LILLE - BREST - TOULON

isen

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**

Parameter space exploration and tools for fast visualization in EM segmentation and MRI bias field correction in Slicer 3

From April 6th to Septemner 11th
At
Surgical Planning Laboratory (SPL)

SPL

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

Supervisors:

- Ron KIKINIS, Surgical Planning Laboratory, Harvard Medical School, Boston, MA - USA

- Sylvain JAUME, CSAIL, Massachusset's Insitut of Technologies, Boston, MA - USA

Referring teachers:

- Dominique MARATRAY, Institut Superieur de l'Electronique et du Numerique, Brest - FRANCE

- Christine CAVARO-MENARD, Universite de Angers, Angers - FRANCE

Student:

- Nicolas RANNOU, Institut Superieur de l'Electronique et du Numerique, Brest

Abstract

A couple of sentences on three or four lines to summarise your work.
This is a LaTeX template for undergraduate project reports.
Its detailed contents evolve to reflect FAQs.
Expectation-maximization is very popular for segmentation but it can be tricky to understand and to use. A full description of the EMS algorithm is done in this report. Different methods for fast parameters exploration are described. As part of the research, preprocessing methods like MRI bias field correction will be explained. The results obtained will be presented. Following the new workflow should allow the user to segment more datasets, more accurately.
Keywords: segmentation, expectation, maximization, correction, bias.

Resumé

Quelques phrases pour resumer mon travail.
C'est un template LaTeX pour les rapport.
Le contenu peut evoluer.
Mots clés :segmentation, expectation, maximisation, correction, biais.

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 the physicist from errors and to help him to proceed to more precise moves. After the surgery, to see if the surgery succeed, or to follow the pathology of a patent. Nevertheless, the informations bring to the physicist by the tool must be accurate, robust and provide a fast feedback.

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

For the MR images segmentation purpose, the Surgical Planning Laboratory (SPL), Harvard Medical School, has developped an EM algorithm to segment brain's MR images. The results obtained are very good until we segment small structures and we select the optimum parameters. The approach used for the intensity inhomogeinities estimation appears less efficient for the particuliar purpose of large structures segmentation WHY. Moreover, the implementation is not widely used so far, regarding the complexity of the segmentation process. In this report we will present an approach to

enhance the segmentation for large structures, correctiong the intensity inhomogeinities in large structures and providing the end-user tools for an easier 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 will be described. We will also present in this chapter the limitations of the current implementation. Chap. 3 describes our contribution. It explains the solution we choose to enhanceme the segmentation and to improve the usability of the current framework. Chap. 4 shows the results achieved. It aslo discussed about what have been done, the limitations of the current module 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, applied to brain segmentation and show firstly a simple approach of the problem followed by a more realistic approach of the problem with different constraints. 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 particular 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 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.

---

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

2.2 Fundamentals

Here we get going with a description of the fundamentals. We begin with a description of the statistical model used for the brain then present all the related elements to have a good understanding of the EM Segmentation algorithm in the next section. 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 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 describe 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.So, the conditional probability function describing Y_i is $p(Y_i|Z_i, \Phi)$.

The easiest model assumes that each intensity in one class is 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 kth tissue and n_i a random sample generated by the corrupting factor(s). 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 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:

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

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:

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

7

$p(Z_i = k|\Phi)$ 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 this simple approach, we will assume that the *prior probability* is constant. The new model we obtain 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 Gaussian (eq. (2.1)). Moreover, we will assume that *prior probability* (eq. (2.2)) 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 $\Phi = (\mu_1, \sigma_1, c_1, ..., \mu_n, \sigma_n, c_n)$.

Using equations (2.1) and (2.3), equation (2.2) becomes:

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

In the case of *Gaussian mixture model*, each voxel is considered to be independent. That means that each voxel will have his 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. The next step is then to find the set (μ_i, σ_i, c_i) of parameter Φ for each voxel, to fit as well as possible the normalized histogram. A convenient way to find it is to use the *Maximum likelihood* principle.

2.2.3 Maximum likelihood

In our case, we know the intensity of each observed pixel y_i and Φ has to be found. $p(Y = y_i|\Phi)$ (eq. (2.3)) is called likellihood function and for each Φ, it returns the value of the likelihood for y_i. We can generalize it to the whole image with $p(Y|\Phi)$.

As we said in SDFSDF, the voxels are considered to be independent. It leads us to:

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

The objective is to find the parameter Φ which will maximize the likelihood of the image. 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 sum. Equation (2.6) becomes:

$$\hat{\Phi} = \arg\max_{\Phi} \log p(Y|\Phi) = \arg\max_{\Phi} L(\Phi) \tag{2.7}$$

where:

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

Finally, in case of GMM, with eq. (2.4) and (2.2), $L(\Phi)$ becomes:

$$L(\Phi) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} G(y, \mu_k, \sigma_k)c_k \tag{2.8}$$

The maximized *log* likelihood can be computed using partial derivatives for each paraemter of Φ. When the partial derivative is 0 for a parameter, we found the maximum likelihood for the parameter. For example, to find the maximum likelihood for μ_k, we have to find when:

$$\frac{\partial}{\partial \mu_k}(L(\Phi)) = 0 \tag{2.9}$$

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

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

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

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

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

Using Bayes formula, we notice that:

$$p(Z_i = k|Y_i = y_i, \Phi) = \frac{p(Y_i = y_i|Z_i = k, \Phi)p(Z_i = k|\Phi)}{\sum_{j=1} p(Y_i = y_i|Z_i = j, \Phi)p(Z_i = j|\Phi)} \tag{2.10}$$

Thus, setting the denominator to 0 yields to:

$$\sum_{i=1}^{n} p(Z_i = k | Y_i = y_i, \Phi)(y - \mu_k) = 0 \qquad (2.11)$$

Let us denote

$$p_{ij} = p(Z_i = j | Y_i = y_i, \Phi) \qquad (2.12)$$

Equation (2.11) leads us to:

$$\mu_j = \frac{\sum_{i=1}^{n} y_i p_{ij}}{\sum_{i=1}^{n} p_{ij}} \qquad (2.13)$$

Proceeding the same way as we did for equation (2.11), we can get similar equations for variance σ_j and weight c_j.

We find that:

$$\sigma_j^2 = \frac{\sum_{i=1}^{n} (y_i - \mu_j)^2 p_{ij}}{\sum_{i=1}^{n} p_{ij}} \qquad (2.14)$$

$$c_j = \frac{1}{n} \sum_{i=1}^{n} p_{ij} \qquad (2.15)$$

Equations (2.13),(2.14), and (2.15) provides us an equation for soft segmentation.

$$p_{ij} = \frac{G(y_i, \mu_j, \sigma_j) c_j}{\sum_{k=1}^{K} G(y_i, \mu_k, \sigma_k) c_k} \qquad (2.16)$$

The segmentation can now be done following an iterative algorithm called *expectation maximization algorithm.*

2.3 Expectation maximization algorithm

The EM algorithm is a method to find the maximum likelihood for a given set of parameter (μ, σ and c in our case). Here we first get going with an intuitive description of the algorithm 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 directly (in the case of GMM). The EM algorithm is an iterative process of two steps: the expectation step (E-Step) and the maximization step (M-Step).

E-step

In this step, we calculate an estimation of soft segmentation $p^{(m+1)}$ with equation 2.16 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 for iteration.

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

M-step

In this step, we estimate the maximum likelihood for parameter $\Phi^{(m+1)}$ with equations (2.13),(2.14), and (2.15) 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.

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

$$(\sigma_j^{(m+1)})^2 = \frac{\sum_{i=1}^{n} (y_i - \mu_j^{(m+1)})^2 p_{ij}^{(m+1)}}{\sum_{i=1}^{n} p_{ij}^{(m+1)}}$$

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

EM algorithm iterates until convergence is reached. As discussed in [7], convergence is assured since the algorithm is guaranted to increase the likelihood at each iteration. The problem is simple as long as we are working with GMM. In the other case, the log-likelihood can not be maximized by direct differenciation and a more generalized approach must be used.

Generalized algorithm

Now we assume that we are no longer working with GMM. 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:

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

Since log is a strictly increasing function, the value of Φ which will maximizes $p(Y|\Phi)$ will also maximizes $L(\Phi)$. We want to maximize $L(\Phi)$. Thus, after the nth iteration, we want an estimated Φ_n such that:

$$L(\Phi) > L(\Phi_n)$$

In other words, we want to maximize the difference $L(\Phi) - L(\Phi_n)$. We can transform this difference as below:

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

We can deduce this inequality from Jensen's inequality since $p(Z_i = k|Y_i = y, \Phi_n)$ is a probability mesure:

$$p(Z_i = k|Y_i = y, \Phi_n) > 0$$

$$\sum_{k} p(Z_i = k|Y_i = y, \Phi_n) = 1$$

Finally,

$$
\sum_{k=1}^{K} p(Z_i = k|Y_i = y, \Phi_n) \log \frac{p(Y_i = y|Z_i = k, \Phi) p(Z_i = k|\Phi)}{p(Z_i = k|Y_i = y, \Phi_n)} - \log p(Y|\Phi_n) = \sum_{k=1}^{K} p(Z_i = k|Y_i = y, \Phi_n)
$$

$$\triangleq \Delta(\Phi|\Phi_n)$$

2.4 Expectation maximization algorithm used in Slicer 3

In this part, we present the algorithm which has been intergated in Slicer 3, and discuss of the limitations of this one. Some informations has been added to the algorithm to get the best and most automatic segmentation as possible.

2.4.1 Spatial information

2.4.2 Structure information

2.4.3 Intensity inhomogeneities information

Discussion

2.5 Nothing

Each chapter should end with a round-up of its contents and a link with the contents of the next.

After formulas, equations and theorems, the next important topics are graphics and tables.

Chapter 3

The contributions

Expectation-maximization is a very popular framework for different classification problems. It became extremly popular in segmentation of brain MRI during the last decade. However, it is not easy to understand the underlying theory.

In this chapter, a very simple approach of the theory will be given.

Finally, some solutions will be presented. Graphics files are inserted via the package `graphicx` which is loaded in the document preamble (in file `report.tex`).

Here we deal firstly with the inclusion of a single unlabelled diagram, and then with captioned, labelled, and multiple diagrams.

After that we consider tables of information.

3.1 MRI Bias Field correction

Here is a colour `.jpg` file, inserted in place without caption and without a number for cross-reference. It's suitable only for a simple self-explanatory diagram, used here and not again.

Bias correction is not efficient so we propose an other method, as part of the segmentation process.
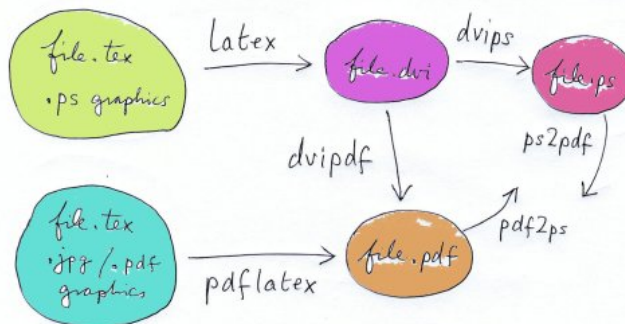Rescaling

3.1.1 Interest

Kilian algorithm is only accurate for small region bias correction, because au the gaussian model he uses. Correction the bias in the whole volume first could enhance results.

3.1.2 Our approach

Gaussian distrib.... ITK filter Automatic

3.1.3 Results

corrected not corrected parameters explanation

It shows the normal routes from a LATEX source (`.tex` file) to `.ps` or `.pdf` output depending on the nature of the graphics files included[1]. *Confusion over these routes is a frequent source of grief.*

This template has diagrams and graphs in `.jpg` format[2] and so its source is compiled to a `.pdf` file with the command or menu button `pdflatex`.

Mathematical software such as *Maple* and *Matlab* can produce either `.ps` or `.jpg` diagrams, while cameras and scanners[3] usually produce `.jpg`. If accurate detail is vital, `.ps` is best.

With care, you can mix some graphics formats within a document. For instance, if you compile to `.pdf` then you can use an arbitrary mix of `.jpg`, `.png` and `.pdf` graphics, but not `.ps`. And a document to be compiled to `.dvi`, including `.ps` graphics, can include other formats too if they are explicitly given '*bounding boxes*'. But it's relatively tricky and you're better-off converting everything to one format.

Note that you can't include `.bmp` or `.gif` files at all, but conversion of graphics files — from `.bmp` to `.jpg` for example — is easily done with software like *PhotoShop* [PS] or *ImageMagick* [IM].

It's simplest to keep your image files in the same directory as your `.tex` files. Otherwise you can explore the mysteries of `\graphicspath` [MG, Sec. 10.2.5]. In any case, use only forward slashes in a directory specification. Names of directories and of graphics files *must not* include spaces!

---

[1]An IDE such as TeXnicCenter has menu buttons for each part of each route.

[2]File extensions must be `.jpg` and not `.JPG` or `.jpeg`.

[3]The figure itself is hand-drawn and scanned.

3.2 Intensity Normalization

3.2.1 Interest

usefull if you need to do a lot of segmentations enhance the rescaling..?

3.2.2 Our approach

background/brain

3.2.3 Our implementation

show results... corrected image...

It's awkward to flow text round a picture, but it can be done with packages such as `picinpar` or `wrapfig`, as *The Companion* explains [MG, Sec. 6.4.1].

And here's a `.jpg` image side-by-side with some explanation, using the `minipage` environment.

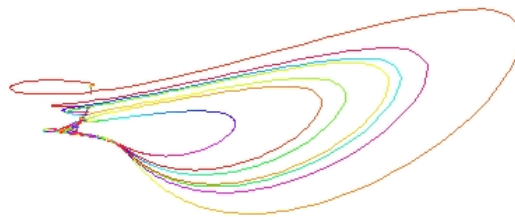Something more elaborate sometimes gives unexpected results.

Figure 3.1: This is a caption explaining the diagram clearly and fully.

That is, a picture 'floating' in a `figure` environment — with a caption and number as in Fig. 3.1 — might be placed by LaTeX on a page well after the text meant to accompany it.

This happens if you have a relatively high density of graphics to text. You may be able to deal with it by re-distributing the pictures, changing the size of some of them, and the careful use of paragraph breaks. Otherwise you may resort to `\clearpage`, which forces insertion of floating objects waiting to go in.

The *Not so Short* book [NSS] has Sec. 2.12 on the issue of 'floating bodies' and *The Companion* has a whole chapter [MG, Chap. 6] on 'mastering floats'.

Note that inside a `figure` (or `table`) environment the `\label` must always *follow* the `\caption` — see [MG, p. 67].
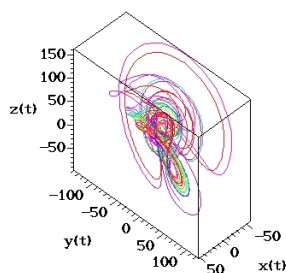

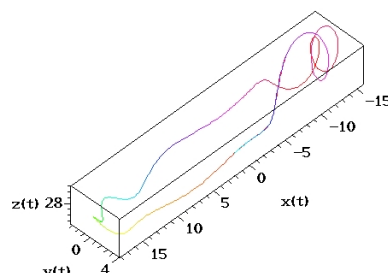Figure 3.2: Caption to explain the diagram clearly and fully.

Figure 3.3: Caption to explain the diagram clearly and fully.

Fig 3.2 and Fig. 3.3 are two pictures floating side-by-side, each in a `minipage` and each with its own caption and label. This can be an economical way to insert multiple diagrams.

But don't get carried away and cram too many pictures together. Be aware of the size-reduction that's often necessary, and maintain legibility by choosing large enough font and weight for symbols, axis-labels and other text.

Figure captions should give plenty of information, because readers will first skim the Introduction and Conclusions — and then look at the pictures. If you want (really really want) to hook them, make sure that every important figure plus its caption is as self-explanatory as possible.

3.3 Class Distribution selection

During parameters initialization, the user has to define each class distribution. The current methods presents some troubles so we present here a new approach to estimate each class distribution.

3.3.1 Interest

So far, the user haf two choices to define each class distribution. He could enter 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 for the user to found the good mean value and variance for each class for each volume. Morever, each time he wants to process a new volume, he will have to redefine mean values and variances. This is not convenient and it can takes a ot of time. 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. With this methos, the problem is that you compute your mean value and variance using a few samples. Then, your mean values and variances are not accurate. That's why we proposed a new approach using a label map, to estimate each class model.

3.3.2 Method used

Create a label map. colors we choose a the same as the one we defined for each class. then run algorithm which will estimate the mean value and covariance values.

Algorithm..?

Explain each colors

3.4 Class Distribution visualization

An important contribution is a tool which allows to visualize the classes to be segmented distribution, through 2 volumes.

3.4.1 Interest

As we discussed in the EMS algorithm requires variance and mean values for each class to be segmented. So far, the user defines these parameters by selecting samples on the dataset. exampleSDFSF A main issue is that user user can know if the pixels he selected are really representative or not of the class he wants to select. Thus, our objective was to find a way to give the user if his samples are reprenstative of not of a tissue.

3.4.2 Method used

Problem's description

Here is a brief description of the situation to let you understanding the problem. As input we have N images, K classes to segment, and for each class a mean value μ, a variance σ and covariance ζ. The covariance is
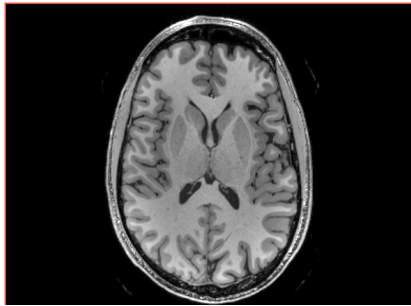
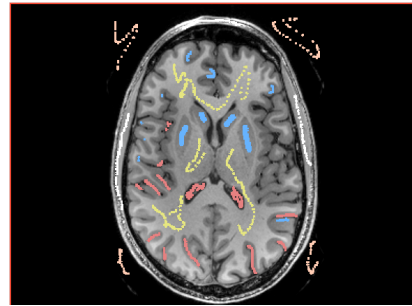Figure 3.4: Axial view of a T1 volume without the label map.

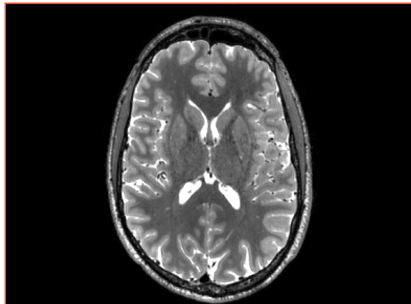Figure 3.5: Axial view of a T1 volume with the label map.

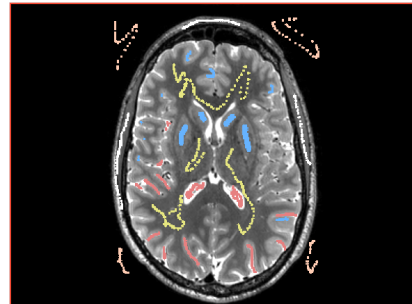Figure 3.6: Axial view of a T2 volume without the label map.

Figure 3.7: Axial view of a T2 volume with the label map.

estimated for each couple of images. We will come back on it later. where Using these informations, we had to find a way to give the user usefull informations about the class distribution.

Choices

Our first idea was to display gaussian curves in 3D, for each class. The X and Y axes would be the intensity ranges for the to images we want to use. The problem with this representation is that it does not contain any information about the number o f pixels . 3d too sophisticated...? To solve this problem, we decided to divide the problem in two parts. The first part will consist in creating a "temperature map" as background. The second part will consists in displaying ellipses over this map. This will give the user a good information about how good his selection was.

Background creation

First of all, the background has to be created. This background is a "temperature map". We can present it as a two dimensions array. The height and the lenght of the array depend of the intensity range of the 2 volumes we want to process. Then, we fill the array in an easy but useful way. Each case of the array contains the number of occurence of an intensity pair at the same position, through the 2 volumes.

Algorithm 1: BACKGROUNDCREATION(V1,V2)

---
define RangeI1 ← intensity range in V1
define RangeI2 ← intensity range in V2
define A(RangeI1,RangeI2) ← background
for each pixel **i** in V1
for each pixel **j** in V2
▷A(i,j)=A(i,j)+1
return A

Ellipse representation

We will first start we a brief rappel.. The covariance matrix is
Representing ellipse associated to this covariance matrix.
Maximum variance axe.Eigen values
Algorithm used. Big and small axes of the ellipse

3.4.3   Results

show result
    First a simple table . . .

row 1	it's	just	as	easy
row 2	as	this	$E = mc^2$	what
row 3	could	be	simpler	?
row 4	as	easy	as	Π

Here it is again, now floating in a `table` environment, with a caption and a label to identify it as Table 3.1. The remarks about giving full information

row 1	it's	just	as	easy
row 2	as	this	$E = mc^2$	what
row 3	could	be	simpler	?
row 4	as	easy	as	Π

Table 3.1: Floating table, with a caption to explain it clearly and fully.

in figures and their captions apply equally to tables, of course. So do the remarks about unexpected placement of floating objects.

If you need a long table, occupying more than a page, and which can't reasonably be divided, then you need to delve into *The Companion* [MG, Chap 5] to find the secret.

3.5   Global Prior Estimation

The last contribution to the EMS is a tool which provides the user an easy and fast way to estimate the global prior weights (GPW).(ref ch2 ...)

3.5.1   Presentation of the problem

This contribution is usefull in many different ways. When you run the segmentation process,at the 6th step of the process, you have to provide to the algorithm an estimation of the GPW for each node in the tree. First of all if there are a lot of structures to segment, they user can spend a lot of time during this step. Indeed, for each part of this tree strcuture, they have to define the GPW. Moreover, the user may not know at all which weights to choose. This new approach will provide the users a good estimations of the weights to use. We must also keep in mind that the end users are physicists. They might don't understand what the parameters meanings and providing them a visual feedback could help them a lot.

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 GPW estimation. The second part will consist in developing an algorithm which fills automatically the tree.

Fast user feedback

We can divide the feedback part in 3 steps: the histogram computation and utilisation, the multicolumn list and the labelmap generated. The histogram allows the user to manual segment classes based on intensity. The multicolumn list allows the user to change the order of the classes in the histogram. The labelmap provides to the user a visual feedback, base on the segmentation realized in the histogram. Using these three complementary tools, the user, even if he is not initiated can estimate easily, accurately and rapidely the GWP.
schema

Global priors evaluation

The algorithm used to estimate the weight of each node is iterative. It starts from the root and goes to the leaves. It evaluates the weight of the childs of the active node at each iteration. Here is a description of the algorithms used to compute the GPW of each node.
DEscirption algo 1

Algorithm 1: TREEWEIGHTESTIMATION(R, W)

---

define C = CHILD(R) ← set of childrens of root R
define LEAF(C) ← set of leaves of tree with roots C
define H ← set of structure-specific information defined by LEAF(C) for each leaf
update W in childrens of root R with the results of WEIGHTESTIMATION(C,LEAF(C),H)
for each node R' in CHILD(R) that is not a leaf
▷TREEWEIGHTESTIMATION(R', W)

Descripton algo2
The algorithm used estimates the global prior of the leaves of the current node, based on the number of pixel which belong to the child classes. This number of pixels is calculated from the segmentation computed in the histogram.(CF ..)

Algorithm 2: WEIGHTESTIMATION(C,LEAF(C), W,H)

---

define T ← set of total weight of leaves in LEAF(C). Leaves weights are contained in H
define E ← set of weight for each node of C
for each node of C
▷E=E+H : Get the total weight of each node
W=E/T
return (W)

3.5.3 Results and discussion

As we can see, the results obtained are good and accurate.The time of processing is fast and most of the people are now able to understand these parameters, even they are not familiar with EMS. This is an intuitive way to get an estimation of the GPW parameters.

Of course some sfsdfs must be done. If there is a strong bias in the images we process, the pixels of a same class will have different values. Then, a segmentation based on intensity will provide bad results. Moreover, if the user wants to segment 2 classes which have the same color, just using the spatial information for example, they won't be able to use it properly.Thus, the user must always keep in mind that it is just an estimation and that he has to check if the values are accurate.

Chapter 4

Results and discussion

This chapter will review what has been done and mentions the main open questions.

    Results using different tools

    limitations

    improvement (in relation to limitations)

4.1 Results

Tests has been performed to show the utility of the work done. The results are presented below. The testings are comparaisons between the results obtained with the previous workflow and the new one. The results obtained will then be reviewed by a specialist to evaluate the previous and the new segmented datasets.

4.1.1 Bias correction

Here we get going with the intensities inhomogeinities correction.

Testing process

we create a mask.

Results

Here are the resultsfdd dddddddddddd dddddddddddd dddddddddddd dddddddddd dddddddddd dddddddddddd dddddddddd dddddddddd dddddddddd dddddddddd dddddddddd dddddddd dddddddd dddddd dddd ddddddddd..

    more comments tttttttttttttttttttttt tttttttttttt tttttttttttt tttttttttttt tttttttttttt tttttttttttt ttttttttttttttttt tttttttttttttttttt tttttttttttttttttt tttttttttttttttttt ttttttttttttttttttttttt
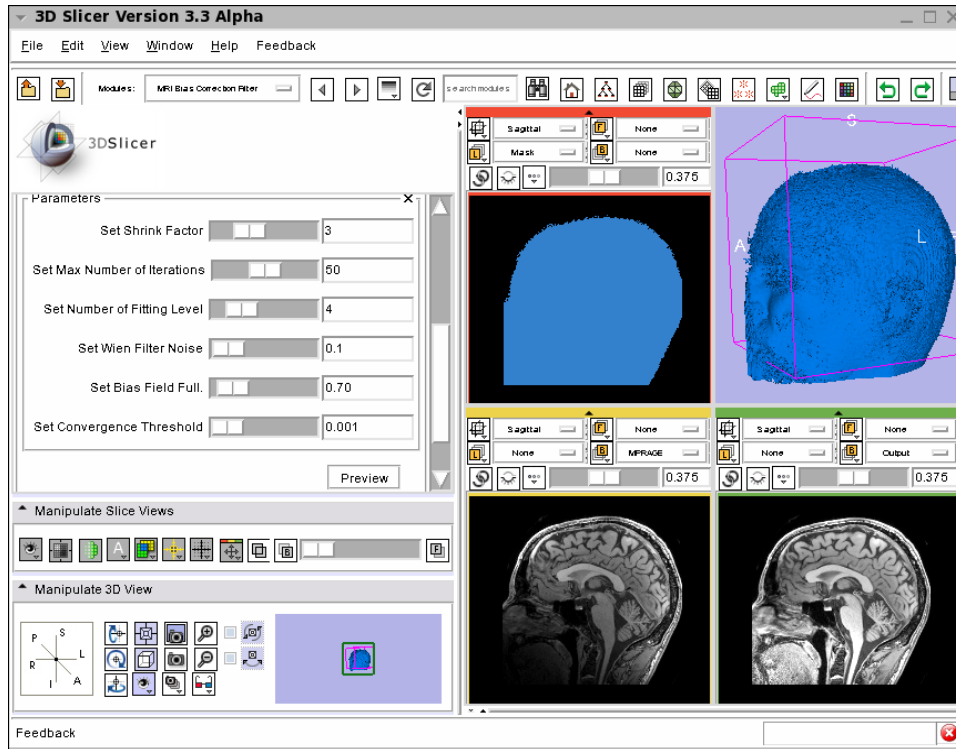
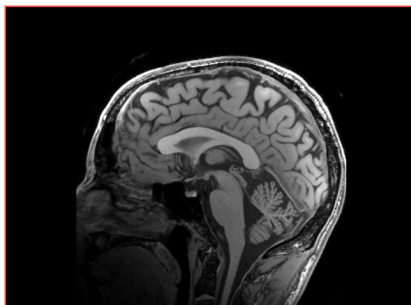Figure 4.1: Module created for the bias correction.

Figure 4.2: Sagittal view of a biased T1 volume.

Figure 4.3: Sagittal view of the T1 volume after bias correction.

Specialist's point of view

4.1.2 Global Prior estimation

Testing process

Results

Specialist's point of view

4.1.3 Class Selection

Testing process

Results

Specialist's point of view

4.2 Limitations

4.3 Future work

Acknowledgements

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.wiki.org/wiki/Expectation-maximization_algorithm*, June 2009.

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

[AC]   A Cottrell, *Word Processors: Stupid and Inefficient*,
                          `www.ecn.wfu.edu/~cottrell/wp.html`

[BR]   Visit `www.dur.ac.uk/library/using/guides/` and click on '*Writing your bibliography and citing references*'.

[ESL]   L Truss, *Eats, Shoots and Leaves*, Profile Books 2003
                          (ISBN `1-86197-612-7`).

[GRM] M Goossens, S Rahtz and F Mittelbach,
                The LATEX Graphics Companion, Addison-Wesley, 1997
                          (ISBN `0-201-85469-4`).

[IM]   *ImageMagick*, `www.dur.ac.uk/its/software/application/...`
                                 `...?application=ImageMagick`

[LAT] *LaTeX stuff*, `maths.dur.ac.uk/Ug/projects/resources/latex/`

[MEM] *Memoir document class*,
           `www.ctan.org/tex-archive/macros/latex/contrib/memoir/`

[MG] F Mittelbach and M Goossens *et al.*, *The LaTeX Companion*,
                 Addison-Wesley, 2nd ed. 2004 (ISBN `0-201-36229-6`).

[MKT] *MikTex Project Page*, `www.miktex.org`

[NSS] T Oetiker, H Partl, I Hyna and E Schlegl,
               The Not So Short Introduction to LaTeX 2_ε,
               `www.ctan.org/tex-archive/info/short`

[PS]   *Photoshop*, `www.dur.ac.uk/its/software/application/...`
                            `...?application=Adobe+Photoshop`

[TXC] *TeXnicCenter*, `www.toolscenter.org`

[WDT] *WinEdt*, `www.winedt.com`

[WL]   *Wikibook on LaTeX*, `en.wikibooks.org/wiki/Latex`

[WO]   *Controlling widows and orphans*,
            `www.tex.ac.uk/cgi-bin/texfaq2html?label=widows`

[WSH] *WinShell*, `www.winshell.de`

Prepared in LaTeX 2_ε by RCJ

Appendix A

About references

Your report will be put together in your own style, mostly using your own words. Much of it will be standard material that you've read and digested, but you may have a fresh example, application or calculation that you've done yourself.

You must make clear what's not your own work by referring suitably to your sources — books or articles or web-pages, etc. Not to do so can count as plagiarism, which is cheating.

This appendix aims to amplify the advice given in the Library's guide [BR], which you should read.

A.1 Organisation

Follow the style of this template — that is, put an orderly list of your sources ('*The Bibliography*') at the end of the main text, and refer to ('*cite*') items on the list by suitable codes placed appropriately in the report's text.

For your bibliography the golden rule is that each listed item must give enough detail to allow readers to follow it up themselves and find the precise part of the book, article, web-site or whatever without ambiguity or delay.

For instance it's no use saying just '*The Times newspaper*' unless you also give the date and page, or just '*Wikipedia*' unless you give the complete URL of the specific page, ... and so on.

It's always better to give slightly too much information — e.g. you might include the ISBN of a book too. A good level of information is exemplified here and recommended in the Library guide [BR].

If you need to cite different parts of e.g. a book for different things, then list it (say AB) in the bibliography and give the different citations as [AB, page 32] and [AB, Sec. 4.7] and so on. The LaTeX \cite command allows for this (look it up!).

Now you see why a bibliography list is better than lots of footnotes — it neatly allows such multiple citations. Too many footnotes make a mess.

- This bibliography uses short letter-code keys in alphabetical order. It's one of several standard possibilities [BR] and is often preferred for economy and because the letter codes[1] can be chosen to be helpful mnemonics. Numerical codes, for instance, can't.

- Long lines (with URLs usually [IM, PS]) may need to be split. The LaTeX hack used here to manage such line-breaks may not appeal to everyone.

A.2 Doing it

What things need a reference? The answer is — anything you didn't work out or invent yourself, but took from someone or somewhere else. You may do that, so long as you say so *and also make it clear you understand it.* Don't copy blindly! Here are some examples.

- '*the following explanation is taken from [LH]*' — if you've copied word-for-word from the article by Laurel & Hardy, which you list as item LH in the bibliography. Direct quotation should be used sparingly, and the text emphasised — perhaps by use of the `quotation` environment in LaTeX.

  Don't be tempted to copy from a book, or `ctrl-C`/`ctrl-V` from the web, *without* clearly admitting it. It's easy to detect, and it's suicide.

- '*the following proof is in many textbooks, eg. [LH, page 16]*' — if it is, and you can't think of a different proof, except perhaps for notation.

- '*the material in this section is based on Sec. 2.6 of [CL] and Chap. 3 of [SH]*' — if you've combined into your own words the account by Cagney & Lacey with that by Starsky & Hutch.

- '*the following proof is adapted from Chap. 4 of [SJ]*' — if (say) you've filled in the gaps in Smith & Jones' proof, or perhaps changed it from the case of general n to your case $n = 2$.

- '*these calculations were done with Matlab [MAT] using the m-file listed in App. C*' — where the reference MAT is to the *MathWorks* website.

- '*these calculations were done with the TISEAN package [TIS]*' — if you've downloaded this specialised package from a web-page whose URL (and author's name, if available) is item TIS in the bibliography.

- '*the graph is taken from Morecambe & Wise [MW, page 16]*' — if you've scanned the figure from their book. Similarly, if you've downloaded a `.jpg` file from a web-site, then give in the bibliography the

---

[1]Note the punctuation of 'letter-code' and 'letter codes' here.

URL and author (if known). Such citations could go either in the figure caption or in the associated text.

- '*the data are taken from [XY]*' — where XY gives the source of the numbers you've analysed. This might be a journal or an online data-bank, *etc.* Generally the numbers themselves should be left out — a summary table or a graph or two (plotted with *R* or *Matlab* or *Maple*) is often enough. Occasionally raw data might be supplied separately — e.g. on a CDrom.

Often your supervisor or someone else gives you something — such as a set of data, or a useful *Maple* worksheet, or help with a proof — when the appropriate form of bibliography item is '*Dr I Newton, private communication, April 2007*'. Similarly for printed course material — '*Dr I Newton, lecture notes for module MATH5033, Durham University, Epiphany Term 2007.*'

Generally, don't refer to things you haven't read. Textbooks may well cite the Serbian-language journal where the result you want was originally published. But you are not in the business of ascribing credit for discovery. In *your* report *you* must cite the book where *you* actually got it from and not give any possibly false impression that you fluently read technical Serbian.

Likewise, you may want to quote from Euclid or Archimedes. But cite the place where you found the English words with something like '*Archimedes said, 'Eureka!' (as quoted by [AB])*'.

Summarising — tell the truth (where you got it from), the whole truth (give full details), and nothing but the truth (you don't read Serbian).

Finally, if in doubt — ask your supervisor.

Appendix B

Long proofs

An Appendix is a good place to put lengthy proofs that must be included but would impede the flow if placed in the main text.

B.1   Proof of theorem 2.4

By inspection.

Appendix C

Computer programs

An Appendix is the place to list computer programs that you've written, using the `verbatim` environment [NSS, Sec. 2.11.4].

For example —

```
10 PRINT "HALLO SAILOR!"
20 GO TO 10
```

Appendix D

Using a PC

You may want LaTeX on your own computer.

A popular version of TeX/LaTeX for a Windows PC is '*MikTex*' [MKT]. *MikTeX* is free, and is available online for download [LAT] or locally on a DVD from `bob.johnson@dur.ac.uk`, room CM315. It's also installed on the ITS Networked PC Service under `Programs | Miscellaneous`.

To use *MikTeX* easily you need a dedicated editor or IDE[1]. A good one is '*WinEdt*' [WDT], which runs under all recent versions of Microsoft Windows and integrates well with *MikTeX*. *WinEdt* is free for 31 days; to use it thereafter costs about $40.

Completely free rivals include '*TeXnicCenter*' [TXC], which is provided on the *MikTeX* DVD and also set up on the ITS Networked PC Service under `Programs | Miscellaneous | MikTeX`.

Others include *eg.* '*WinShell*' [WSH], but are untried.

All these editors/IDEs have a familiar style of graphical user-interface with a toolbar and pull-down menus for all the common tasks involved in editing source files, running LaTeX and viewing the results.

---

[1]Computer scientists say 'Integrated Development Environment'.

Appendix E

Typography and grammar

Some errors crop up time and again, and a few of the commonest — mostly relating to use of LaTeX — are mentioned in the body of this template. Some others are given here, aiming to help you avoid them.

Layout errors include starting a sentence with a symbol, or leaving a single symbol that ends a sentence hanging alone on a line. Related to this are '*widows*' and '*orphans*' [WO]. Can you spot any in this template?

Grammar — illiteracy makes good mathematical work look like rubbish. Of course, everyone makes mistakes, but a final version with many small errors in fact signals '*I am ignorant and I just don't care*'. Read, mark, learn, and inwardly digest the wisdom of the great goddess Truss [ESL].

Apart from spelling blunders, the top five usual suspects in recent project reports are (in random order) —

- '*sentences*' with no verb;

- mis-use of apostrophes;

- mis- or non-use of hyphens;

- writing '*comprises of*';

- unpunctuated displayed equations.

A tip for proof-reading — a friend or relative may spot things you've become blind to. And remember in practise their are a massive number of errors in grammer and stile, and so fourth its only to easy to make that you're spell checker on it's own wont find (14 in this sentence).