. The relation color/class is stored in parameter $H$, in the EM algorithm. This relation color/class is set up during the tree creation step (section 2.5.1).

The user creates a labelmap by coloring characterisc regions for each tissue to segment in the appropriate color. This gives a spatial information to the algorithm. The labelmap allows to compute automatically the mean value and covariance of each class, for each tissue. In the next section we explain why we compute the covariance matrix instead of the variance.
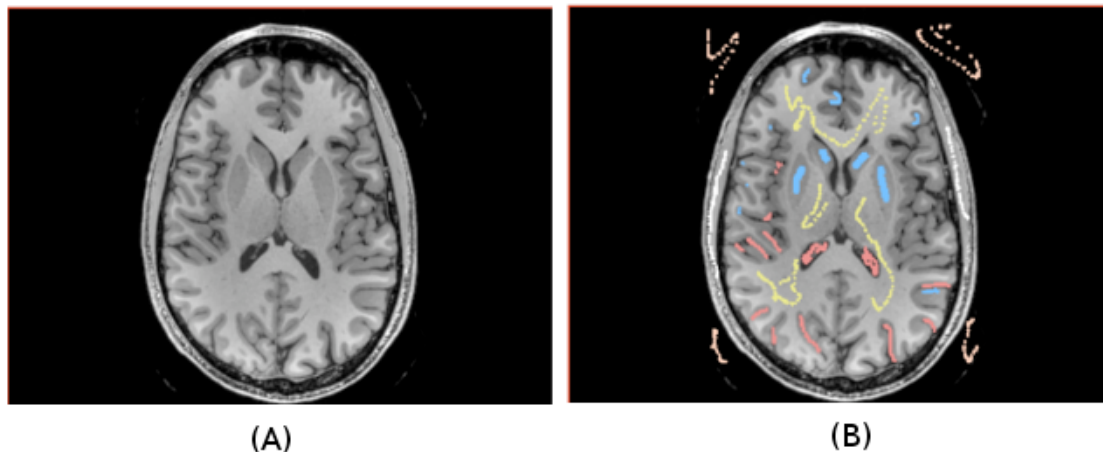
Figure 3.1: Axial view of the labelmap
Left image (A) represents an axial view of the T1 volume to be segmented. Right image (B) represents the volume overlays by the labelmap. Pink represents the air. White represents the skull. Yellow represents the white matter (WM). Blue represents the grey matter (GM). Red represents the cerebrospnial fluid(CSF).

The labelmap sampling is a convenient solution because a large number of sample can be drawn for each class. Besides the sampling is now reproducible since we can store then re-use it later.

Figure (3.1) illustrates an example of a labelmap. Left image (A) represents an axial view of the T1 volume to be segmented and right image (B) the labelmap created for this volume. Each color represents a tissue to be segmented.

### 3.1.3   Evaluation

To estimate the importance of this contribution, we perform a simple comparison. We select manually ten points of the white matter as accurately as possible. With this manual sampling method, we estimate mean value and covariance matrix, within two volumes. The second step of the evaluation consists of estimating the same values for the same tissue using a labelmap (more than 200 sampling points).

**Results:**

$$\mu_{Manual1} = 543 \quad, \quad \mu_{Manual2} = 93 \quad \text{and} \quad \mathbf{\Sigma_{Manual}} = \begin{bmatrix} 1105 & -25 \\ -25 & 1308 \end{bmatrix}$$

$$\mu_{Label1} = 489 \quad, \quad \mu_{Label2} = 92 \quad \text{and} \quad \mathbf{\Sigma_{Label}} = \begin{bmatrix} 592 & -201 \\ -201 & 280 \end{bmatrix}$$

The mean values ($\mu_{Manual}$ and $\mu_{Label}$) differ slightly and covariance matrices ($\Sigma_{Manual}$ and $\Sigma_{Label}$) differ significantly. It means that our approach is useful and it shows the importance of having a large sample to evaluate means and covariance values. The square variance of the class for the first volume, is in position $(1, 1)$ in the covariance matrix. For the second volume, it is in position $(2, 2)$. Variance expresses the range through which the class is expected to be. The estimation of this interval is significantly more accurate with labelmap sampling than manual sampling. We can explain it with the important number of samples used for the estimation.

## 3.2   Class Distribution visualization

An important contribution is a tool which allows to visualize the distribution of the classes to be segmented.

### 3.2.1   Motivation

As discussed before, the algorithm is sensible to the initalization. In other words, the quality of the initialization is important. Once the parameters are chosen, the user has no tools to know if his selection is accurate. Two classes to segment can not have too close means and variances and even if the user sees the values obtained after sampling, it is challenging to know if two classes to be segmented are too similar or not.

### 3.2.2   Solution

The objective is to provide the user the most accurate and usefull vizualisation tool as possible.

We first assume that each class has a normal distribution. This assumption is correct since the EM algorithm present in Slicer 3 is based on Gaussian mixture models. We decided to plot the Gaussian in the 3D space, using the multivariate normal distribution. In the 2D non-singular case, the probability density function is

$$f(x,y) = \frac{1}{2\pi\sigma_y\sigma_y\sqrt{1-\rho^2}}\exp\left(-\frac{1}{2(1-\rho^2)}\left(\frac{(x-\mu_x)^2}{\sigma_x^2}+\frac{(y-\mu_y)^2}{\sigma_y^2}-\frac{2\rho(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y}\right)\right) \quad (3.1)$$

(see [15]). $x$ and $y$ are the positions of the pixel in the plane. $f(x,y)$ will return the value (height) of the $(x,y)$ pixel. The $X$ axis represents a volume while the $Y$ axis represents another volume. Let's first say that the range of the $X$ and $Y$ axes are the intensity range of the $X$ and $Y$ volumes. $\mu_x$ is the mean value of the class in the $X$ volume. $\mu_y$ is the mean value of the class in the $Y$ volume. $\sigma_x$ and $\sigma_y$ are the variance of the tissue in its respective volume. $\rho$ is the correlation between $X$ and $Y$. It indicates the strength and direction of a linear relationship between two random variables ( see [16]).

We can easily deduce $\rho$ from the covariance matrix $\Sigma$ (see [17]). Indeed, in the 2 dimensional case, the covariance matrix can be expressed as

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}$$

The covariance matrix and the mean values for each class to segment for each image are computed during the labelmap sampling (section 3.1).

Please note that if $\rho = 1$, it means that the two radom variables are the same and we can not use the same density probability function anymore. We express the classic normal distribution

$$f(x,y) = A\exp-\left(\frac{x-\mu_x}{2\sigma_X^2}+\frac{y-\mu_y}{2\sigma_Y^2}\right) \quad (3.2)$$

$A$ is the amplitude of the Gaussian.

We said that the range of the $X$ and $Y$ axes are the intensity range of the $X$ and $Y$ volumes. The problem with this approach appears if the classes to segment are not spread over all the intensities. Indeed, the vizualisation is poor then: the Gaussian is only localized in a small portion of the 2D plane. We want to "zoom" on the region of interest. We decided to change the range of the two axes. Then range will be redefined for each image. Let's present it for a given image $X$. The range is now defined as the difference between $Max$, the maximum value extracted with the labelmap, between all the samples between all the classes for the image $X$, and $Min$, its opposite.
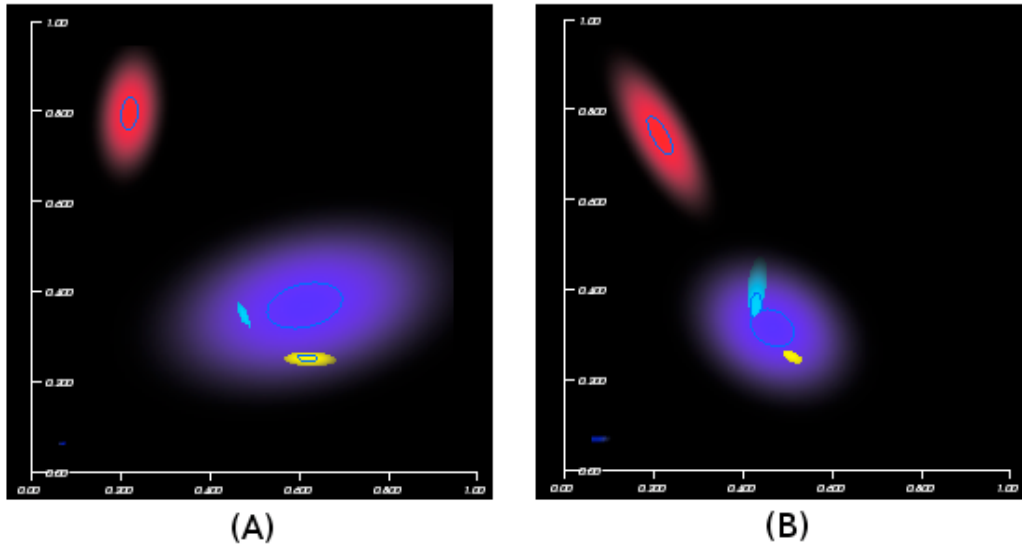
Figure 3.2: Distribution of the class to be segmented in an homogeneous volume
Left image (A) represents the distribution of the class after a manual sampling. Right (B) image
represents the class distribution after a labelmap sampling. Air is plotted in dark blue, skull in
blue, WM in yellow, GM in green and CSF in red. X axis represents a T1 volume and Y axis
represents a T2 volume.

For our particular purpose of tissues distributions vizualisation we did not use exactly the formulas (3.1) and (3.2). Indeed, we do not normalize the curves: we set $\frac{1}{2\pi\sigma_y\sigma_y\sqrt{1-\rho^2}}$ to 1 in Equations 3.1 and $A$ to 1 in Equation 3.2. We do it because we do not know the importance of each class so we do not want to "disavantage" any one in the visualization. A compromise could be to have an amplitude factor proportional to the number of pixels which constitute the class.

### 3.2.3   Evaluation

We finally obtain the results Figure 3.2 for different sampling methods. The left Figure (A) illustrates the visualization we obtain after a manual sampling and (B) the results after a labelmap sampling. The distributions slightly differ. The dark blue point, on the left corner represents the air. The skull is represented in purple, the white matter in yellow, the gray matter in green and the cerebrospinal fluid (CSF) in red. The X axis represents the T1 volume and the Y axis the T2 volume. In left image (A), the skull is significantly more spread than in right image (B), especially along the X axis (T1). According to a clinician expert, the labelmap sampling offers better results. Indeed, the skull should almost have the same variance in the T1 and T2 images, which is illustrated in Figure (B). According to the same expert, the intensities of CSF and grey matter, in the T1 volume can not be as distant as these are in image (A). The distributions of these two tissues should be close, as it is in right image (B). These two observations validates the utility of the labelmap sampling in order to have accurate tissue distributions.

## 3.3   MRI Bias Field Correction

The registration step could present some problems if the image to segment exhibits intensity inhomogineity. Moreover, bias field can also be a problem regarding the accuracy of the class distributions. We will first remind the problems and then propose some solutions.

### 3.3.1   Motivation

In the segmentation process, a registration step is required. Registration consists of finding a transformation to fit two images as well as possible. The main methods are described in [21]. Only one pre-processing (intensity normalization) is done before the registration. The problem is that the algorithm is designed to process MR images. MR images are often corrupted by a bias field. Thus, the image to register exhibits intensity inhomogeneity. These inhomogeinities significantly affect the registration.

On Figure 3.3, we present the result of the registration between an atlas (left image (A)) and a biased MR image (right image (B)). Note that the target MR image has been normalized to have the same mean value as the atlas. The image (C) shows that the registration is poor.
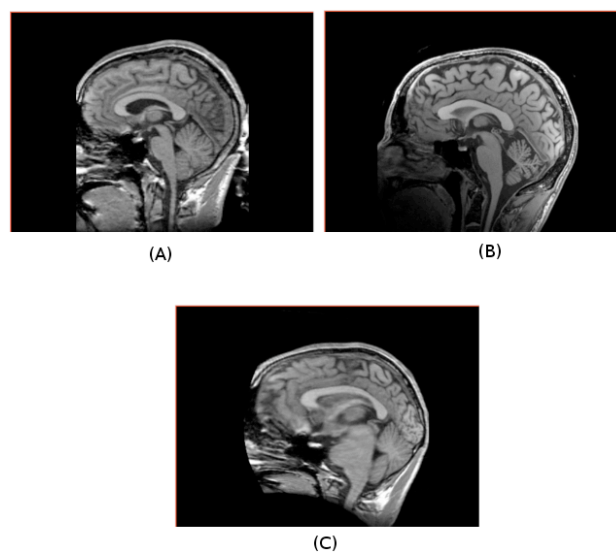


Figure 3.3: Registration of a MR image which exhibits intensity inhomogeneity
Image (A) is the atlas to be registered. Image (B) is the target of the registration. Image (C) is the image (A) after registration to image (B)

Moreover this bias field leads to another important issue. Indeed, if the MR image intensities are biased, the intensity of a given tissue varies significantly in the volume, even if it should not. Then the algorithm will be initialized with wrong mean and variance values and the segmentation will not be as good as it should be.

### 3.3.2   Solution

The idea simply consists of correcting the bias field of the MR image before the intensity normalization step. Thus, the registration and the class distributions will be significantly enhanced. Since the registrationand class distributions is better, it should also improve the segmentation.

To correct the bias field, we used the non-parametric approach presented by Sled *et al* in [19]. We choose a non-parametric approach because it does not require prior information like the number of tissues to correct or the mean value of each tissue to be corrected. We implement an ITK filter [13] and [14] in Slicer 3.

We can describe the new segmentation workflow in Slicer 3 as we do in Figure 3.4.

We choose not to implement it in Slicer 3 as part as the EM Segment module. Indeed, users may want to correct the bias field in MR images for other purposes. Moreover, it is possible because it would be the first pre-processing step. The user will first have to correct the intensity inhomogineity via the module developped for this particular purpose then use the corrected images in the EM segmentation module.
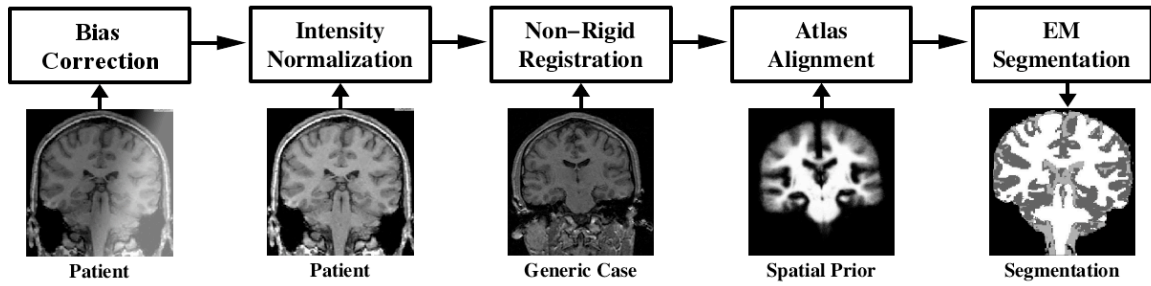
Figure 3.4: New algorithm segmentation pipeline in Slicer 3
First the intensity inhomogeneity are corrected. Second the intensity is normalized. Third an estimation of the transformation to apply for the registration is done. Fouth the atlases are aligned using the transformation. Finally, the EM segmentation algorithm is performed.
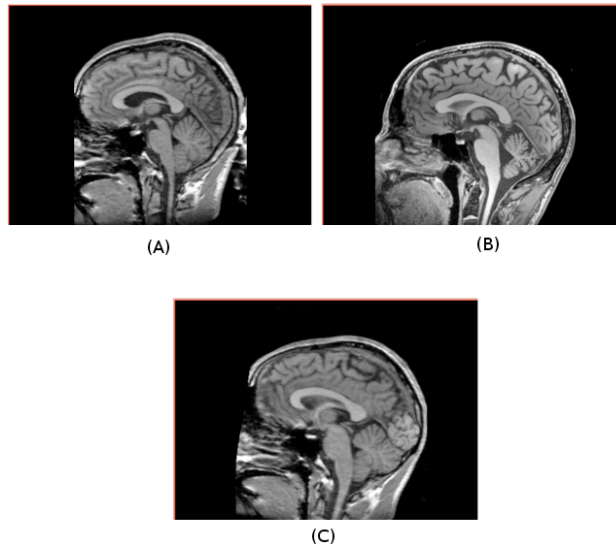


Figure 3.5: Registration of a MR image without intensity inhomogeneity
Image (A) is the atlas to be registered. Image (B) is the target of the registration. Image (C) is the image (A) after registration to image (B)

After the bias correction, we obtain interesting results presented in Figure3.5. (A) represents the atlas to be registered. (B) represents the target volume for the registration. (C) represents the atlas after the registration. The result of the registration visually appears to be better but we can not be satisfied of this visual assessment. We need a qualitative evaluation method.

### 3.3.3  Evaluation

We evaluated accuracy of the registration using the joint histograms method. The joint histogram evaluation method is basic comparison between two images. Let $A$ be a matrix of size $WxL$. $W$ will be the intensity range of the first image used for the comparison. $L$ will be the intensity range of the second image to be compared. The matrix is initialized to zero. Each time that in the same position, there is the same intensities in the two images, we add one in the corresponding cell in the matrix. Thus, a perfect registration, would lead to an array of zeros, expect on the diagonal. After the joint histogram creation, the value at the coordinate $i, j$ in the matrix is the number of pixel pairs having intensity $i$ in image onee and intensity $j$ in image two, at the
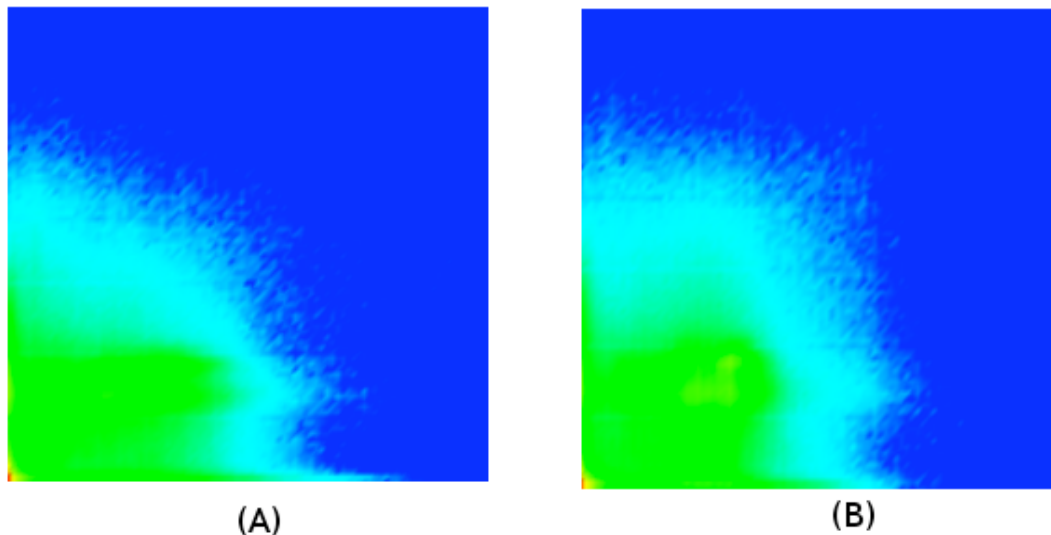
Figure 3.6: Joint histograms to evaluate the registration of the inhomogeneous MR image
Figure (A) represents the joint histogram of the inhomogeneous MR image and its atlas before
registration. Figure (B) represents the joint histogram of the inhomogeneous MR image and its
atlas after registration.

same position in the volume.

In Figure 3.6, the left image (A) and the right image (B) compares the joint-histogram of the
biased image and its atlas, respectively before and after registration. The color scale used is the
following one: if there are a lot of pixels in a cell of the array, the cell will be displayed in red. We
the number of points decreases continously. The color scale goes from red to blue. The color fo
medium intensities is yellow.

In Figure 3.7, left image (A) and right image (B) compare the joint-histograms of the image
after a bias field correction and its atlas, before and after registration.

It appears that the difference between before and after registration is more significant if the bias
field is corrected. Indeed, we compare the joint histograms for volumes which exhibits intensity
inhomogeneity: there is no significative difference between before and after registration. That
means that the registration did not improve the similarity between the images. On the contrary,
if we compare joint histogram for an intensity inhomogeneity free vollume, the number of points
around the matrix diagonal increases. It means the images are more similiar after than before
registration. It shows the utility of the contribution. The influence of these results after the whole
segmentation process will be presented in the next chapter.

Regarding the effect of the intensity inhomogeneity on tissue intensity distribution, we will
present results using the tool we developped in the previous section, to evaluate the classe distri-
bution. Using the same labelmap for sampling, we obtain two significantly different distributions
for a image corrupted with a bias field and the same image corrected. Figure 3.8 shows the two dif-
ferent distributions. The relation tissue/color is the same as the one in Figure 3.2. The left image
(A) presents the distribution before bias field correction. The right (B) presents the distribution
in the corrected volume.

In the left image (A), the tissues variances are huge, especially for the CSF (red) in the T2
volume. According to an expert, it is not a good assumption. CSF should on appear in the high
intensities of the T2 volume, as presented in image (B). Image (A) means that the CSF class
contents pixels of high and low intensities in the T2 volume. It is a bad assumption and we can
explain it because of the intensity inhomogeneity.

Intuitively, we understand that the segmentation process should be a lot affected because of the
two issues we presented (registration and distribution). Concrete results on the whole segmentation
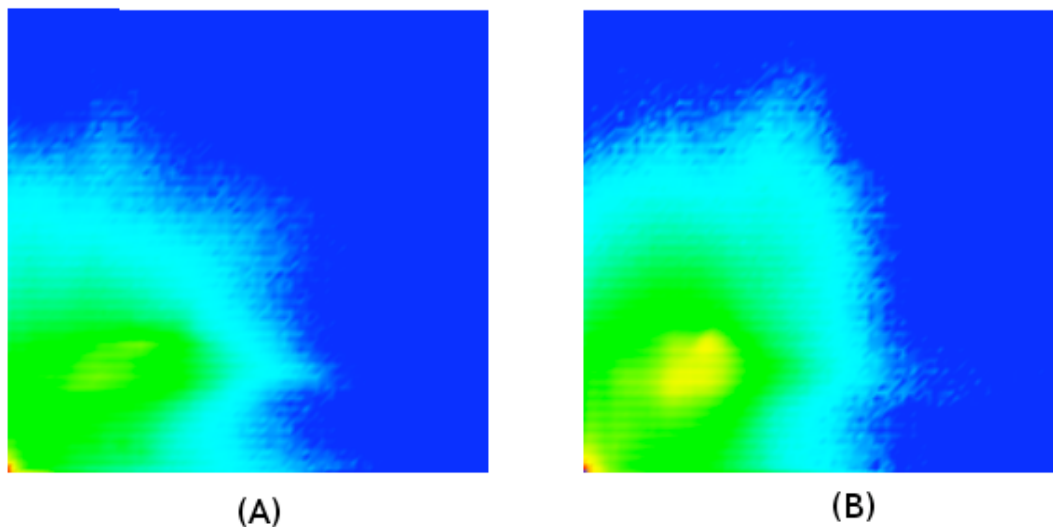
Figure 3.7: Joint histograms to evaluate the registration of the homogeneous MR image
Figure (A) represents the joint histogram of the homogeneous MR image and its atlas before registration. Figure (B) represents the joint histogram of the homogeneous MR image and its atlas after registration.
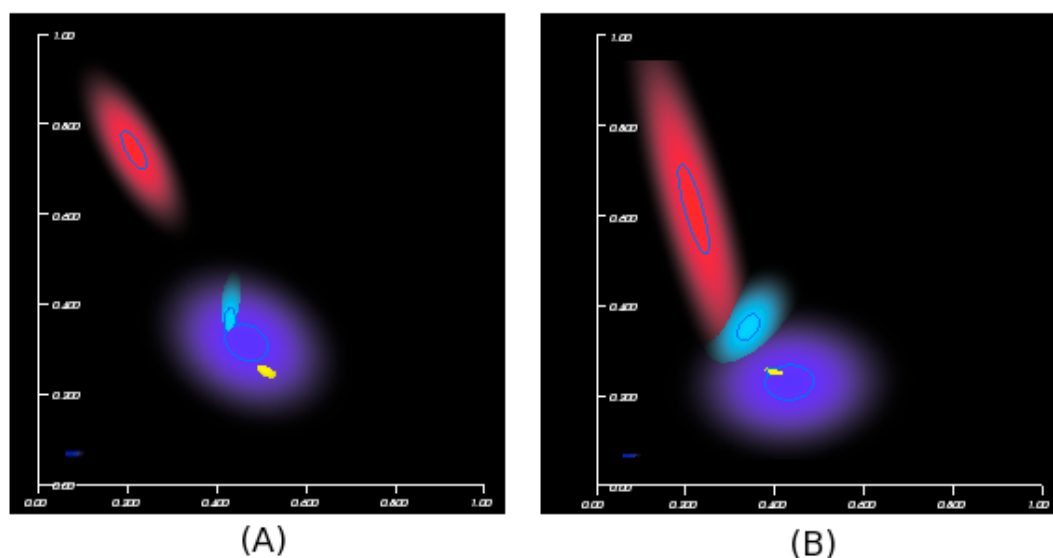


Figure 3.8: Distribution of the class to be segmented in an inhomogeneous volume
Left image (A) represents the class distribution in an homogeneous volume. Right image (B)represents the distribution of the class in a volume which exhibits intensity inhomogeneity. Air is plotted in dark blue, skull in blue, WM in yellow, GM in green and CSF in red. X axis represents a T1 volume and Y axis represents a T2 volume.

process will be presented in the next chapter.

### 3.3.4   Registration parameters

Even if we are performing a non-parametric registration, some parameters have to be defined. "Non-parametric" means no information about the volume and classes to correct. We will first present and explain the parameters. In a second step, we will propose some parameters adjusted to different problems.

- **Shrink factor**

    It is a factor which is used to reduce the size of the image to be processed. A down-sampling is done by the bias correction filter.

- **Maximum number of iterations**

    Optimization of the bias field occurs iteratively until the number of iterations exceeds the maximum specified by this variable.

- **Bias field full width at maximum iteration**

    The bias field is modelled with a Gaussian. This variable characterizes this Gaussian (see [20]) and can be presented as a parameter which defines the strength of the bias.

From the understanding of the parameters, it makes sense that if the time of processing has tp be reduced, the shrink factor must be increased and the maximum number of iteration must be reduced. The limitation is that it can deteriore the bias field correction.

The bias field full width at maximum iteration (BFFWMI) can also be reduced or increased, depending on the importance of the bias. If the bias field deteriorates significantly the image, an important BFFWMI must be used. On the contrary, if the intensity inhomogeneities are not significant, BFFWMI can be small.

## 3.4   Intensity Normalization

Another useful contribution is a tool which helps the user to determine the good normalization value.

### 3.4.1   Motivation

As discussed in section (2.5.1), at the step 4, an intensity normalization is done. We have already presented the utility of an intensity normalization in the same section. The problem is that the user has no tool to find the optimum values for the segmentation. The user has to find the mean intensity of the voxels in the MR image, excluding the background. In practise it is challenging to estimate the good value since no tool exist for this particular purpose.

### 3.4.2   Solution

We implemented a tool, to allow the user to find easily and accurately this normalization value.

The first step of the work consited in creating the histogram corresponding to the image. The $Y$ axis which presents the number of pixels for each intensity in the volume uses a log-scale because the range is large. The log scale significantly reduces the range. We then added a cursor in this histogram. With this feature, the user can choose the intensity which will be the threshold between the background and structure of interest. Finally, while the cursor is moving, our algorithm computes automatically the mean value of the voxels in the volume, from this intensity range to the highest intensity range. This is the normalization value.
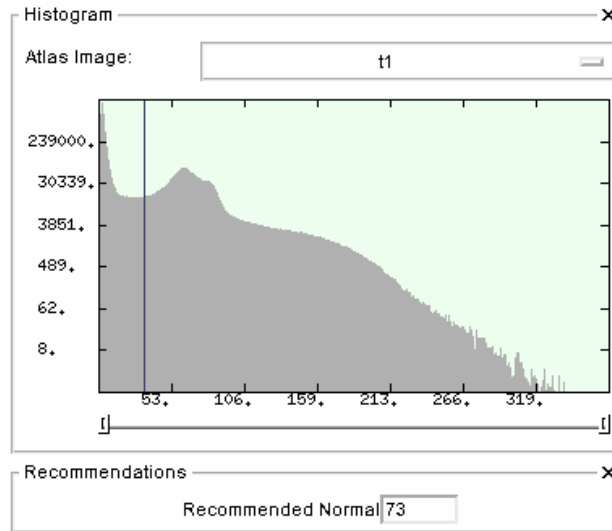
Figure 3.9: Tool developped for the intensity normalization parameter estimation
The user moves the cursor and get an immediate feedback about the normalization value to use
in the "recommended normal" box

We present in Figure 3.9 the tool we developped. A T1 volume has been loaded. The user can move the cursor in the histogram. While moving, it returns in real-time the normalization value for the given position of the cursor in the lower frame.

## 3.5   Global Prior Weights Estimation

The last contribution to the EM segmentation is a tool which provides the user an easy and fast way to estimate the approximate size (also called Global Prior Weights) of each class to be segmented.

### 3.5.1   Motivation

During the segmentation process,at the $6^{th}$ step of the intialization (section 2.5.1), the user has to provide to the algorithm an estimation of the size of each tissue to be segmented. First of all if there are a lot of structures to segment, this step is time consuming. Moreover, the user may not know at all which size to choose. A tool to estimate of the good sizes to choose is needed.

### 3.5.2   Solution

Our solution is to divide the problem in two parts. We provide the user with a real-time feedback regarding the global prior weights estimation. The second part consists of developping an algorithm which fills automatically the tree.
The new tool is presented in Figure 3.10, in the left image (A). When the user moves a cursor, it changes the intensity range for each class. The tool provides a real time feedback to the user. The user sees Figure 3.10) (B) which is updated in real time, regarding the position of all the cursors consequence. Clicking on "update tree" (in Figure 3.10) (A)), the estimated size of a class is computed and the information is stored into the tree structure (parameter $H$).
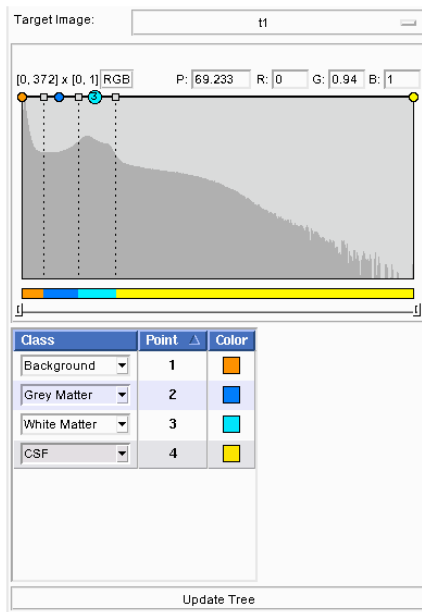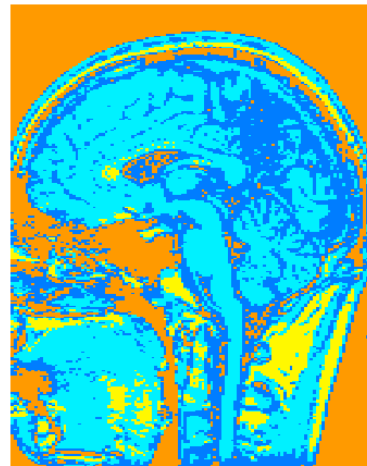
Figure 3.10: Tool developped for the global prior weights estimation
Left image (A) presents the tool we developped for the global prior weights estimation. Right image (B) shows the visual feedback provided to the user given the tool presented in (A)

In this chapter we have presented our contribution to the EM segmentation algorithm. The illustrations shows that the selection of the parameter has been improved: in the next chapter we present the impact on the segmentation.

# Chapter 4

# Results and discussion

We will now discuss the impact of the enhanced usability on the segmentation results. Firstly we present the influence of each contribution on the final segmentation. Secondly suggest some potential improvements for the EM segment module in Slicer.

## 4.1   Results

Here we start with a presentation of different results, using the different contributions. The results obtained will then be reviewed by a expert radiologist. The expert who accepted to evaluate the results is Dr. KIKINIS, professor of radiology at Harvard Medical School. We report his assessments for each segmentation in the "expert's validation" section. We work with the same dataset for all the segmentation. We chose the datasets with significant intensity inhomogeneity in order to show the importance of our contributions.

### 4.1.1   Original segmentation

First we present the segmentation obtained without any contribution. Based on this segmentation, the importance of the contributions will be presented in the next sections. First we describe the testing process. Second we present the results of the segmentation. Then the clinical expert's assessment will be reported. Finally, we discuss about the expert's observations.

#### Method

Thanks to Sonia Pujol, a tutorial for the EM segmentation in Slicer 3 is available at `http://www.na-mic.org/Wiki/images/2/2f/AutomaticSegmentation_SoniaPujol_Munich2008.ppt`. We follow it with an image which exhibits intensity inhomogeneity. The atlases we use for the segmentation are the one available at `http://www.na-mic.org/Wiki/images/b/b7/AutomaticSegmentation.tar.gz`.

#### Results

Figure 4.1 shows the result of the segmentation. The left image (A) presents an axial view of the T1 target volume to be segmented. The right image (B) presents and axial view of the segmentation's results.

#### Expert's validation

Based on the assessment of the expert, the grey matter is significantly over estimated in the whole volume. The segmentation is poor and can not be use for any medical application purpose.
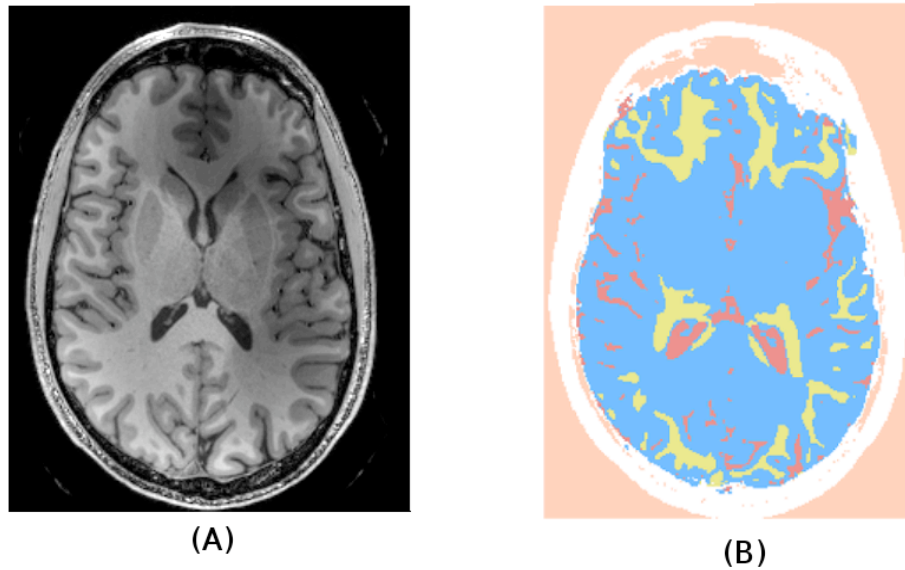
Figure 4.1: Results of the segmentation without bias field correction
Left figure (A) is an axial view of the T1 the target volume to be segmented. Right figure (B) is an axial view of the segmentation's result. Air is represented in pink, skull in white, white matter in yellow, grey matter in blue and cerebrospinal fluid in red.

### Discussion

We speculate that the reasons of the poor segmentation are the incorrect registration and the class ditributions which is poorly defined. It leads to inaccurate means and excessively large variances which deteriorate a lot the segmentation. This is also illustrated Figure 3.8 where the right image shows that grey matter (light blue) is overestimated in the right image (B) in comparison to the left image (A).

### 4.1.2   Bias corrected segmentation

Here we present the results of a segmentation using the intensity inhomogeinity correction tool developped. First we describe the testing process. Second we present the results of the segmentation. Then the clinical expert's assessment will be reported. Finally, we discuss about the expert's observations.

### Method

Similarly to the previous section, we follow the EM segmentation tutorial with one extra-step: the bias field correction.

### Results

The results of the final segmentation are presented in Figure 4.2. The left image (A) presents an axial view of the T1 target volume to be segmented. The right image (B) presents and axial view of the segmentation's results.

### Expert's validation

The expert pointed misegmentation in the skull area. Indeed, some voxels in the skull are incorrectly classified as grey matter and white matter. The expert explained that this misclassification is due to partial volume effect. The partial volume effect is caused by imaging voxels containing
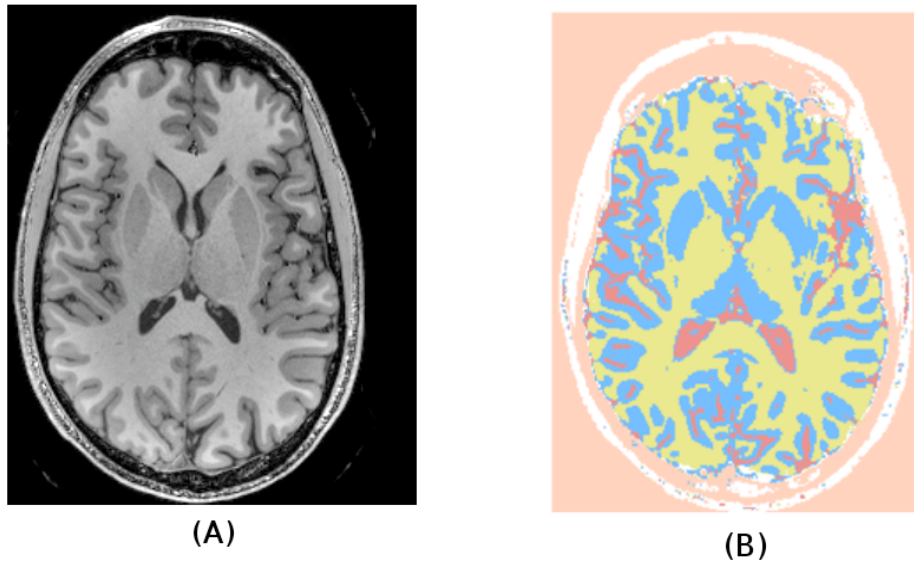
Figure 4.2: Results of the segmentation with bias correction
Left figure (A) is an axial view of the T1 the target volume to be segmented. Right figure (B) is an axial view of the segmentation's result. Air is represented in pink, skull in white, white matter in yellow, grey matter in blue and cerebrospinal fluid in red.

two different tissues (skull and air in our case). The mesured intensity for that voxel will be an average of the two tissues at that location. Another misclassification occurs in areas where the skull is classified as air. The reason is that in some regions, the skull is porous and contains air. Even is pores are not significant, it leads to partial volume effect the misclassification.

**Discussion**

Based on our understanding of the algorithm this better segmentation can be explained by a better registration as shown section (3.3). Moroever, as demonstrated it in Figure 3.8, the tissues have a better distribution, especially the grey matter (light blue) in this case. The misclassifications in the skull could be corrected by a pre-processing step using skull stripping algorithm.

**4.1.3   Label map sampling segmentation**

This segmentation will now let us evaluate the influence of the sampling method on the segmentation. First we describe the testing process. Second we present the results of the segmentation. Then the clinical expert's assessment will be reported. Finally, we discuss about the expert's observations.

**Method**

Similarly to the previous section, we follow the EM segmentation tutorial . The extra step that we take are the bias field correction and the initialization of the class distribution using a labelmap sampling.

**Results**

The results of the final segmentation are presented in Figure B.3. The left image (A) presents an axial view of the T1 target volume to be segmented. The right image (B) presents and axial view of the segmentation's results.
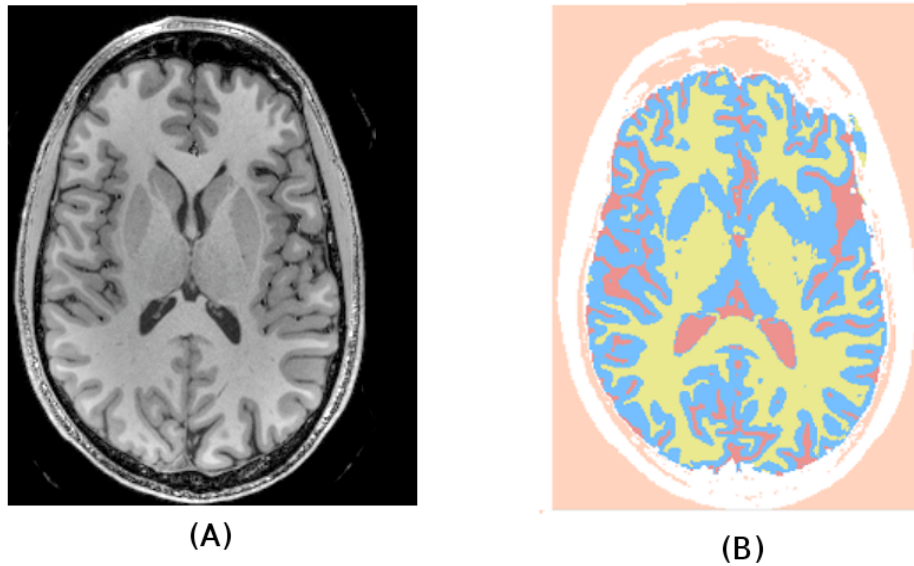
Figure 4.3: Results of the segmentation with labelmap sampling
Left figure (A) is an axial view of the T1 the target volume to be segmented. Right figure (B) is an axial view of the segmentation's result. Air is represented in pink, skull in white, white matter in yellow, grey matter in blue and cerebrospinal fluid in red.

**Expert's validation**

The skull is better segmented than in the manual sampling segmentation. Moreover, there is less misclassification than before. Nevertheless, in cerebulum, the segmentation differs from the previous one. It is underestimated with the labelmap sampling whereas before it was over estimated. The cerebulum is a specific area of the brain because there is a lot of white and grey matter in a small area. The segmentation is challenging and similarly as the point raised during the validation of the bias field correction segmentation, the results are due to partial volume effects. According to our expert segmentation of this area is a complex task. A solution to improve the segmention the cerebrulum properly, would be to extract the region of interest and to apply a finer algorithm to this area.

**Discussion**

We explain the enhanced classification because the tissues have more located distributions than in the previous segmentation, as we se in Figure 3.2. More screenshots with a better resolution are presented in App. B to have a better understanding of the interest of of a manual sampling instead of a labelmap sampling.

## 4.2   Perspectives

Our work focused on improving the usability of EM segment and the selection of the optimum parameters. As a future direction, some effort could spend on the optimization of the algorithm. Indeed, the long processing time is a major drawback of the current implementation.

On one side the processing time both for the bias field correction and for the EM segmentation needs to be improved. A number of users reported this issue as the main limitation of the current implementation. We observe that the processing for 255x255x100 volume is approximatly 30 minutes. Towards this goal the code of the bias field correction algorithm needs to be optimized.

Another issue occurs with images where the distributions of two classes have similar means. In

this case, the exploration of the parameters is challenging, especially for the global prior weights. A useful contribution could be to represent each class as a gaussian instead of a finite portion of the histogram.

# Conclusion

We have presented an improved algorithm and implementation of the EM segmentation. Our main contributions are the addition of a bias field correction step and the creation of tools to select the optimum parameters. First we implemented a tool to apply the non parametric method for automatic correction of intensity uniformity in MR images of Sled et al. [19]. Our second contribution enables the visualization of the tissue classes as Gaussian distributions and the selection of the optimum statistical parameters.

We have reported our segmentation results and our qualitative validation performed by a clinical expert. The expert assessed our interface as a significant improvement in usability and validated our segmentation improved results.

This work opens new perspectives for improving the impact of medical imaging in a clinical environment. We suggest that the future effort should focus on reducing the processing time. The interaction with clinicians at Harvard Medical School make us believe that the outcome of this internship will be developped further and ultimatly make a difference for the patients.

# Bibliography

[1]   A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the em algorithm", *Journal of the Royal Statistical Society: Series B*, vol. 39, pp. 1-38, November 1977.

[2]   Y. Weiss, "Bayesian motion estimation and segmentation", *PhD thesis*, Massachussets Intitute of Technology, May 1998.

[3]   R.C. Hardie, K.J. Barnard, and E.E. Armstrong, "Joint MAP registration and high-resolution image estimation using a sequence undersampled images", *IEEE Transaction on Image Processing*, vol. 6, pp. 1621-1633, December 1997.

[4]   M. Murgasova, "Tutorial on Expectation-Maximization: Application to Segmentation of Brain MRI", May 2007.

[5]   S. Borman, "The Expectation Maximization Algorithm",January 2009.

[6]   Wikipedia, "Expectation-Maximization algorithm", `http://en.wikipedia.org/wiki/Expectation-maximization_algorithm`, June 2009.

[7]   G. McLachlan, and T. Krishnan, "The EM Algorithm and Extensions", *John Wiley & Sons*, New York, 1996.

[8]   K. V. Leemput, F. Maes, D. Vandermeulen et al., "Automated model-based tissue classification of MR images of the brain", *IEEE Transaction on Medical Imaging*, 18(10), pp. 857-908, 1999.

[9]   K. V. Leemput, F. Maes, D. Vandermeulen et al., "Automated model-based bias field correction of MR images of the brain", *IEEE Transaction on Medical Imaging* 18(10), pp. 885-896, 1999.

[10]  W. M. Wells III, W.E.L. Grimson, R. Kikinis et al., "Adaptative segmentation of MRI data", *IEEE Transaction on Medical Imaging* 15(4), pp. 429-442, 1996.

[11]  K.M. Pohl et al., "A Hierarchical Algorithm for MR Brain Image Parcellation", *IEEE Transaction on Medical Imaging* 26(9), pp. 1201-1212, 2007.

[12]  D. Hoa, and A. Micheau, "e-MRI, Magnetic Resonance Imaging physics and technique course on the web", *http://e-mri.org*, 2007.

[13]  Kitware, "Insight Toolkit", `http://www.itk.org/`, 2009.

[14]  N. Tustison, "Nick's N3 ITK Implementation For MRI Bias Field Correction", `http://www.insight-journal.org/browse/publication/640`, June 2009.

[15]  Wikipedia, "Multivariate Normal Distribution", `http://en.wikipedia.org/wiki/Multivariate_normal_distribution`, June 2009.

[16]     Wikipedia, "Correlation", `http://en.wikipedia.org/wiki/Correlation`, June 2009.

[17]     Wikipedia, "Covariance matrix", `http://en.wikipedia.org/wiki/Covariance$_$matrix`, June 2009.

[18]     K.M. Pohl et al., "Automatic Segmentation Using Non-Rigid Registration", *MICCAI*, 2005.

[19]     J.G. Sled, A.P. Zijdenbos, and A.C. Evans, "A non parametric method for automatic correction of intensity nonuniformity in mri data", *IEEE Transactions on Medical Imaging* 17(1), pp. 87-97, February 1998.

[20]     Wikipedia, "Full width at half maximum", `http://en.wikipedia.org/wiki/Full_width_at_half_maximum`, July 2009.

[21]     B. Zitova, and J. Flusser, "Image registration methods: a survey", *Image and vision computing* 21, pp. 977-1000, 2003.

[22]     S. Pieper,M. Halle, and R. Kikinis, "3D SLICER", *Proceedings of the 1st IEEE International Symposium on Biomedical Imaging* 1, pp. 632-635, 2004.

# Appendix A

# Statistics

Here we present the main formulas we used in this reports and some fundamentals about probabilities.

## A.1   Fundamentals

$P(A)$ is the probability that $A$ is realized.

$P(A|B)$ is the probability that $A$ is realized, knowing $B$.

$P(A, B)$ is the probability that $A$ and $B$ are realized at the same time.

$P(A|B) = \frac{P(A,B)}{P(B)}$

## A.2   Bayes' theorem

### A.2.1   Theorem

Let $S$ be a sample of space. If $A_1, A_2, ...A_n$ are mutually exclusive and exhaustive events such as $P(A_i) \neq 0$ for all $i$. Then for any event $A$ which is a subset of $S = A1 \cup A_2 \cup ... \cup A_n$ and $P(A) > 0$ we have

$$P(A_i|A) = \frac{P(A_i)P(A|A_i)}{\sum_{j=1}^{n} P(A_j)P(A|A_j)}$$

### A.2.2   Proof

We have $S = A1 \cup A_2 \cup ... \cup A_n$ and $A_i \cap A_j = \varnothing$ for $i \neq j$. Since $A \subseteq S$

$$\begin{aligned} \Rightarrow A &= A \cap S \\ &= A \cap (A1 \cup A_2 \cup ... \cup A_n) \\ &= (A \cap A_1) \cup (A \cap A_2) \cup ... \cup (A \cap A_n) \end{aligned}$$

Moreover

$$P(A \cap A_i) = P(A)P(A_i|A)$$

So

$$\begin{aligned} P(A) &= P(A \cap A_1) + P(A \cap A_2) + ... + P(A \cap A_n) \\ &= P(A)P(A_1|A) + P(A)P(A_2|A) + ... + P(A)P(A_n|A) \end{aligned}$$

And

$$P(A|A_i) = \frac{P(A \cap A_i)}{P(A)}$$

Finally we obtain

$$P(A_i|A) = \frac{P(A_i)P(A|A_i)}{\sum_{j=1}^{n} P(A_j)P(A|A_j)}$$

## A.3   Jensen's inequality

### A.3.1   Inequality

Let $f$ be a convex function defined on an interval $I$. If $x_1, x_2, ..., x_n \in I$ and $\lambda_1, \lambda_2, ..., \lambda_n \geq 0$ with $\sum_{i=1}^{n} \lambda_i = 1$,

$$f(\sum_{i=1}^{n} \lambda_i x_i) \leq \sum_{i=1}^{n} \lambda_i f(x_i)$$

### A.3.2   Proof

To show that the theorem is true we proceed by induction.

- **Initialization**
  This is trivial for $n = 1$.

- **Hypothesis at rank $n$**

$$f(\sum_{i=1}^{n} \lambda_i x_i \leq \sum_{i=1}^{n} \lambda_i f(x_i))$$

- **Demonstration at rank $n + 1$**

$$f(\sum_{i=1}^{n+1} \lambda_i x_i) = f(\lambda_{n+1} x_{n+1} + \sum_{i=1}^{n} \lambda_i x_i)$$

$$= f(\lambda_{n+1} x_{n+1} + (1 - \lambda_{n+1}) \frac{1}{1 - \lambda_{n+1}} \sum_{i=1}^{n} \lambda_i x_i)$$

$$\leq \lambda_{n+1} f(x_{n+1}) + (1 - \lambda_{n+1}) f(\frac{1}{1 - \lambda_{n+1}} \sum_{i=1}^{n} \lambda_i x_i)$$

$$= \lambda_{n+1} f(x_{n+1}) + (1 - \lambda_{n+1}) f(\sum_{i=1}^{n} \frac{\lambda_i}{1 - \lambda_{n+1}} x_i)$$

$$\leq \lambda_{n+1} f(x_{n+1}) + (1 - \lambda_{n+1}) \sum_{i=1}^{n} \frac{\lambda_i}{1 - \lambda_{n+1}} f(x_i)$$

$$= \lambda_{n+1} f(x_{n+1}) + \sum_{i=1}^{n} \lambda_i f(x_i)$$

$$= \sum_{i=1}^{n+1} \lambda_i f(x_i)$$

With a concave function (in opposition to convex), the inequality becomes:

$$f(\sum_{i=1}^{n} \lambda_i x_i) \geq \sum_{i=1}^{n} \lambda_i f(x_i)$$

# Appendix B

# Results of the segmentation

Here we present the results of segmentations in the case of manual and labelmap sampling. The left image represents the results of the segmentation after a manual sampling and the other image represents the results after a labelmap sampling.
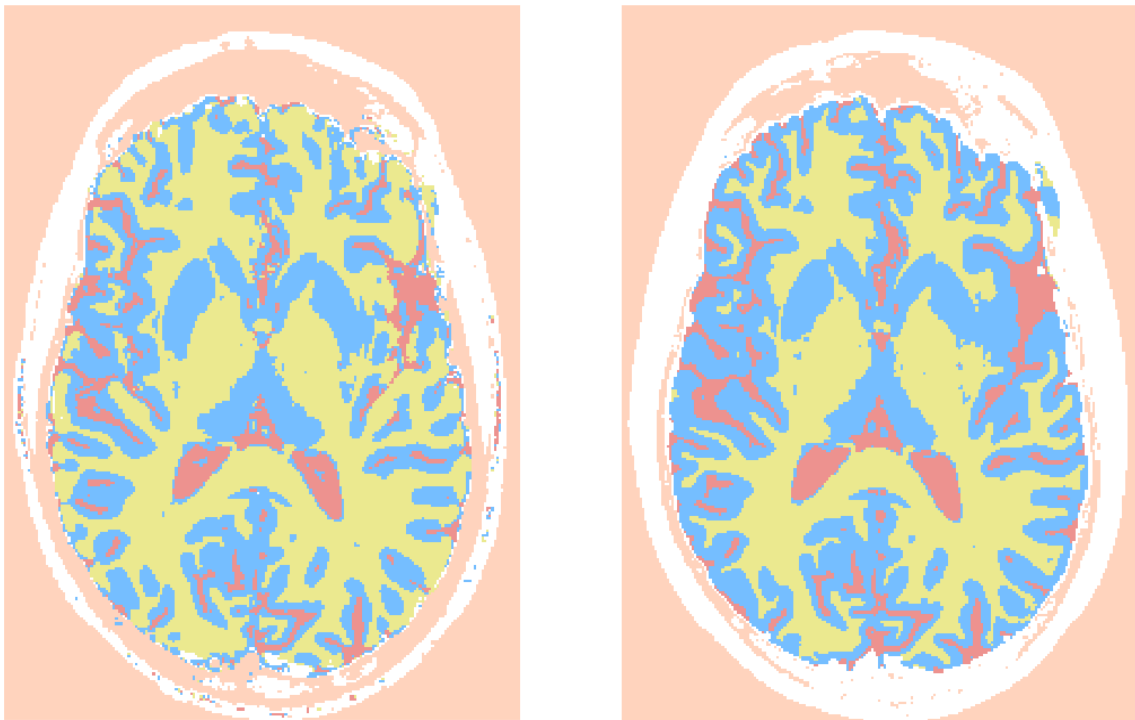


Figure B.1: Axial view of the segmentation with different sampling methods
Left figure (A) is an axial view of the T1 the target volume to be segmented. Right figure (B) is an axial view of the segmentation's result. Air is represented in pink, skull in white, white matter in yellow, grey matter in blue and cerebrospinal fluid in red.
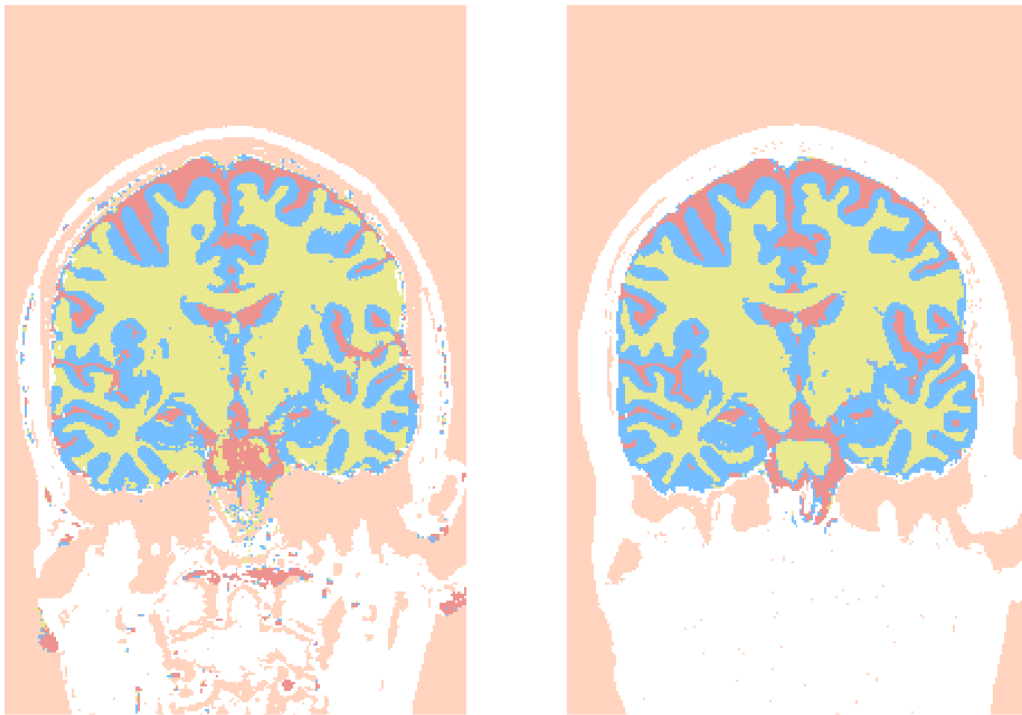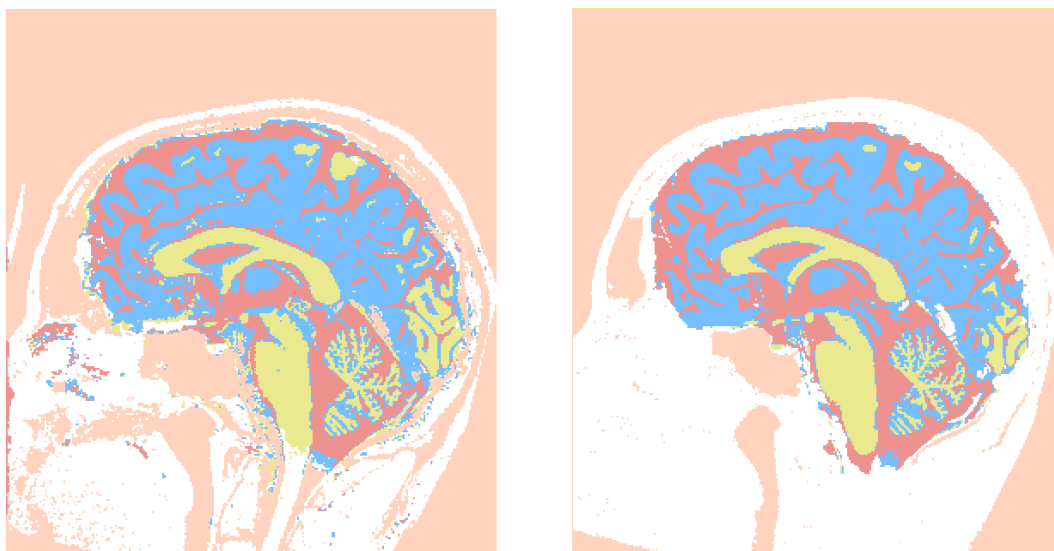
Figure B.2: Coronal view of the segmentation with different sampling methods
Left figure (A) is an axial view of the T1 the target volume to be segmented. Right figure (B) is an axial view of the segmentation's result. Air is represented in pink, skull in white, white matter in yellow, grey matter in blue and cerebrospinal fluid in red.



Left figure (A) is an axial view of the T1 the target volume to be segmented. Right figure (B) is an axial view of the segmentation's result. Air is represented in pink, skull in white, white matter in yellow, grey matter in blue and cerebrospinal fluid in red.

Figure B.3: Sagittal view of the segmentation with different sampling methods