LILLE - BREST - TOULON

ECOLE D'INGENIEURS

Institut Suprieur de l'Electronique et du Numrique
Tel. : +33 (0)2.98.03.84.00
Fax : +33 (0)2.98.03.84.10
CS 42807 - 29228 BREST Cedex 2 - FRANCE

N5

School Year 2008/2009
Training Period

New Expectation Maximization segmentation workflow 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, Massachusset's Insitut of Technologies - sylvain@csail.mit.edu

Referring Tearchers: 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 have 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 tissue segmentations 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 developped its own algorithm but it can be hard to understand and to use.

In this context, this document first proposes a full description of the EM algorithms. Then solutions are brought to enhance the segmentation. Contributions are very various, from core the algorithm to the graphic user interface (GUI). We added a bias field correction to correct the inhomogeinities in the MR image in the algorithm, as a preprocessing step in the algorithm. We also proposed 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 are also presented. Finally, the results obtained with the new segmentation workflow are evaluated by an expert.

Keywords: segmentation, expectation, maximization, class distribution, multivariate normal distribution, bias correction, intensity inhomogeinities

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 mais celui-ci peut etre difficile a comprendre et à utiliser.

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 inhomogéinites dans l'image RM, en tant que pré-traitement dans l'algorithme. De nouvelles méthodes pour definir la distribution des tissus a segmenter, via l'interface graphique utilisateur sont proposées. 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 des classes, distribution normale avec plusieurs variables, 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 physicists from errors and to help him to proceed to more precise moves. After the surgery, to see if it succeeded, or to follow the pathology of a patent. The informations brought to the physicists by the tools must be accurates, robusts and provide a fast feedback.

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

For the MR images segmentation purpose, the Surgical Planning Laboratory (SPL), Harvard Medical School and Brigham and Women's Hospital, has developped an EM algorithm to segment brain's magnettic resonance (MR) images. The results obtained are good until the intensity of the image to segment is homogeneous. Moreover, the implementation is not widely used so far, regarding the complexity of the segmentation process. Finally, some procedures seems not very accurate. In this report we will present an approach to enhance the segmentatin for inhomogenous MR images, correctiong 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.

Chap. 2 deals with the EM segmentation. Fundamentals will be reminded and the algorithm used in the SPL will be fully described. We will also present in this chapter the limitations of the current implementation. Chap. 3 describes our contributions. It explains the solutions brought to enhance the segmentation and to improve the usability of the current framework. Chap. 4 shows the results obtained. It aslo discussed about what have been done, the limitations of the new workflow and the next work wich has to be done.

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 firstly a simple approach of the problem, in the particular case of gaussian mixture (GM) models followed by a more generalistic approach. The simple approach will give you 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 used in Slicer 3[1].

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 in several times 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], motion estimation[2], etc.. Since the generalization of the algorithm, a lot of related papers were proposed. Most of them bring algorithms derived from the original one to adapt it to particuliar problems using additional informations.

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, structural and intensity inhomogeneities informations to segment the brain.

2.2 Fundamentals

Here we get going with a presentation of all the fundamentals you need 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. This part is mainly inspired from [4], [5] and [6]

2.2.1 Statistical model used for the brain

We define the voxel intensities of a MR image as $Y = \{y_1, ..., y_n\}$ when the image consisted in n voxels. Each y intensity is called *observed data* because this is the the data we see when we

[1] open source software developped in the SPL for biomedical engineering purpose

observe the image. Each y is a realization of the random variable Y. The real labelling of the image is Z. Z is called *hidden data* because we don't 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 Φ. The parameter Φ can either be a probability density function, noise, bias field, etc., 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 realisation 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 this intensity is corrupted by factors like noise, with a Gaussian Distribution. We can describe the relationship as below:

$$y_i = \mu_k + n_i$$

where μ_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 σ 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:

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

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

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

$p(Z_i = k|\Phi)$ is the *prior probability*. It expresses the probability that a voxel i belongs to a class k. $p(Y_i = y|Z_i = k, \Phi)$ is the *likelihood*. In our case, we will assume that the *prior probability* is constant. The new model we obtain for the labelling, via a gaussian distribution, 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 Φ contains unknown means, variances and weights for each tissue. Then we can express Φ 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:

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

In the case of *gaussian mixture 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 approxmation of the sum of all the probability density functions.

2.2.3 Maximum likelihood

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

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

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

The objective is to find the parameter Φ 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:

$$\begin{aligned} L(\Phi) &\triangleq \log p(Y|\Phi) \\ &= \sum_{i=1}^{n} \log \sum_{k=1}^{K} p(Y_i = y|Z_i = k, \Phi) p(Z_i = k|\Phi) \end{aligned}$$

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 Φ.

For example, to find the maximum likelihood regarding μ_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 μ_k:

$$\begin{aligned} \frac{\partial}{\partial \mu_k}(L(\Phi)) &= \frac{\partial}{\partial \mu_k}(\sum_{i=1}^{n} \log \sum_{k=1}^{K} G(y_i, \mu_k, \sigma_k) c_k) \\ &= \sum_{i=1}^{n} \frac{G(y_i, \mu_k, \sigma_k) c_k}{\sum_{j=1}^{K} G(y_i, \mu_j, \sigma_j) c_j} \frac{\partial}{\partial \mu_k}(-\frac{(y_i - \mu_k)^2}{2\sigma_k^2}) \\ &= \sum_{i=1}^{n} \frac{G(y_i, \mu_k, \sigma_k) c_k}{\sum_{j=1}^{K} G(y_i, \mu_j, \sigma_j) c_j} (\frac{(y_i - \mu_k)}{\sigma_k^2}) \\ &= \sum_{i=1}^{n} \frac{p(Y_i = y|Z_i = k, \Phi) p(Z_i = k|\Phi)}{\sum_{j=1}^{K} p(Y_i = y|Z_i = j, \Phi) p(Z_i = j|\Phi)} (\frac{(y_i - \mu_k)}{\sigma_k^2}) \end{aligned} \tag{2.7}$$

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

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

Thus, setting the denominator to 0 in equation (2.7) and using equation (2.8)yields:

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

Let us denote

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

Equation (2.9) leads us to:

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

Proceeding the same way as we did for equation (2.9), we can get similar equations for variance σ_k and weight c_k.

We find that:

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

$$c_k = \frac{1}{n} \sum_{i=1}^{n} p_{ik} \tag{2.13}$$

Equations (2.11),(2.12), and (2.13) provides us a formulation for soft segmentation.

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

This equation expresses that a voxel i belongs to the class k. p_{ik} is called soft assignement. 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 pixel 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 (Φ in our case). Here we first get going with an intuitive description of the algorithm in the particular case of GM 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 GMM). 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 GM model. In the other case, the log-likelihood can not be maximized by direct differenciation and a generalized approach must be used.

Generalized algorithm

Now we assume that we are no longer working with GM 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 Φ such as $p(Y|\Phi)$ is a maximum. We have:

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

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

$$\begin{aligned}
L(\Phi) - L(\Phi^{(m)}) &= \log p(Y|\Phi) - \log p(Y|\Phi^{(m)}) \\
&= \sum_{i=1}^{n} \{\log \sum_{k=1}^{K} p(Y_i|z_{ik},\Phi)p(z_{ik}|\Phi) - \log p(Y_i|\Phi^{(m)})\} \\
&= \sum_{i=1}^{n} \{\log \sum_{k=1}^{K} p(Y_i|z_{ik},\Phi)p(z_{ik}|\Phi).\frac{p(z_{ik}|Y_i,\Phi(m))}{p(z_{ik}|Y_i,\Phi^{(m)})} - \log p(Y_i|\Phi(m))\} \\
&= \sum_{i=1}^{n} \{\log \sum_{k=1}^{K} p(z_{ik}|Y_i,\Phi^{(m)}).\frac{p(Y_i|z_{ik},\Phi)p(z_{ik}|\Phi)}{p(z_{ik}|Y_i,\Phi_n)} - \log p(Y_i|\Phi_n)\} \\
&\geq \sum_{i=1}^{n} \{\sum_{k=1}^{K} p(z_{ik}|Y_i,\Phi^{(m)}) \log \frac{p(Y_i|z_{ik},\Phi)p(z_{ik}|\Phi)}{p(z_{ik}|Y_i,\Phi^{(m)})} - \log p(Y_i|\Phi^{(m)})\}
\end{aligned}$$

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 ln 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 consisted in n voxels. For example, $z_{ik} = e_k = \{0,...,0,1,0,...,0\}$ means that voxel i belongs to class k.

$$\begin{aligned}
L(\Phi) - L(\Phi^{(m)}) &\geq \sum_{i=1}^{n} \{\sum_{k=1}^{K} p(z_{ik}|Y_i,\Phi^{(m)}) \log \frac{p(Y_i|z_{ik},\Phi)p(z_{ik}\Phi)}{p(z_{ik}|Y_i,\Phi^{(m)})} - \log p(Y|\Phi^{(m)})\} \\
&= \sum_{i=1}^{n} \{\sum_{k=1}^{K} p(z_{ik}|Y_i,\Phi^{(m)}) \log \frac{p(Y_i|z_{ik},\Phi)p(z_{ik}|\Phi)}{p(z_{ik}|Y_i,\Phi^{(m)})p(Y_i|\Phi^{(m)})}\} \\
&= \sum_{k=1}^{K} p(e_k|Y,\Phi^{(m)}) \log \frac{p(Y|e_k,\Phi)p(e_k|\Phi)}{p(e_k|Y,\Phi^{(m)})p(Y|\Phi^{(m)})} \\
&\triangleq \Delta(\Phi|\Phi^{(m)})
\end{aligned}$$

We can then conclude that:

$$\begin{aligned}
L(\Phi) &\geq L(\Phi^{(m)}) + \Delta(\Phi|\Phi^{(m)}) \\
&\geq l(\Phi|\Phi^{(m)})
\end{aligned}$$

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 Φ 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 Φ for which the fonction $(l(\Phi,\Phi^{(m)}))$ and $L(\Phi)$ is $\Phi = \Phi^{(m)}$. Therefore, any Φ 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 research. $\Phi^{(m+1)}$ is the updated value which is get after maximization of Φ using $\Phi^{(m)}$.

$$\begin{aligned}
\Phi^{(m+1)} &= \arg\max_{\Phi}\{l(\Phi|\Phi^{(m)})\} \\
&= \arg\max_{\Phi}\{L(\Phi^{(m)}) + \sum_{k=1}^{K} p(e_k|Y,\Phi^{(m)}) \log \frac{p(Y|e_k,\Phi)p(e_k|\Phi)}{p(e_k|Y,\Phi^{(m)})p(Y|\Phi^{(m)})}\} \\
&\text{As } p(e_k|Y,\Phi^{(m)}) \text{ and } p(Y|\Phi^{(m)}) \text{ do not depend on } \Phi \\
&= \arg\max_{\Phi}\{\sum_{k=1}^{K} p(e_k|Y_i,\Phi^{(m)}) \log p(Y|e_k,\Phi)p(e_k|\Phi)\} \\
&= \arg\max_{\Phi}\{\sum_{k=1}^{K} p(e_k|Y,\Phi^{(m)}) \log \frac{p(Y,e_k,\Phi)p(e_k,\Phi)}{p(e_k,\Phi)p(\Phi)}\} \\
&= \arg\max_{\Phi}\{\sum_{k=1}^{K} p(e_k|Y,\Phi^{(m)}) \log p(Y,e_k|\Phi)\} \qquad (2.15) \\
&= \arg\max_{\Phi}\{E_{Z|Y,\Phi^{(m)}}\{\log p(Y,Z|\Phi)\}\}
\end{aligned}$$

We notice from equation (2.15), that , for a given voxel i:

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

Now, both expectation and maximization steps are apparents.

- **E-step**
  This this 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.

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

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

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

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

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 Φ, at step m and step $m+1$ is smaller than ε:

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

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 clearly and we can summarize the basic algorithm with the figure (2.1):

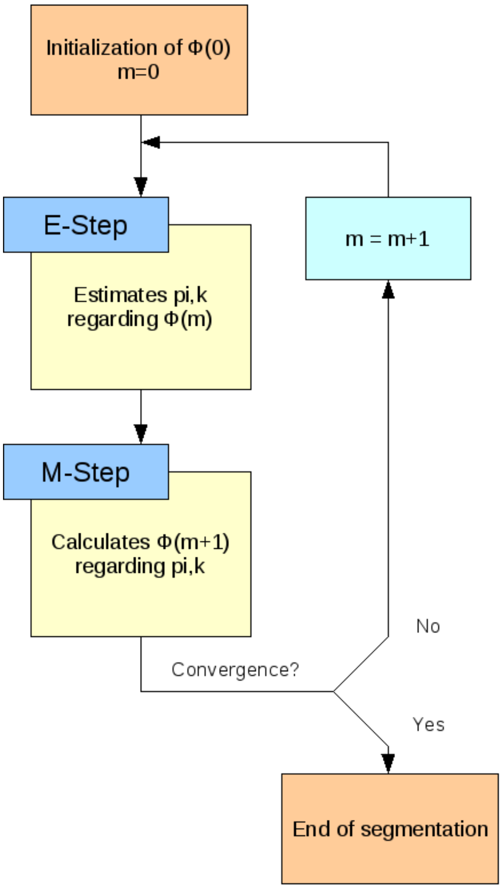


Figure 2.1: Basic EM algorithm

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 of 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 GM 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 you want 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 modalities are used to process brain segmentation. Indeed, the best suited modality to use depends on the tissue you want to segment. For example, T1[2] MR images are well suited to segment white matter (WM) but are not accurate for cerebrospnial fluid (CSF). On the contrary, T2[3] MR images are well suited for CSF and not for WM. To formalize the utilisation of different MR images sequences during the segmentation, we will change the definition of y_i, μ_k and σ_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 or/and to patient induced electrodynimic 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. Then, the degradation at each voxel can 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 transforms makes the bias field becomes 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 basics functions $\Psi_l(x)$ ([13]). Let b_i be the realisation 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 GM model, bias field can then be estimated using EM framework. The bias field parameter A will be used during the E-Step , through bi 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]):

[2] MR images acquisition sequence designed to enhance the grey matter/white matter contrast. See [12].

[3] MR images acquisition sequence designed to enhance the grey matter/cerebrospinal fluid contrast. See [12].

- **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 not to be 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 informations about the possible error for each voxel. The covariance matrix will be descibed 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 GMM. 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, duringthe maaximization:

$$\begin{aligned}\hat{\Phi} &= \arg\max_{\Phi} p(\Phi|Y) \\ &\text{As Bayes's theorem can be applied} \\ &= \arg\max_{\Phi} \frac{p(Y|\Phi)p(\Phi)}{pY} \\ &\text{As } p(Y) \text{ do not depend on } \Phi \\ &= \arg\max_{\Phi} p(Y|\Phi)p(\Phi)\end{aligned}$$

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

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

In Wells' method, the only parameter to be 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: Σ_B^{-1}. We also set F to unix matrix 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 subdivise the segmentation process into subproblems, that are easier to solve, according to Pohl.

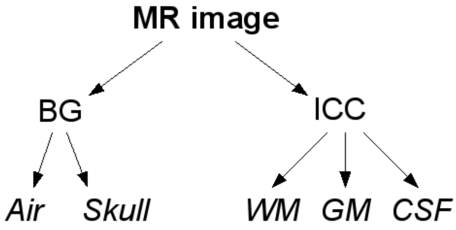


Figure 2.2: A simple tree structure of the brain

Here we continue with an intuitive description of the process. It is a brief explanation of how the tree figure (2.2) would be segmented using the hierarchical information. 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 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.

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

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 voxels 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

Each tissue will then 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 worklfow 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 remind why is each initialisation step important and where is this information used.

2.5.1 User interface

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

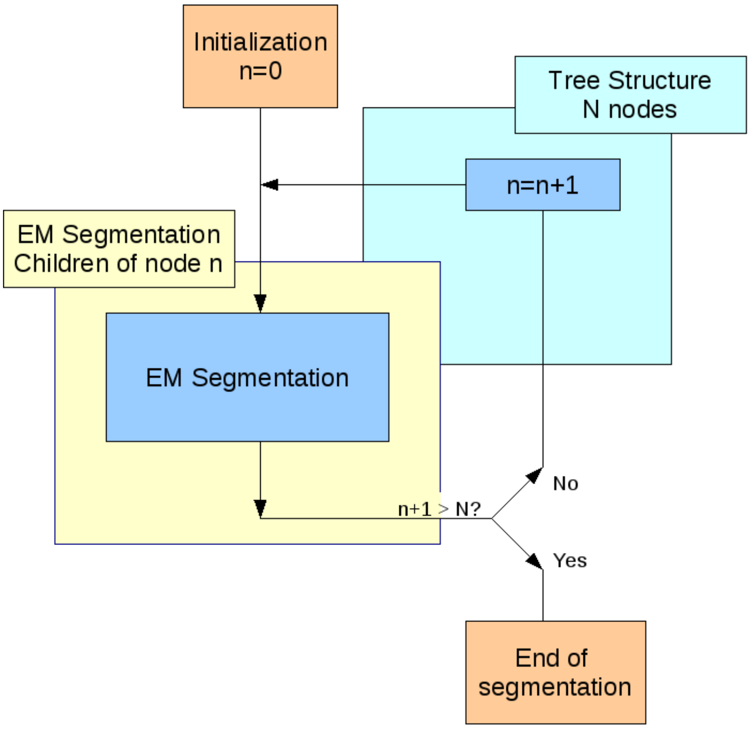


Figure 2.3: EM segment algorithm in Slicer

- **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. to each tissue to segment, the related atlas. It will be used for the spatial information (section (2.4.1)). It implies that you need an atlas for each structure that you want to segment.

- **Step 3: Multimodal segmentation**
  We choose the images we want to use for the segmentation. As discussed in section (2.4.2), it is usefull for the mutlichannel segmentation since some tissues are not clearly visible 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. Moreover, it is precise way to initialiase the class tissues distribution for the algorithm. It it usefull because the EM algorithm only estimates the bias field but still requires informations about the tissues to be segmented. Using these values, the algorithm is initialized then estimates the bias field.

- **Step 6: Hierarchical paramters**
  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 usefull informations like which modality is the more relevant for the class and the size that the class is supposed to be.

For example, for the CSF. Let's assume that we proceed to a multi channel segmentation with T1 and T2 MR images. 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 in consequence and only use the information from T2 to do 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 not.

- **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 point of choising a registration method is presented in the next section.

2.5.2 Algorithm

After all the initialisation steps done via the GUI (section 2.5.1), we will no present the core of the segmentation pipeline.

- **Step 1: Intensity normalisation**
  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 dected in the histogram, using its low intensity and the important number of pixels of which it is consitituated. 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 excpeted). This normalization is usefull if we want to use the command-line module (i.e. if we want to run a lot of segmentation 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. It's is usefull because now, all the target images should have the same initial parameters.

- **Step 2: Images registration**
  In order for the atlas to guide the segmentation (spatial information), it as 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 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 mutltichannel segmentation, using two $200*300*300$ targets volumes (T1 and T2 MR images) and segmenting fives tissues, lasts around twenty minutes.

  The whole algorithm pipeline in summarized in figure (2.4). It is inspired from [18].

2.5.3 Summary

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

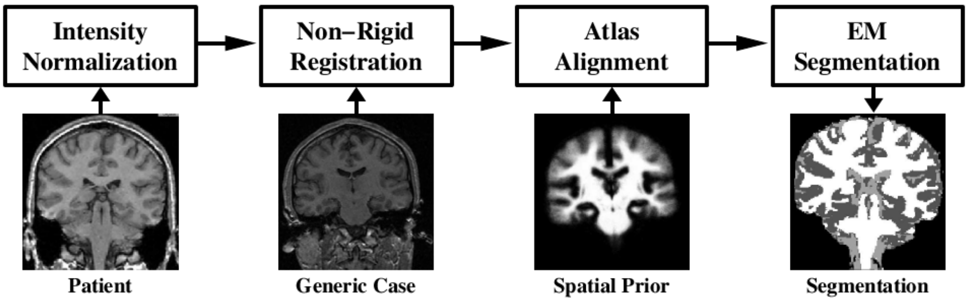


Figure 2.4: The algorithm segmentation pipeline in Slicer 3

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. The user has no tools to find the good normalization value. He has to guess it. 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 tricky and we provide a tool to help the user (3.5).

The last problem we encounter appears after the initialization, during the pre-processings. Only one pre-processing (intensity normalization) is done on the target images before the atlases are registered to it . Since the EM algorithm is mainly designed to work on MR images, bias field is a recurrent problem. Proceeding to a registration before a bias correction can deteriore a lot the results of the registration. This problem will be discussed in details in section (3.3).

The whole EM segmentation algorithm use in Slicer 3 has been presented. We also defined of the main issues of the current workflow. In the next chapter, we will explain the problems more deeply and show you the solutions we brought to solve them.

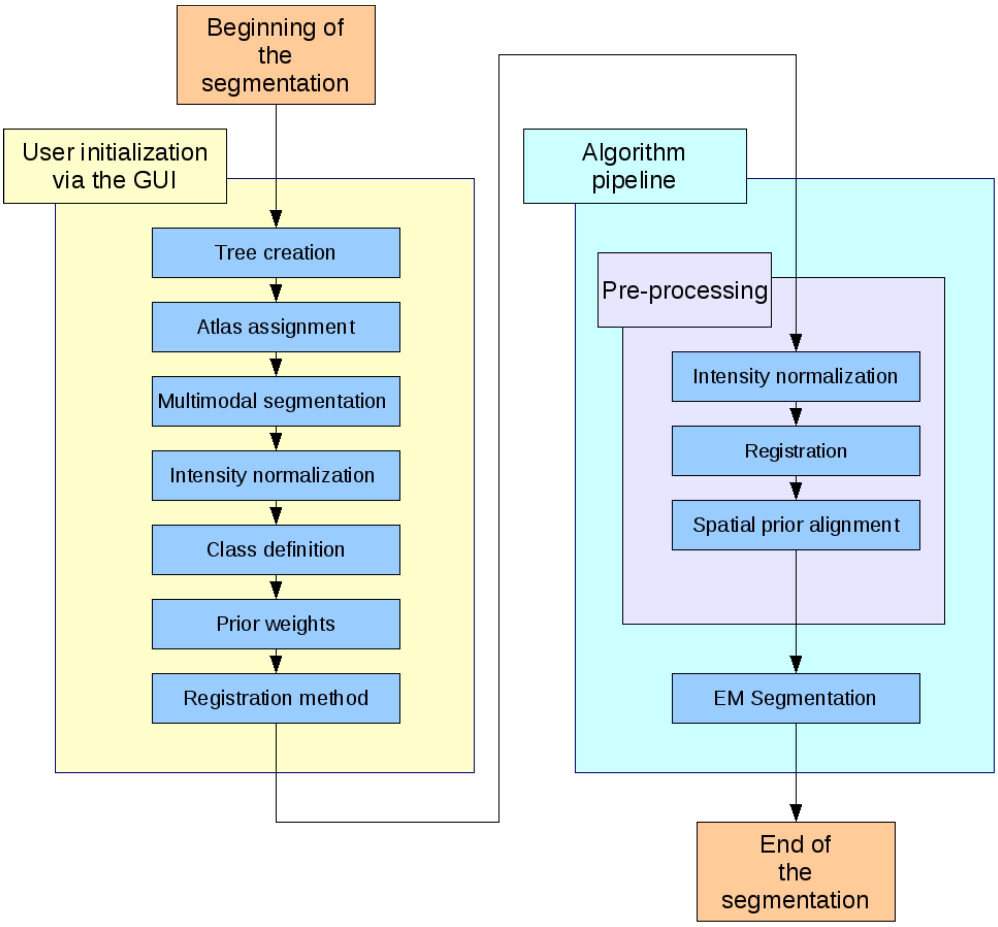


Figure 2.5: The whole segmentation pipeline in Slicer 3

Chapter 3

Contributions

In this chapter we will present all the contribution brought to enhance the segmentation workflow in Slicer 3. We propose solution to the problem cited in the previous chapter (2.6). We propose solutions to enhance the class selection method and to allow the user to evaluate his selection. We deal with the registration problem in a third step. The Finally, we present some tools we added to help the user to find the good intensity normalization value and an estimation of 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 Interest

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

The first possiblity consists in entering manually the intensities mean value and variance for each class, for each volume to be processed. This way, the user can be very precise and accurate when he defines each class. But it is very hard to find the good mean value and variance for each class for each volume. Morever, each time we want to process a new volume, we have to redefine mean values and variances. It is not convenient and it can a lot of time to find accurate values for the parameter initialization.

The next approch consisted in defining a class model by manual sampling. For each class, the user clics in the related part of the volume. The problem with this method is that you compute your mean value and variance using only a few samples. Your sample will never be bigger that one hundred points because it is not convenient. Then, your mean values and variances are not accurate. Moreover, results are not reproducible with this method. This the number of samples is reduced, means and variances can vary a lot with one more sample and you can never reproduce two times the same initialization.

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

3.1.2 Method used

The idea is to create a label map. This map contains colors. There is one color for each class we want to segment. 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 label map by coloring caracterisc regions for each tissue to segment, in the appropriate color. This gives a spatial information to the algorithm. It can now estimate automaticly the mean value and covariance of each class, for each tissue, using this label map.

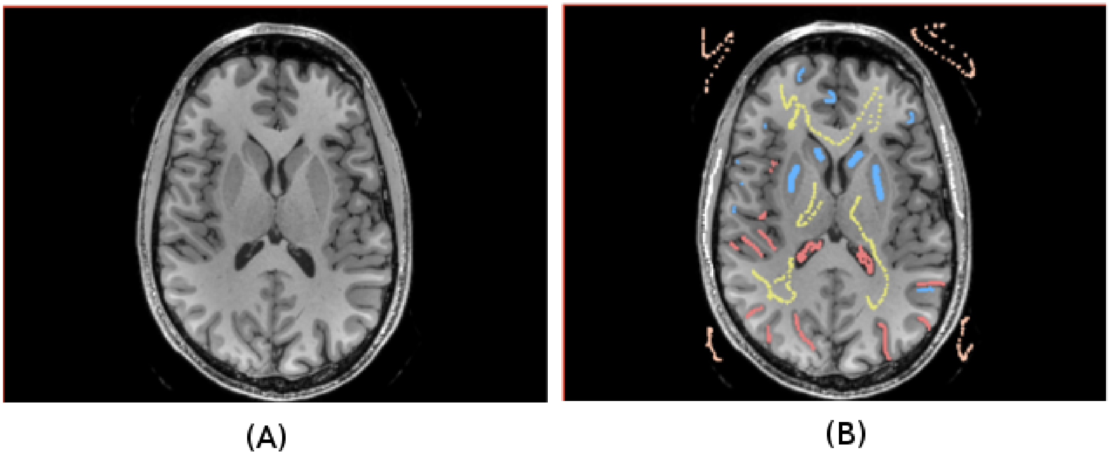


Figure 3.1: Axial view of the label map.

The reason why covariance matrix is estimated, instead of the simple variance is presented in the next section.

It is very convenient because, since the algorithm needs a good initialization, we can easily define a sample of hundred of points for each class. The results will representatives. Moreover, the results are now reproducible. Indeed, we can store then re-use the same label map. The results will remain the same.

Figure (3.1) represent an example of a label map. (A) represent an axial view of the vloume to be segmented and (B) the label map created for this volume. Each color represents a tissue to be segmented.

3.1.3 Evaluation

To estimate the influence of the contribution, we processed to a simple comparaison. We selected manually ten points of the white matter as fine as possible. With this sample, we estimate mean value and covariance matrix, through 2 volumes. The second step of the evaluation consisted in estimating the same values for the same tissue using a label map (sample bigger than 200 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 (μ_{Manual} and μ_{Labell}) differs slightly and covariance matrices (Σ_{Manual} and Σ_{Label}) differ significantly. It means that our approach is usefull and it shows the importance of having a large sample to evaluate means and covariance values in a such process. The square variance of the class, is in position $(1,1)$ in the covariance matrix for the first volume. 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 way more precise with label map than with 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 Interest

As discussed before, the algorithm is sensible to the initalization. It means that the initialization has to be good. Once the parameters are chosen, the user has no tools to know if his selection is accurate. Two classes to segment can't have too close means and variances. Even if the user sees the values he chooses, it is not easy to know if two classes to be segmented are too similar or not.

3.2.2 Our approach

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

We first assumed that each class has a normal distribution. We first decided to plot the gaussian in 3D, using the multivariate normal distribution. In the 2-dimensional nonsingular case, the probability density function is

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

(see [15]). x and y are the position of the pixel in the 2D space. $f(x,y)$ will return the value (heigth) of the (x,y) pixel. Each X and Y axe represent one volume. Let's first say that the range of the X and Y axes are the intensity range of the X and Y volumes. μ_x is the mean value of the class in the X volume. μ_y is the mean value of the class in the Y volume. σ_x and σ_y are the variance of the tissue in its respective volume. ρ 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 ρ from the covariance matrix Σ (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't use the same density probability function anymore. We use the classic normal distribution's formulation

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

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 not good 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 re-defined for each image. Let's present it for a given image X. It is now the difference between Max, the maximum value extracted with the label map, between all the samples between all the classes for the image X, and Min, its opposite.

For our particular purpose of tissues' distributions' vizualisation we didn't use exactly the formulas (3.1) and (3.2). Indeed, we don't normalize the curves: we set $\frac{1}{2\pi\sigma_y\sigma_y\sqrt{1-\rho^2}}$ to 1 in (3.1) and A to 1 in (3.2). We do it because we have no clue about the importance of each class so

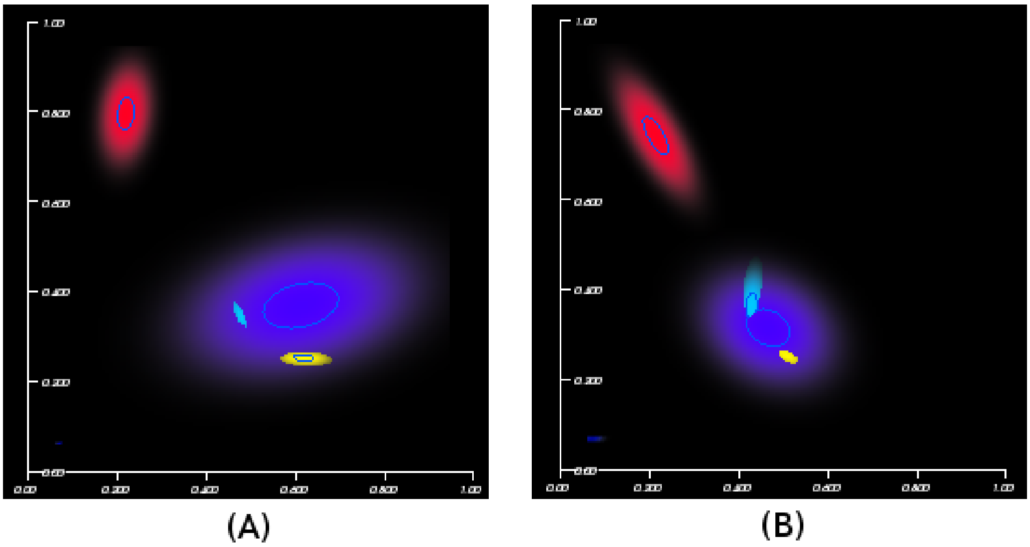


Figure 3.2: Distribution of the class to be segmented

we don't want to "disavantage" any one. A compromise could be to have an amplitude factor proportional to the number of pixels which are supposed to constitute the class.

We finally obtain the results figure (3.2) for different sampling methods.

(A) present the visualization we obtain after a manual sampling and (B) the results after a label map 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. On (A), the skull is significantly more spreaded than in (B), especially along the X axe (T1). According to an expert, this result is better in the case of labelmap sampling, indeed, the skull should almost have the same variance through T1 and T2, what is almost the case in (B). Moreover, regarding the CSF and the grey matter, the distribution is better in (B) again. According to the same expert, the intensities of CSF and grey matter, in the T1 volume are close and can't be as distant as in (A). These two observations show 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 has intensities inhomogeinities. Moreover, bias field canal present problems regarding the classes distribution. We will first remind the problems, then we present the solutions proposed.

3.3.1 Interest

In the segmentation process, a registration step is required. Registration consists in finding a trasformation 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 treat MR images. MR images are often corrupted by a bias field. Thus, the image to register present intensities inhomogeinities. These inhomogeinities can deteriore a lot the registration.

On figure (3.3), we present the result of the registration between an atlas and a biased MR image. Note that the target MR image has been normalized to have the same mean value as the atlas. The results is clearly bad. A solution must be brought to enhance this step and so the segmentation.

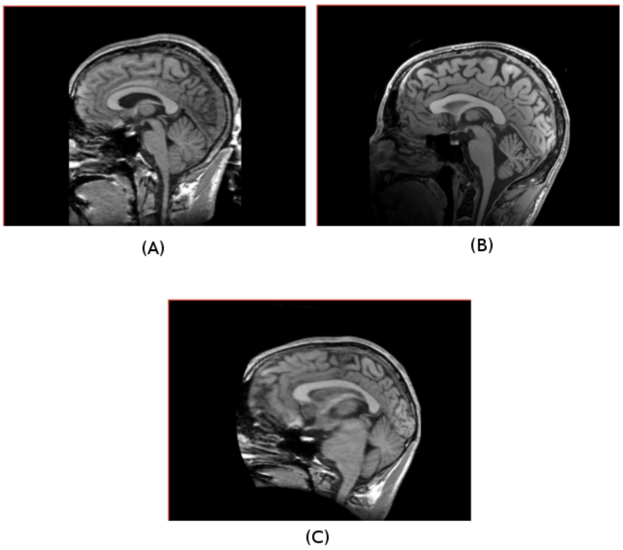


Figure 3.3: Result of registration of a biased MR image without correction

Moreover, this bias field lead to another main issue. Indeed, if the MR image is biased, a the intensity of a given tissue will vary a lot through the volume, even if it should not. Then the algorithm will be initialized with wrong mean and variance values and the segmentation won't be as good as it would be with a corrected image.

3.3.2 Our approach

The idea simply consists in correcting the bias field of the MR image before this step. Thus, the registration will be significantly enhanced.Since the registration is better, it should also increase 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 doesn't require prior information like the number of tissues to correct or the mean value of each tissue to be corrected. We implement an ITK[1] filter ([14]) in Slicer 3.

We can describe the new segmentation workflow in Slicer 3 as we do in figure (3.4).

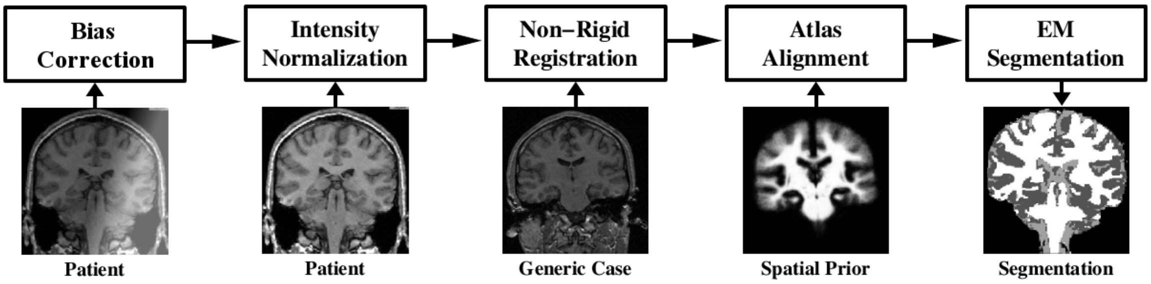


Figure 3.4: New algorithm pipeline

[1] open-source C++ toolkit for segmentation and registration. See [13].

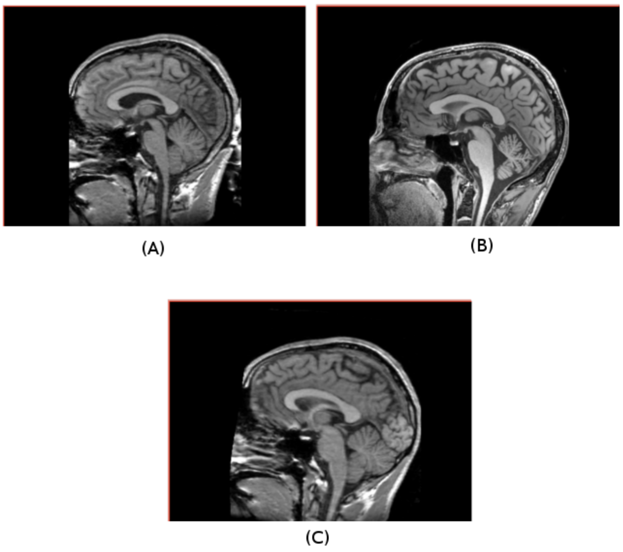


Figure 3.5: Registration after bias correction.

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, because it would be the first pre-processing step, it is possible to do so. The user will first have to correct the intensities inhomogeinities via the module then use the corrected images in the EM segmentation module.

After the bias correction, we obtain interesting results (3.5). (A) represents the atlas to be registered. (B) represents the target volume for the registration. (C) repre sents the atlas after the registration. The result of the registration visually appears to be better but we can't be satisfied of this visual verification. We need a formal evaluation method.

3.3.3 Evaluation

We evaluated accuracy of the registration using the joint histograms method. The joint histogram evaluation method is basic comparaison between two images. Let A be a matrix of size $W * L$. W will be the intensity range of the first image used for the comparaison. L will be the intensity range of the second image to be compared. The matrix is initialized to 0. Each time that in the same position, there is the same intensities in the two images, we add 1 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 gray level i at position x, y.

In figure (3.6), (A) and (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. Red becomes orange, yellow, green then blue.

In figure (3.7), (A) and (B) compares the joint-histograms of the corrected image and its atlas, before (A) and after (B) registration.

It clearly appears the the difference between before and after registration is more significant if the bias field is corrected. Indeed, we compare the biased joint histogram, there is no significative difference between before and after registration. That means that the registration didn't improve the similarity between the images. On the contrary, if we correct the bias field, it the number of point around the matix's diagonal clearly increases. It means that now, the images are more similiars than before. It shows the utility of the contribution. The influence of these results after the whole segmentation process will be presented in the next chapter.

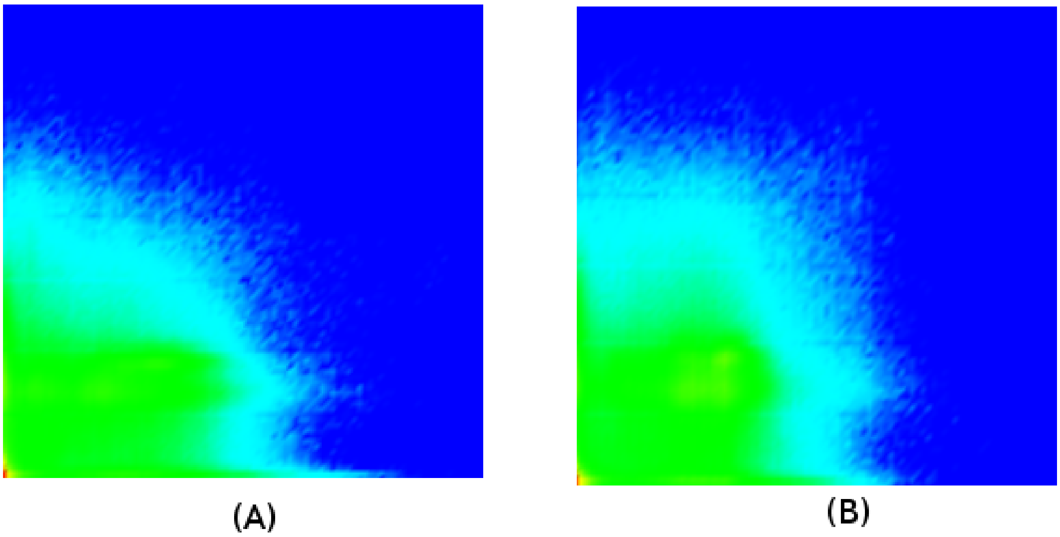


Figure 3.6: Joint histograms to evaluate the registration.

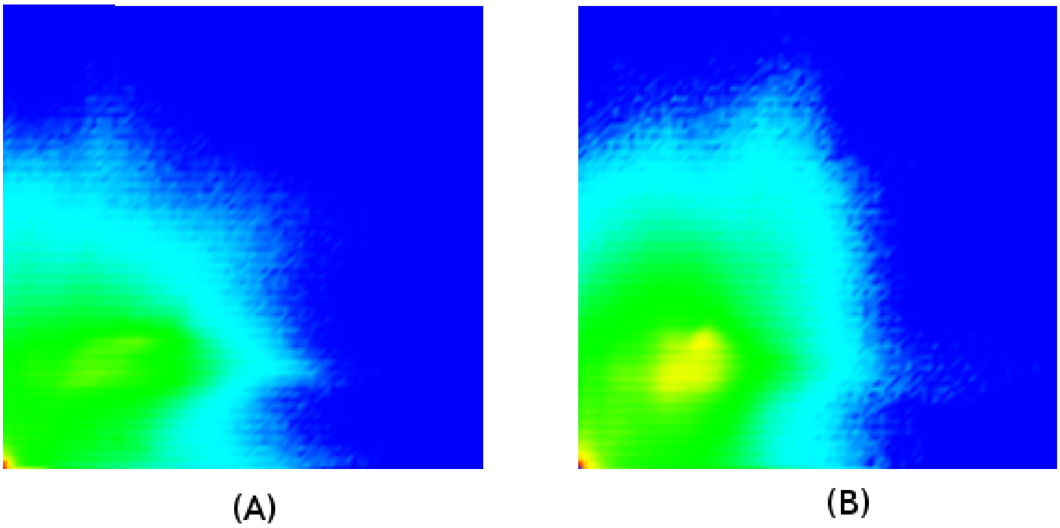


Figure 3.7: Joint histograms to evaluate the registration.

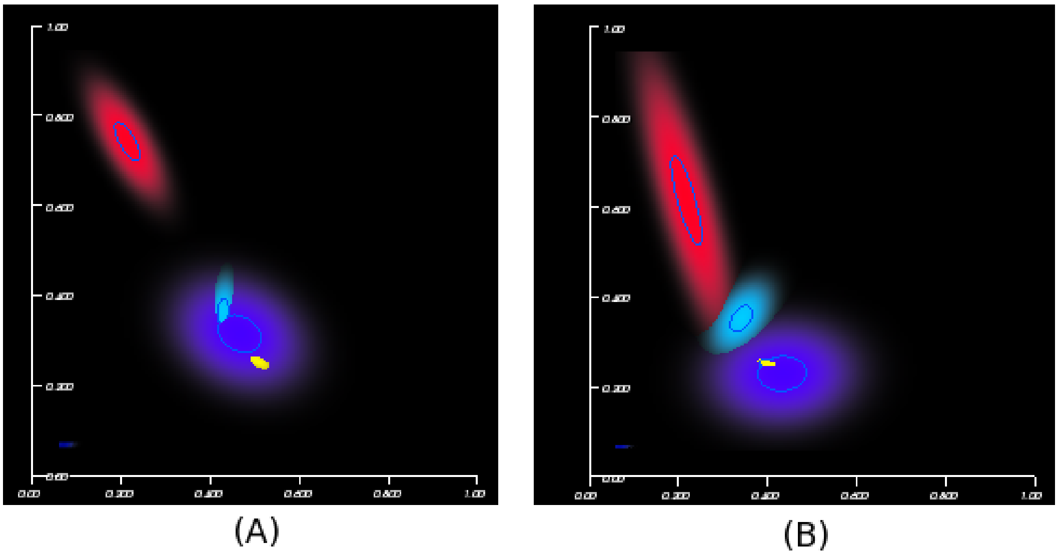


Figure 3.8: Tissue distributions.

Regarding the tissue intensity distribution effect, we will present some results using the tool we developped in the previous section, to evaluate the classes' distribution. Using the same labelmap for sampling, we obtain two totaly different distributions. Figure (3.8) clearly show the two different distributions. The relation tissue/color is the same as the one in figure (3.2). (A) presents the distribution before bias correction. (B) presents the distribution through the corrected volume.

The tissues variances are huge, especially for the CSF (red) through T2. Its distribution through the T2 volume is huge. According to an expert, it should clearly not. It should only be in the high intensities of the T2 volume. Our result means that the CSF class contents high and low intensities. It is due to the bias and is not acceptable.

Intuitively, we understand that the segmentation process should be a lot deteriored because of the two issues we presented (registration and distribution).

3.3.4 Registration parameters

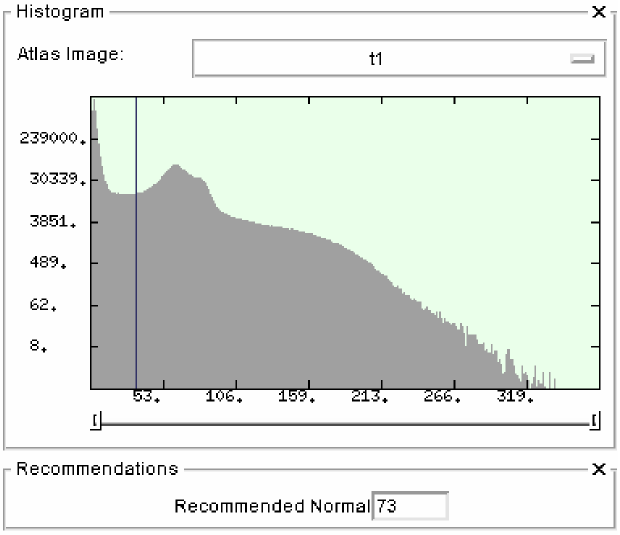
Even if we are doing a non-parametric registration, some parameters has to be defined. ”Non-parametric” means no information about the volume and classes to correct. We will first present and explain you the parameters. In a second step, we will propose you some parameters adaptated to different problems.

- **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 iterativly 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 is now obvious that if you want to increase the time of processing, you should increase the shrink factor or reduce the maximum number of iteration. The limitation is that is can deteriore the bias field correction. You can also increase or reduce the Bias field full width at maximum iteration, depending on the importance of the bias.



Atlas Image: t1

Recommended Normal 73

Figure 3.9: Tool developped for the intensity normalization parameter estimation

3.4 Intensity Normalization

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

3.4.1 Interest

As discussed in section (2.5.1), at the step 4, an intensity normalization is done. Will already presented the utility of an intensity normalization in the same section. The problem is that the user has no tools to find the good values for the segmentation. He has to guess the mean intensity of the voxels in the MR image, background exluded. This is of course not doingable in practise.

3.4.2 Our approach

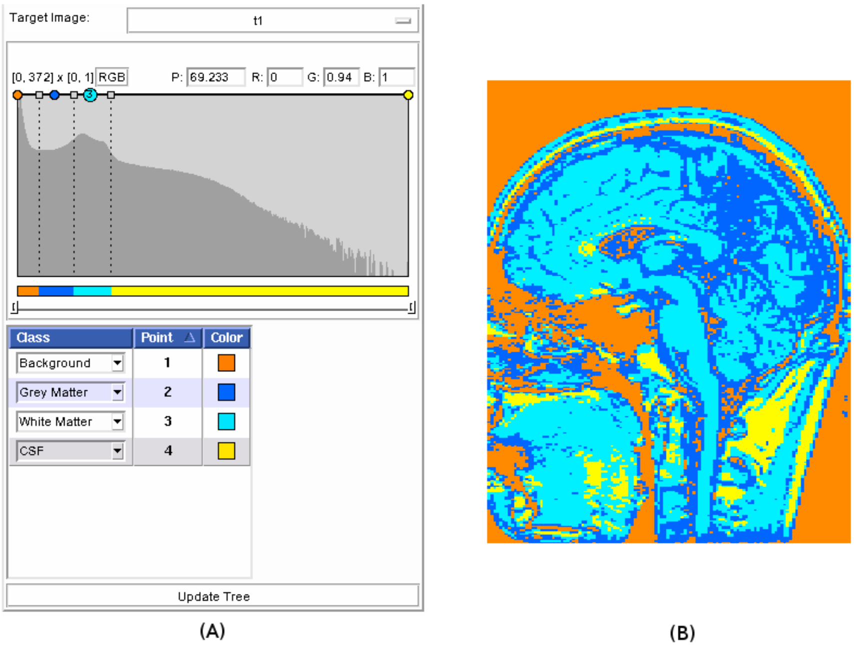
We implemented a simple tool, to allow the user to find easily and accurately this normalization value.

The first step of the work consited in creating the histogram associated to the image. The Y axe which presents the number of pixels for each intensity in the volume uses a log-scale because the range is huge. The log scale reduces considerably the range. We then added a cursor in this histogram. Using it, the user can choose the intensity which will be the ”limit” between the background and part 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.

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



Target Image: t1

[0, 372] x [0, 1] RGB P: 69.233 R: 0 G: 0.94 B: 1

Class	Point	Color
Background	1	🟧
Grey Matter	2	🟦
White Matter	3	🟦
CSF	4	🟨

Figure 3.10: Tool developped for the global prior weights estimation

3.5.1 Presentation of the problem

During the segmentation process,at the 6^{th} step of the intialization (section 2.5.1), you have 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, they user can spend a lot of time during this step. 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 Our approach

We divided the problem in two parts. The first part will be about providing the user a real-time feedback regarding the global prior weights estimation. The second part will consist in developping an algorithm which fills automatically the tree.

The new tool is presented in figure (3.10) (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, about about what he is doing. 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).

We presented the contribution we did to the EM semgentation module in Slicer 3. It lookds usefull, nevertheless we still don't know if it will have a real impact on the whole segmentation process. In the next chapter, we present the influence of the contributions on the whole segmentation proocess.

Chapter 4

Results and discussion

This chapter will present the importance of the contribution we did, regarding the final segmentation. We will not discuss about the enhanced usability of the module. It will allow you to see how the segmentation has changed. We will then discuss about the next contribution which could be brought to enhance the segmentation and the usability of a such module.

4.1 Results

Here we get going with a presentation of different results, using the different contributions. The results obtained will then be reviewed by a specialist to evaluate the segmentation. We work with the same dataset for all the segmentation. We chose a biased one to show the importance of the bias correction.

4.1.1 Original segmentation

We first present the segmentation obtained without any contribution. From these results, it will show the importance of the contributions done. We will first describe the testing process. Then we will present the results of the segmentation and finally, an expert will estimate it.

Testing process

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 followed it with a biased image. The atlases we used for the segmentation are the one available at `http://www.na-mic.org/Wiki/images/b/b7/AutomaticSegmentation.tar.gz`.

Results

The results of the segmentation are presented in figure (4.1).

We can explain the results in many different ways. First of all, the bias field was not corrected (see section 3.3). That means that the registration was poor . Moreover, tissues' distribution was bad. The means values were not accurate and the variances were too big. The initialization was bad and lead to a bad registration. In addition to the problem of intensities inhomogeinities during the tissues' distributions' initialization, we did a manual sampling. As presented in section 3.1, it can have importance during the segmentation.

Expert's point of view

It is clearly bad and we didn't really need an expert to see that the segmentation is not good. From the expert point of view, the grey matter is over estimated. This is exactly what we saw in

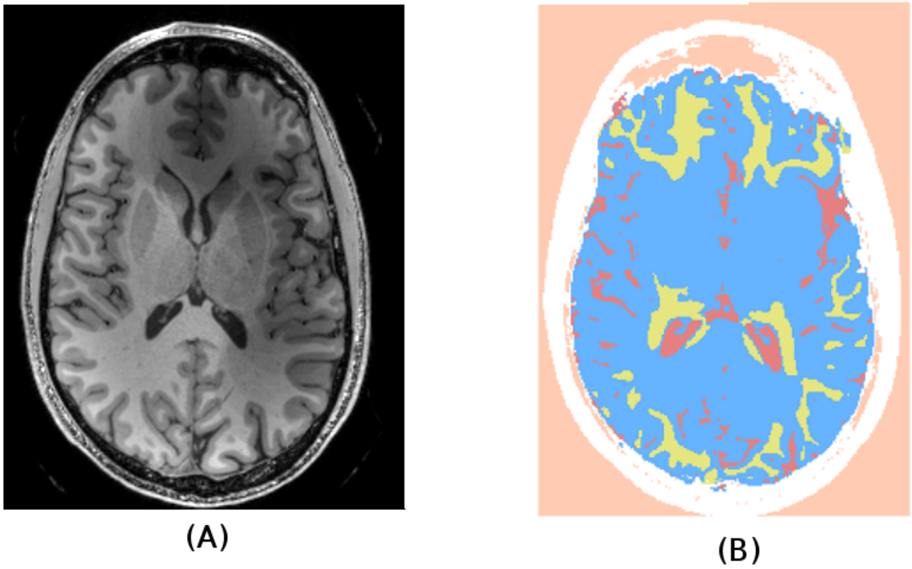


Figure 4.1: Results of the segmentation with bias correction

figure (3.8). The grey matter (light blue) is overestimated in (B), the class distribution through the biased volume.

4.1.2 Bias corrected segmentation

Here we present the results of a segmentation using the intensities inhomogeinities correction. We first describe the testing process. Then we present and interpret the results obtained. Finally an expert evaluates the results.

Testing process

We followed the same tutorial we used in the previous section. We proceeded exactly the same way. The only difference is that we now correct the image intensities inhomogeinities before the segmentation.

Results

The results of the final segmentation are presented in figure (4.2). (A) presents the original unbiased image, (B) the segmented image, using a manual sampling. Visually, the results appears better but as long as we are not experts, an expert will estimate the result of the segmentation in the next section.

We can explain the difference. It is due to a better registration and a better class distribution. We showed in section (3.3) that the registration was a lot improved. Moreover, this is not the only avantage. Indeed, correcting the bias field, the classes to be segmented have a better distribution. We demonstrated it in figure (3.8). The tissues have a better distribution, especially gray matter (light blue) in this case.

Expert's point of view

Some troubles appear in the skull's segmentation. The segmentation returns that there is grey matter and white matter in the bones.A bad label is attribuated to a tissue after the segmentation. It is called misclassification. It can be attribuated the the partial volume artifacts. The partial volume artifact effect is caused by imaging voxels containing two different tissues (skull and air in

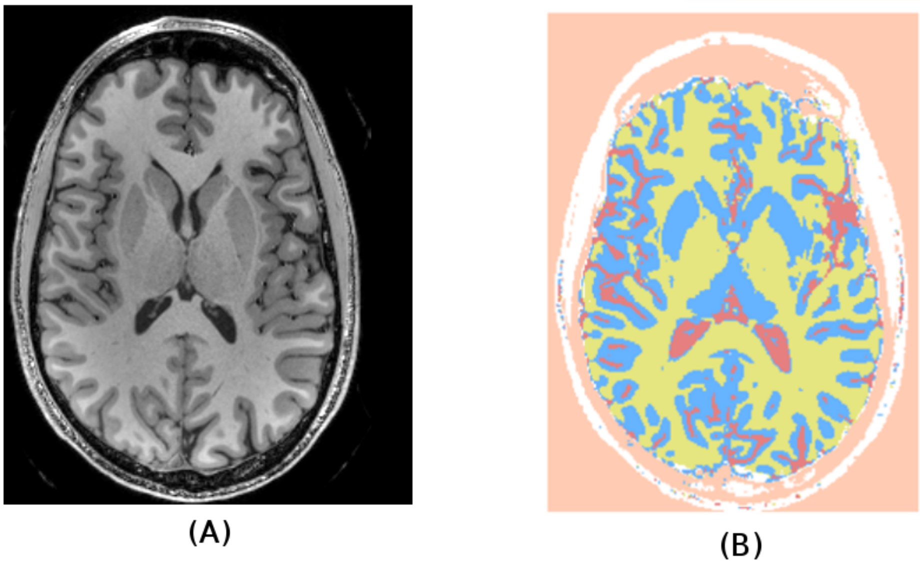


Figure 4.2: Results of the segmentation with label map

our case). The voxel's returning intensity will be an average of the two tissue present in the voxel. Moreover the skull is not perfectly segmented because sometimes it appears that there is big holes in the skull with air inside, which is not accurate. The misclassification in the other areas are mostly due to the partial volume artifacts.

4.1.3 Label map sampling segmentation

The segmentation will now let us evaluate the influence of the sampling method on the segmentation. We first describe the testing process. Then we present and interpret the results obtained. Finally an expert evaluates the results.

Testing process

We proceeded exactly the same way as we did during the bias corrected segmentation. The only difference is that now, we used a labelmap to initialize the classes distributions.

Results

The results of the final segmentation are presented in figure (B.3). (A) presents the original corrected image, (B) the result of the segmentation. Visually, we can't prononce ourselves and an expert will estimate the result of the segmentation in the next section. This time, the classes are supposed to have a better distribution than in the previous segmentation, as we saw in figure (3.2). HOLES FILLED, CLASS MORE HOMOGENEOUS. More screenshots with a better resolution are presented in App. B to have a better understanding of the differences between manual and labelmap sampling.

Expert's point of view

The results are pretty good too. The skull is better segmented. 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 very specific area of the brain because there is a lot of white and grey matter in a small area. The segmentation is hard and one time again, the results must due to partial volume effects. According to our expert is a complex task.

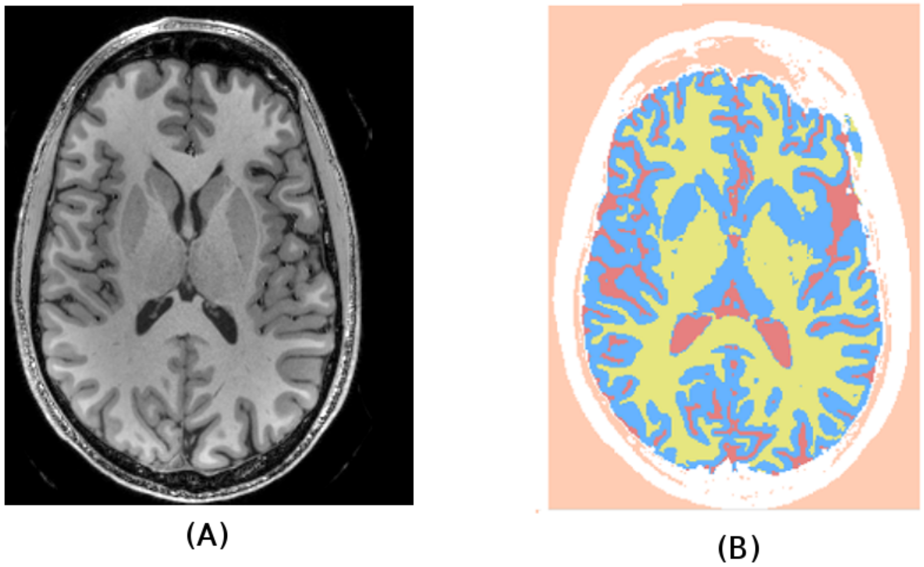


Figure 4.3: Results of the segmentation with bias correction

We can't easily segment it if we segment the whole brain. To segment the cerebrulum properly, we should only segment the region of interest.

4.2 Future work

The results are good. Nevertheless, a lot of work has still to be done in order to enhance the segmentation process. The main things to be done will be presented now.

Here we get going with the correction of the bias field in the MR image. The results are good but a major issue appeared: the time of processing to correct the bias. On a a 255x255x255 volume, it typically last more than thirty minutes. This is not acceptable and we have to go through the algorithm to have a deep understanding of the bias correction. It will be then possible to adapt the ITK filter to our problem. The time of processing can then be a lot reduced, according to Tustison.

Regarding the global prior weights estimation, a problem appears. When the image to be segmented contains some classes which have a close distribution, the tool is no longer effective. Indeed the estimation is only based on the intensity of the voxels. A usefull contribution could consist in representing each class as a gaussian, instead of a finite portion of the histogram. The histogram would be the result of the summation of all the gaussians. Assuming the each class is a gaussian is not a bad assumption since we us this definition in the EM algorithm.

These are the main improvement to be done in the next months.

Conclusion

We presented a new 5-steps algorithm to process segmentation on brain's MR images. It differs from the previous segmentation algorithm of Slicer 3 by the addition of a bias correction as a preprocessing step. The new approach is very usefull since MR images are often corrupted with a bias field. We also brought a lot a contributions to make the intialization more accurate and the segmentation process easier.

Results obtained with new segmentation pipeline are good. We first evalutated the results we could, like the quality of the registration, from our engineer point of view. Moreover, an expert evaluated the accuracy of the final segmentation and presents the positives and negatives points of all the tests processed.

In this context, using the expert advices, new perspectives appeared. Nevertheless, segmentation of the brain's tissues in MR images is a complex task. Indeed, a lot of tests have to be processed to ensure that the results are usefull and that the contribution will not deterior the current segmentation process. More generaly, there are still a lot of things to do to make this segmentation algorithm faster and easier to use. Since the EM algorithm is very flexible, a new algorithm will not have to be developped. The one we brought will then remain as an important contribution and will be updated and used until a new method, better than EM segmentation, appears for MR brain's images segmentation.

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 ot 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.

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 $n + 1$**

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

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.

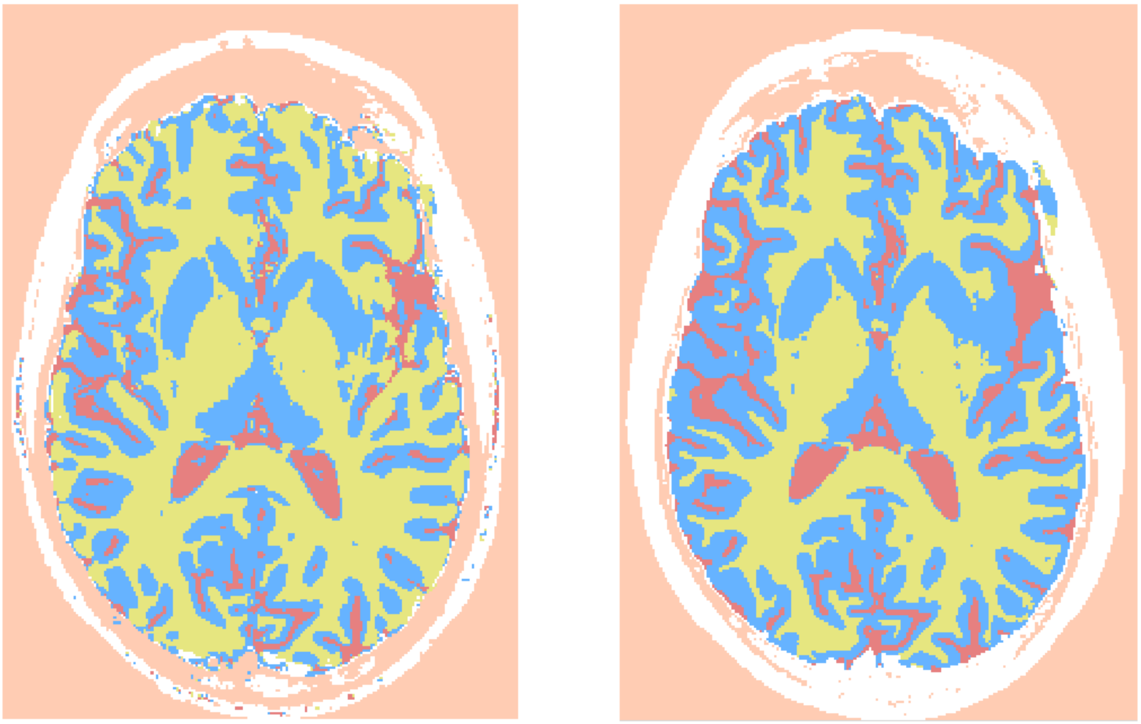


Figure B.1: Axial view of the segmentation with different sampling methods

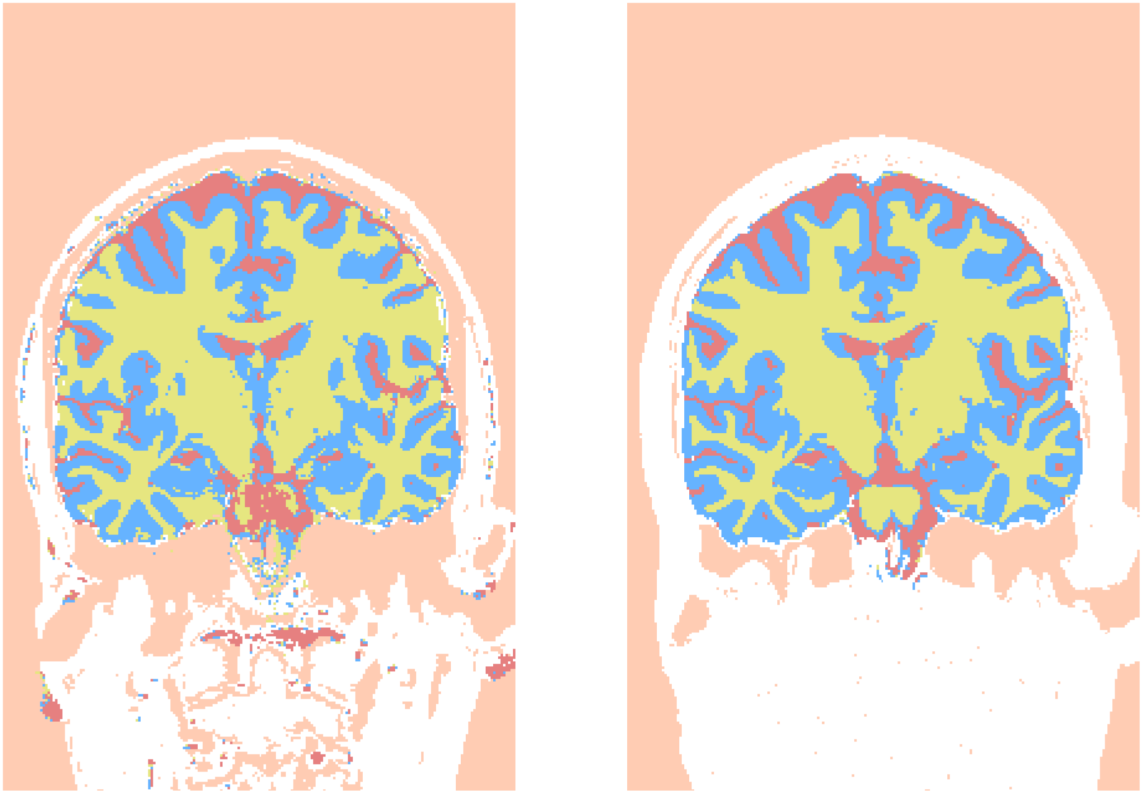


Figure B.2: Coronal view of the segmentation with different sampling methods

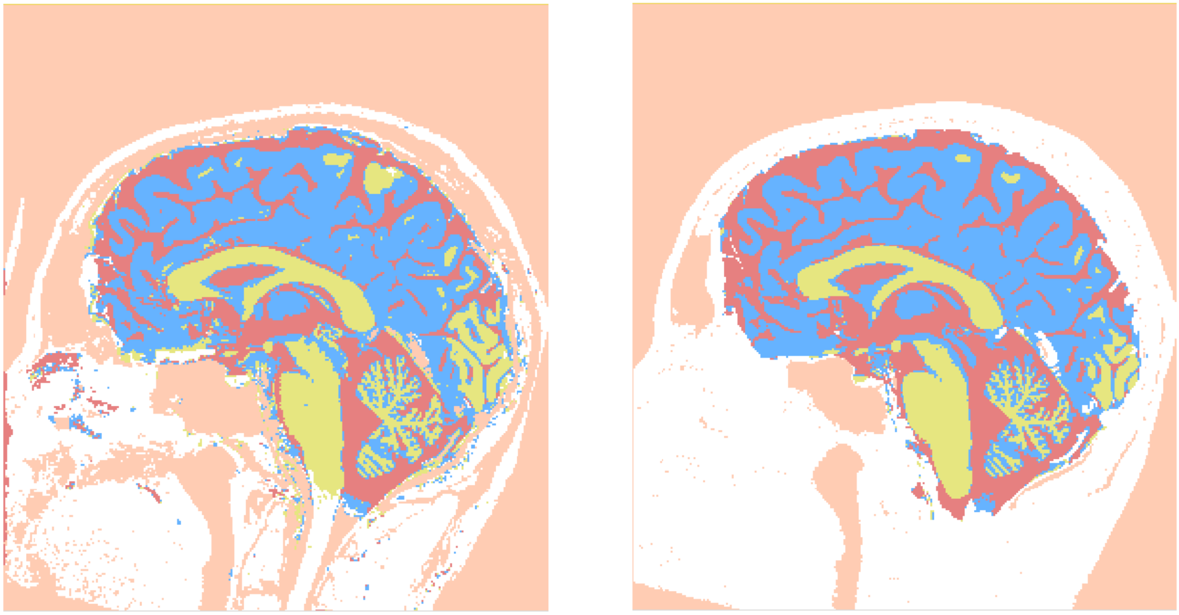


Figure B.3: Sagittal view of the segmentation with different sampling methods