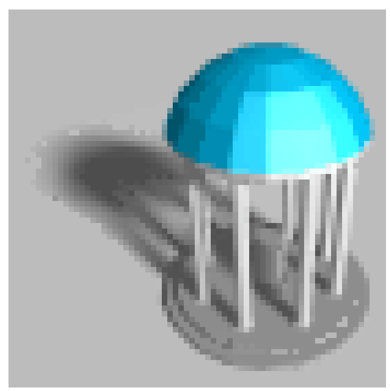
Automatic Brain Tissue Segmentation

Natacha Laurant, Martin Styner, Guido Gerig

Department of Computer Science
Medical Image Display and Analysis Group
University of North Carolina at Chapel Hill, USA



Introduction

- 3D images segmentation is important for :

  - Quantification.
  - Assessment of pathology.
  - 3D visualization for diagnosis, planning and therapy.

- The manual slice segmentation is performed but is :

  - The only method of choice.
  - Time consuming.

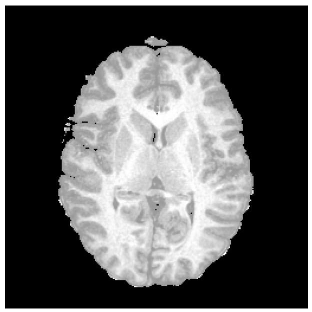
  It uses the dual-echo images.

Aim of the project

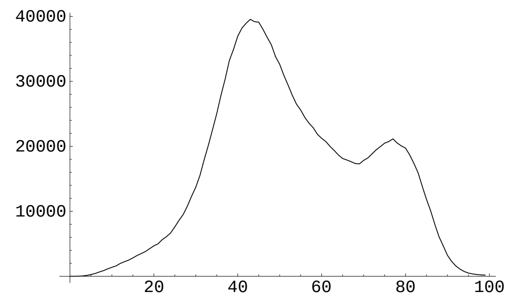
- Data driven segmentation using the intensity values on Gradient echo images.
- Reproducible segmentation (no intra or inter variability).
- Driving application :

  - Volume estimation of Grey Matter and White Matter.
  - Psychiatry : Schizophrenia, Neurofibromatosis 1...

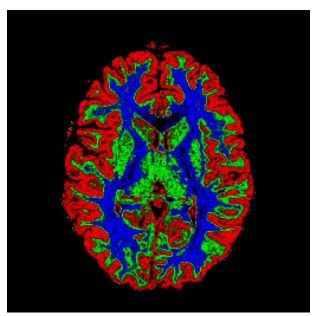
Aim of the project



⟹

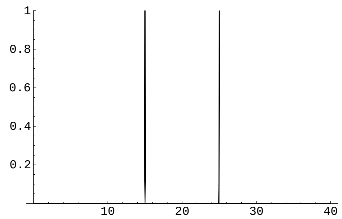


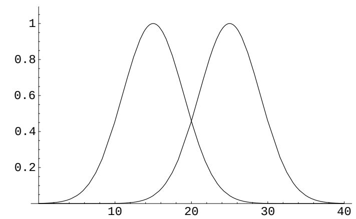
Clusters ↙ ↘ Threshold

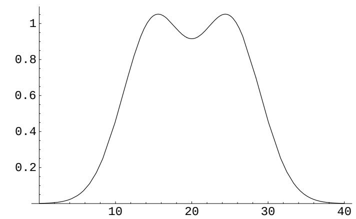


Preliminaries

Example with 2 clusters :



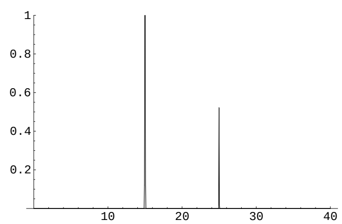


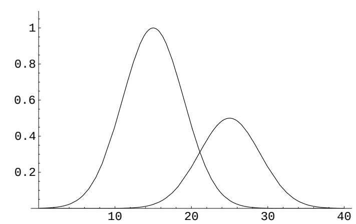


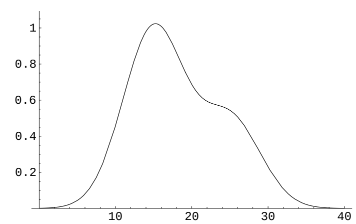
No noise
Homogeneity
No partial voluming

Noise added

Histograms

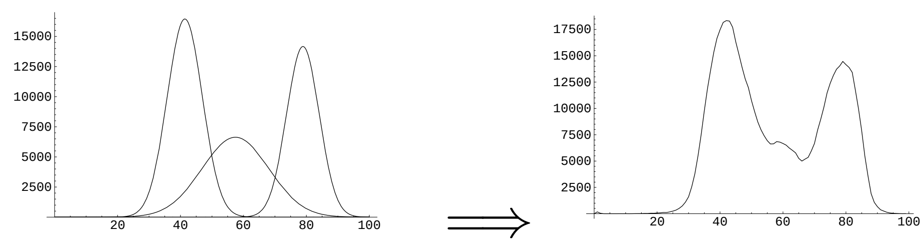






Our model

- Our model is a weighted sum of Gaussians : $G = \sum_i a_i G_i$



- Our problem : To find the weighted Gaussians that explain the histogram

A study in 3 steps

1. How to automatically mask the brain ?

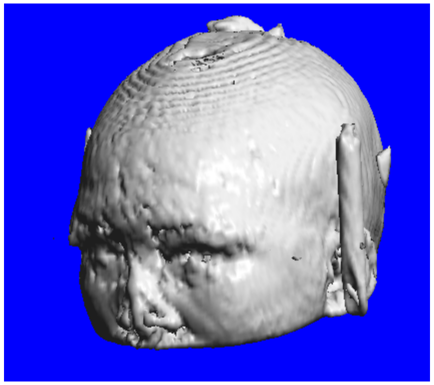
2. Two methods developed to fit the histograms :

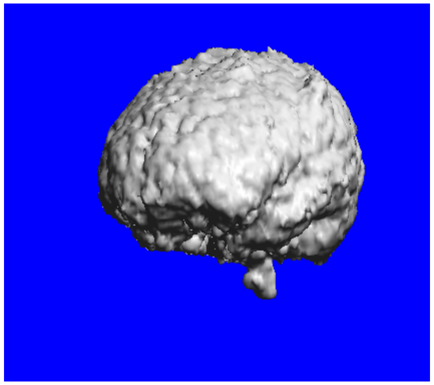
   - Levenberg-Marquardt algorithm
   - $1 + 1$-Evolution Strategy algorithm

3. Threshold estimation.

First step :

Masking the brain



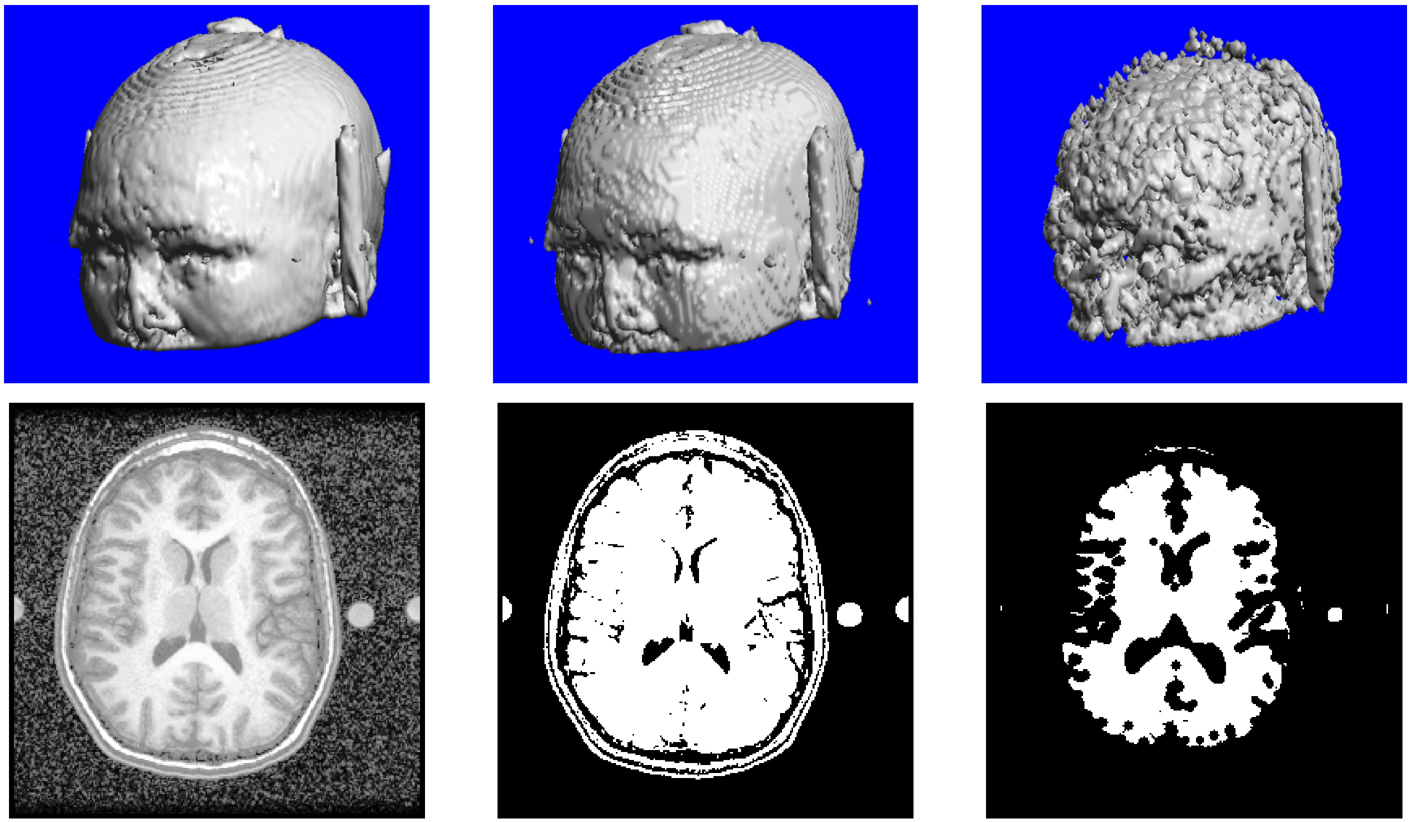


The different steps

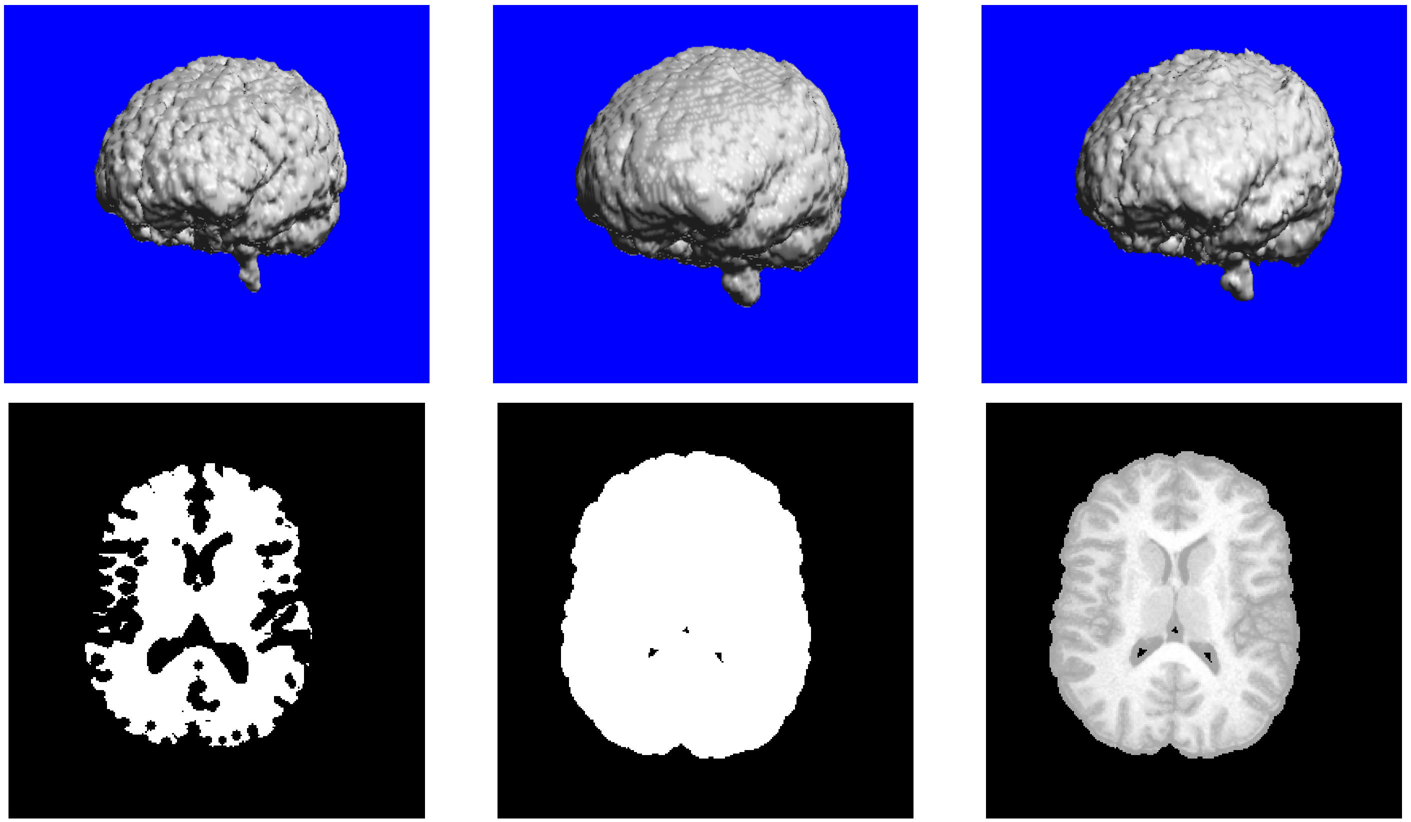
- Thresholding the background
- 3D erosion to eliminate non-brain structures
- Select the largest connected component
- 3D dilation to rebuild the original size
- Masking the original image

How to automatically mask the brain ?

Result on a brain in 3D and 2D

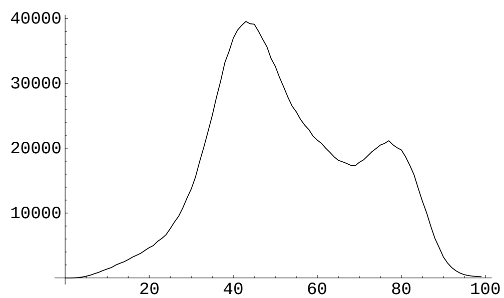


Result on a brain in 3D and 2D

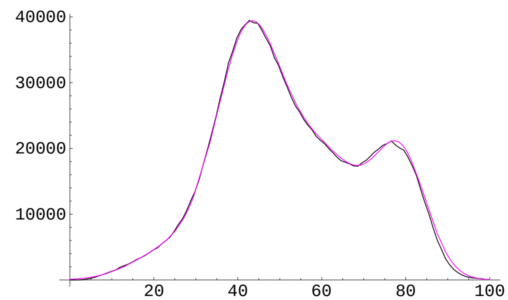


Second step :

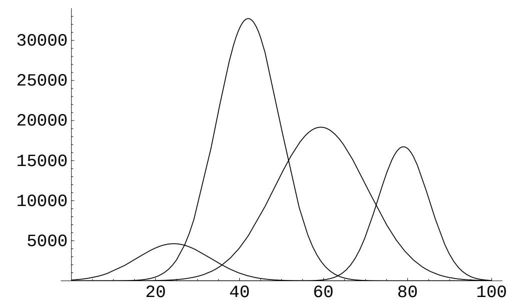
Fitting the histogram



Original histogram



Original and fit



Fit model : Sum of Gaussians

Approach

- Solve non-linear least squares problem
- Search for the best fit :

$$
\begin{aligned}
min\chi^2 &= min\sum_i \frac{[y(i) - f(a, x(i))]^2}{\sigma_i^2} \\
f(a, x(i)) &= f(A, \mu, \sigma, x(i)) = \sum_k A_k e^{-(\frac{x(i)-\mu_k}{\sigma_k})^2}
\end{aligned}
$$

Approach

- Uses the First Derivative of χ^2 with respect to the parameters a(j)
- Uses the Steepest Descent method and, as the minimum is approached, uses the Gauss-Newton method

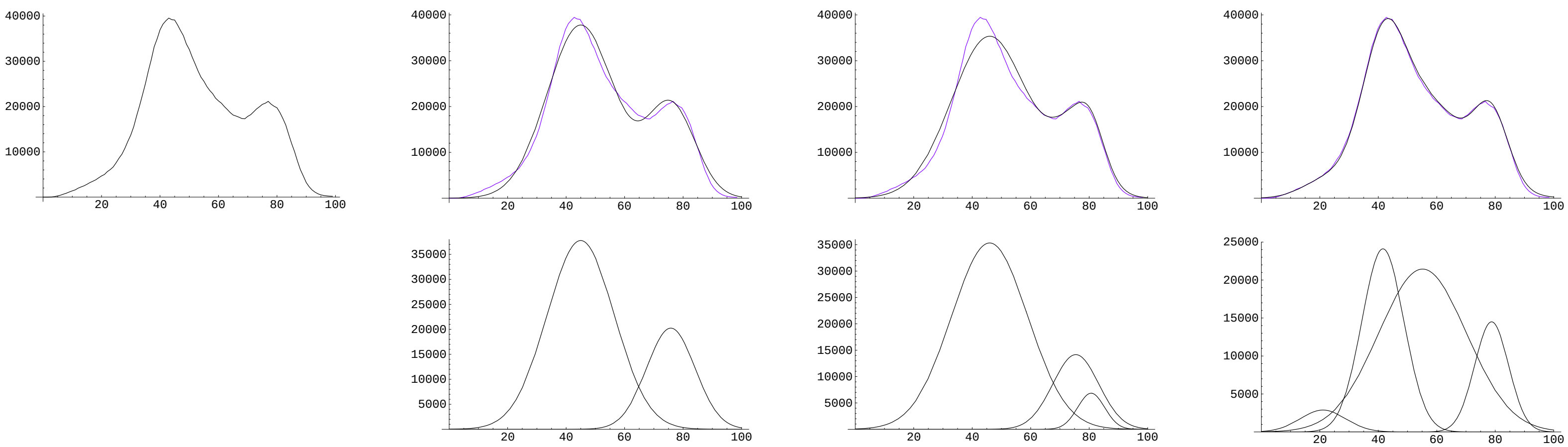
$\Longrightarrow$ Converge to a stationary point, but :

- Can be the optimal minimum
- Can be a local minimum
- One or more of the parameter values may be infinite

Levenberg-Marquardt Algorithm

Results

Fit with 2, 3 and 4 Gaussians :

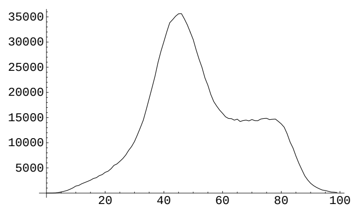


With 4 Gaussians, the clusters represent :

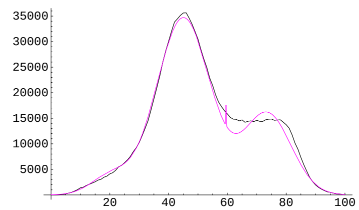
- Noise (background, CSF)
- Cortical Grey Matter
- Deep Grey Matter
- White Matter

Problems

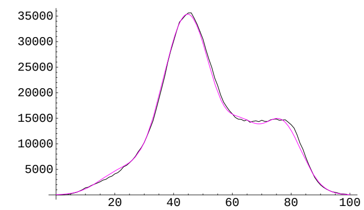
Difficulties to find the good starting values.



Histogram



Fit with the first guessed parameters



Fit after "playing" with the parameters

$\Longrightarrow$ The method is reproducible but the results are not accurate. The algorithm may end in a local minimum.

Approach

- The fitness is represented by an energy value :
  Small energy value $\rightarrow$ High fitness

- From one point in the parameter space (parent vector), an optimization results in a new point (child vector) with a lower energy value. An optimization is a mutation by a random vector.

- We search for the minimal energy value in the parameter space.

$1+1$-Evolution Strategy Algorithm

- Algorithm (first version) :

$$
\begin{aligned}
\vec{r}_t &\sim N(\vec{0}, I) \quad \text{Multidimensional Random Vector } (\mu = 0, \sigma = 1) \\
\vec{x}_1 &= \vec{x}_0 + a_0 \vec{r}_0 \\
a_{t+1} &= \begin{cases} a_t \cdot c_g & \text{if } f(\vec{x}_{t+1}) < f(\vec{x}_{opt}) \\ a_t \cdot c_s & \text{otherwise} \end{cases} \\
\vec{x}_{opt} &= \begin{cases} \vec{x}_{t+1} & \text{if } f(\vec{x}_{t+1}) < f(\vec{x}_{opt}) \\ \vec{x}_{opt} & \text{otherwise} \end{cases} \\
\vec{x}_{t+1} &= \vec{x}_t + a_t \vec{r}_t \quad \text{Add the new direction to the old position}
\end{aligned}
$$

$1+1$-Evolution Strategy Algorithm

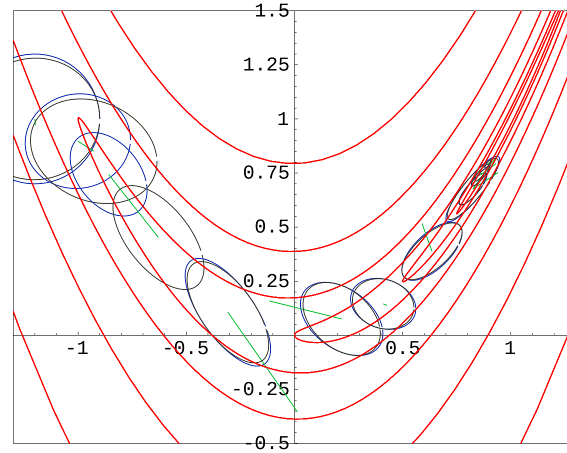
- Algorithm (second version) :

The first version does not do any adjustments to unequally scaled parameters :

$$\text{scalar } a_t \Rightarrow \text{matrix } A$$

$$\begin{aligned} \vec{r}_t &\sim N(\vec{0}, I) \\ A\vec{r}_t + \vec{m} &\sim N(\vec{m}, A \cdot A^T) \end{aligned}$$

After each iteration, the covariance matrix $\Sigma = A \cdot A^T$ is updated.



Convergence of the $(1+1)$-ES algorithm. Illustration of the iterative search procedure, starting upper left. The shapes of the ellipses represent isolevels of the probability distributions for finding a new point.

$1+1$-Evolution Strategy Algorithm

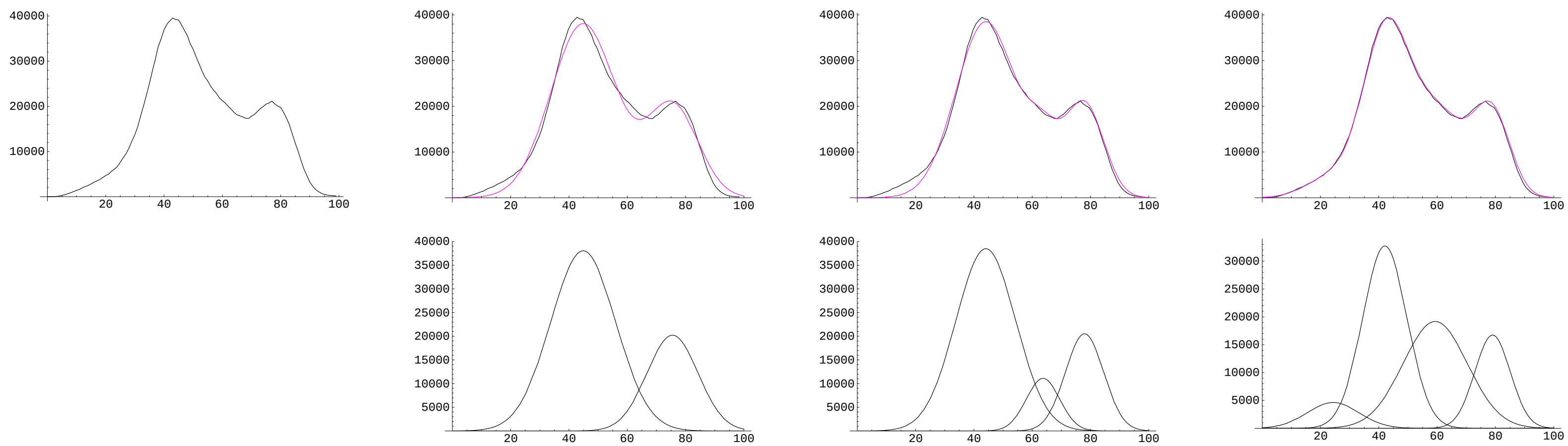
The algorithm becomes :

$$
\begin{aligned}
\vec{r}_t &\sim N(\vec{0}, I) \\
\vec{x}_{t+1} &= \vec{x}_t + A_t \cdot \vec{r}_t \\
A_{t+1} &= \begin{cases} A_t \cdot (I + (c_{grow} - 1) \cdot \dfrac{r_t \cdot r_t^T}{r_t^T \cdot r_t})) & \text{if } f(\vec{x}_{t+1}) < f(\vec{x}_{opt}) \\ A_t \cdot (I + (c_{shrink} - 1) \cdot \dfrac{r_t \cdot r_t^T}{r_t^T \cdot r_t})) & \text{otherwise} \end{cases} \\
\vec{x}_{opt} &= \begin{cases} \vec{x}_{t+1} & \text{if } f(\vec{x}_{t+1}) < f(\vec{x}_{opt}) \\ \vec{x}_{opt} & \text{otherwise} \end{cases}
\end{aligned}
$$

$1+1$-Evolution Strategy Algorithm

Results

Fit with 2, 3 and 4 Gaussians :

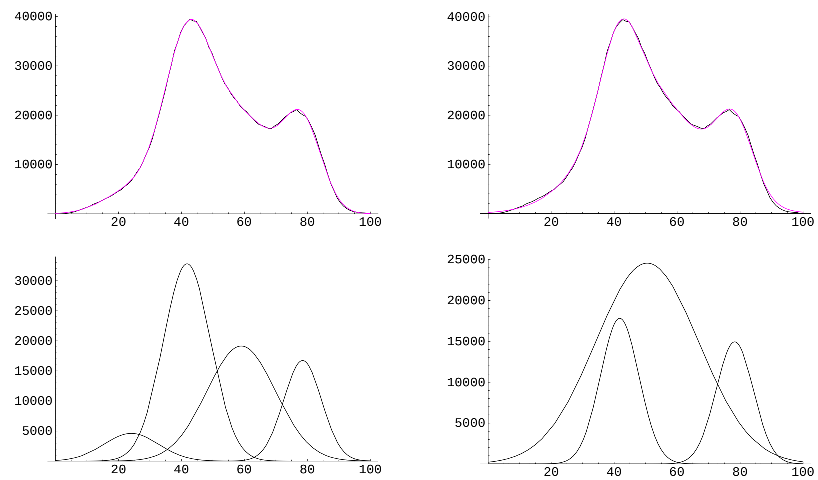


With 4 Gaussians, the clusters represent :

- Noise (Background, CSF)
- Cortical Grey Matter
- Deep Grey Matter
- White Matter

Discussion

The starting values can be chosen farther from the real parameters. This algorithm is a Random Search. We found out that the fit is not unique.



Discussion

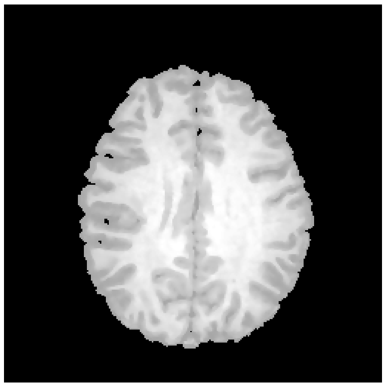
To improve the fit of the histograms, we need to :

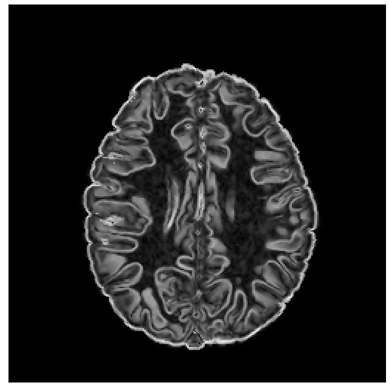
- Reduce the parameter space to a range of possible parameter values :
  - $a \geq 0$
  - $\mu_{min} \leq \mu \leq \mu_{max}$
  - $\sigma_{min} \leq \sigma \leq \sigma_{max}$

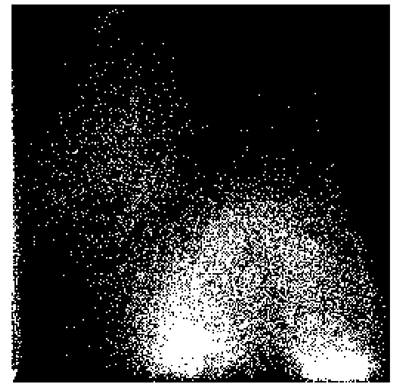
  If the algorithm ends in a point outside the range, we "jump" out of the position : $c_{grow} = c_{grow}^2$.

- Reduce the middle cluster due to :
  - Deep structures
  - Noise
  - Partial voluming at boundaries

Preprocessing







The scatterplot $f(I) vs. \mid \nabla I \mid$ shows :

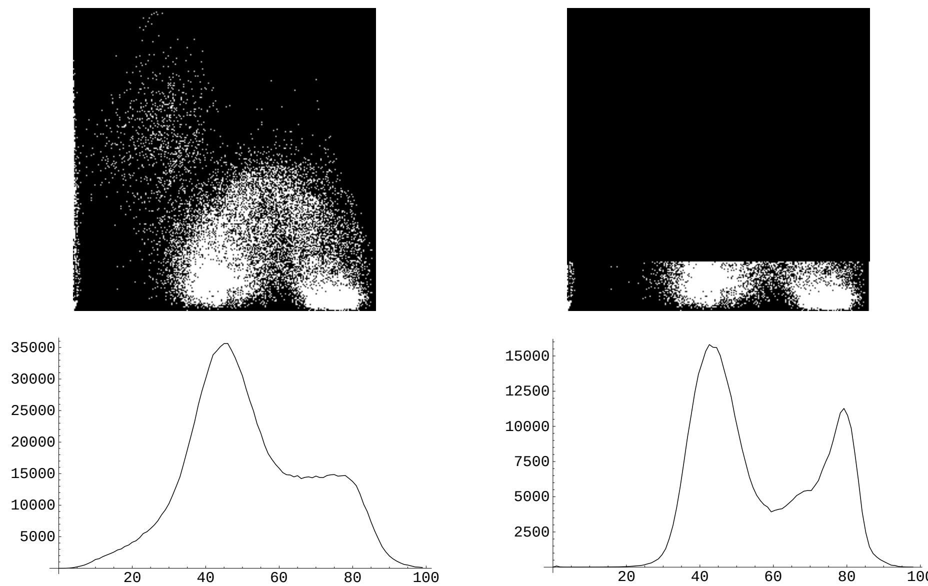
- 2 main clusters : Grey matter and White matter
- Partial voluming between clusters with higher Gradient $\mid \nabla I \mid$
- Noise

Preprocessing

- Noise reduction : Edge preserving smoothing
- Partial voluming reduction : 3D Gradient magnitude on the smoothed image
- Combination of the 2 images to obtain the masked image :

Preprocessing

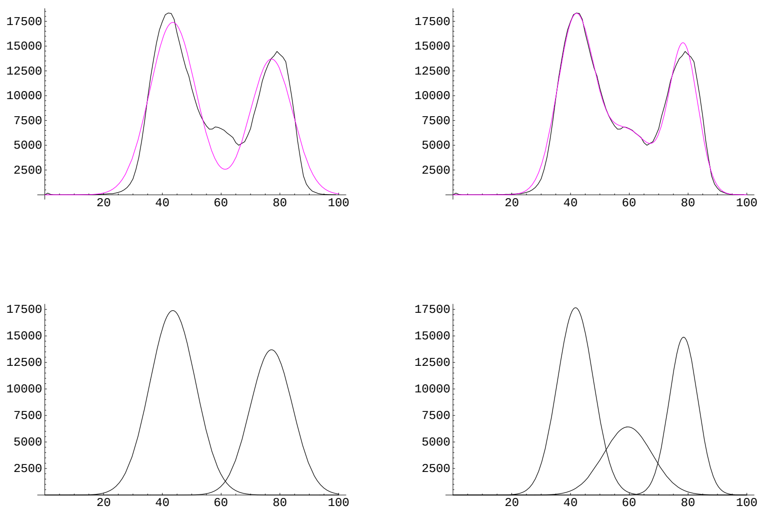
This results in a smaller overlap and the disappearance of the noise cluster :



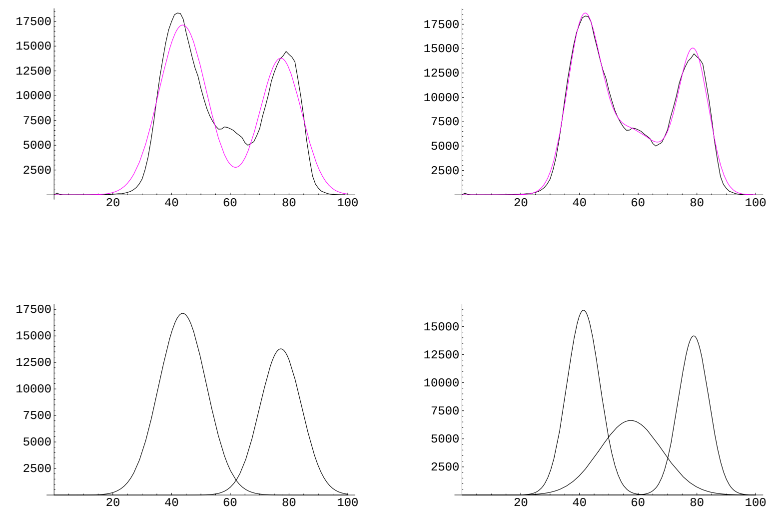
Fit improvement

Results

Levenberg-Marquardt Algorithm



$1 + 1$-ES Algorithm

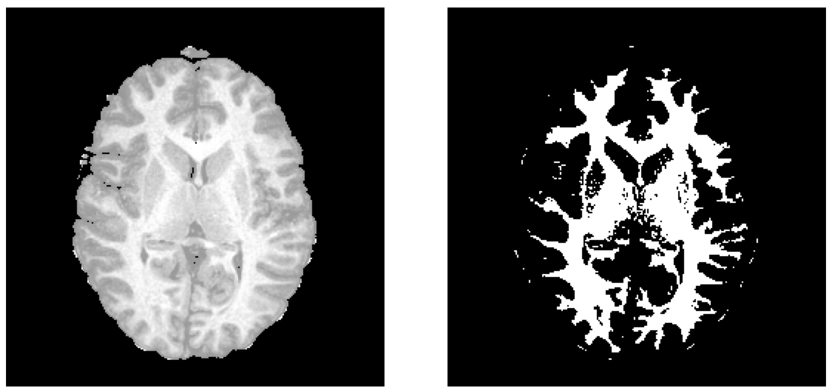


Conclusion

- Preprocessing allows an easier fit of the histograms.
- With LM, the starting parameter values are not as strict as before.
- With $1+1$-ES, only one unique solution came out.
- The input parameter values with $1+1$-ES are more reproducible from one dataset to another with the same protocol than with LM.

Third step :

Estimating the best threshold for brain tissue segmentation



Threshold estimation

We are using the sum of weighted Gaussians, we need to normalized them :

Weighted Gaussians : $G(x) = ae^{-(\frac{x-\mu}{\sigma})^2} \longrightarrow G(x) = \frac{a}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$

with $\int p(x/\omega_i)dx = \int \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} = 1$

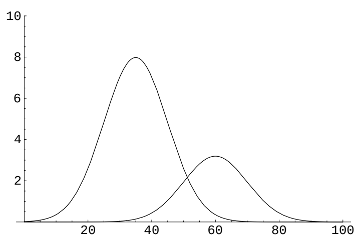
2 classifiers are mainly used in statistics :

- Maximum A Posteriori Classifier : we consider the scale a_i
- Maximum Likelihood Classifier : $a_i = 1$

Maximum A Posteriori Classifier

$$P(\omega_1)p(x/\omega_1) = P(\omega_2)p(x/\omega_2)$$

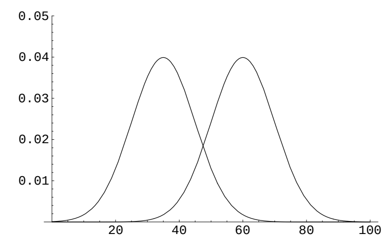
$$\frac{a_1}{\sqrt{2\pi}\sigma_1}e^{-\frac{1}{2}(\frac{x-\mu_1}{\sigma_1})^2} = \frac{a_2}{\sqrt{2\pi}\sigma_2}e^{-\frac{1}{2}(\frac{x-\mu_2}{\sigma_2})^2}$$



Maximum Likelihood Classifier

$$p(x/\omega_1) = p(x/\omega_2)$$

$$\frac{1}{\sqrt{2\pi}\sigma_1}e^{-\frac{1}{2}(\frac{x-\mu_1}{\sigma_1})^2} = \frac{1}{\sqrt{2\pi}\sigma_2}e^{-\frac{1}{2}(\frac{x-\mu_2}{\sigma_2})^2}$$



Supporting Data

Images provided by Dr. James McFall
Image processing Group of Psychiatry
Duke Hospital, Durham, North carolina, USA

- 27 brain MR images of neurofibromatosis 1 cases.
- 12 of them have been segmented with MrX by Martha Payne.

Histogram study

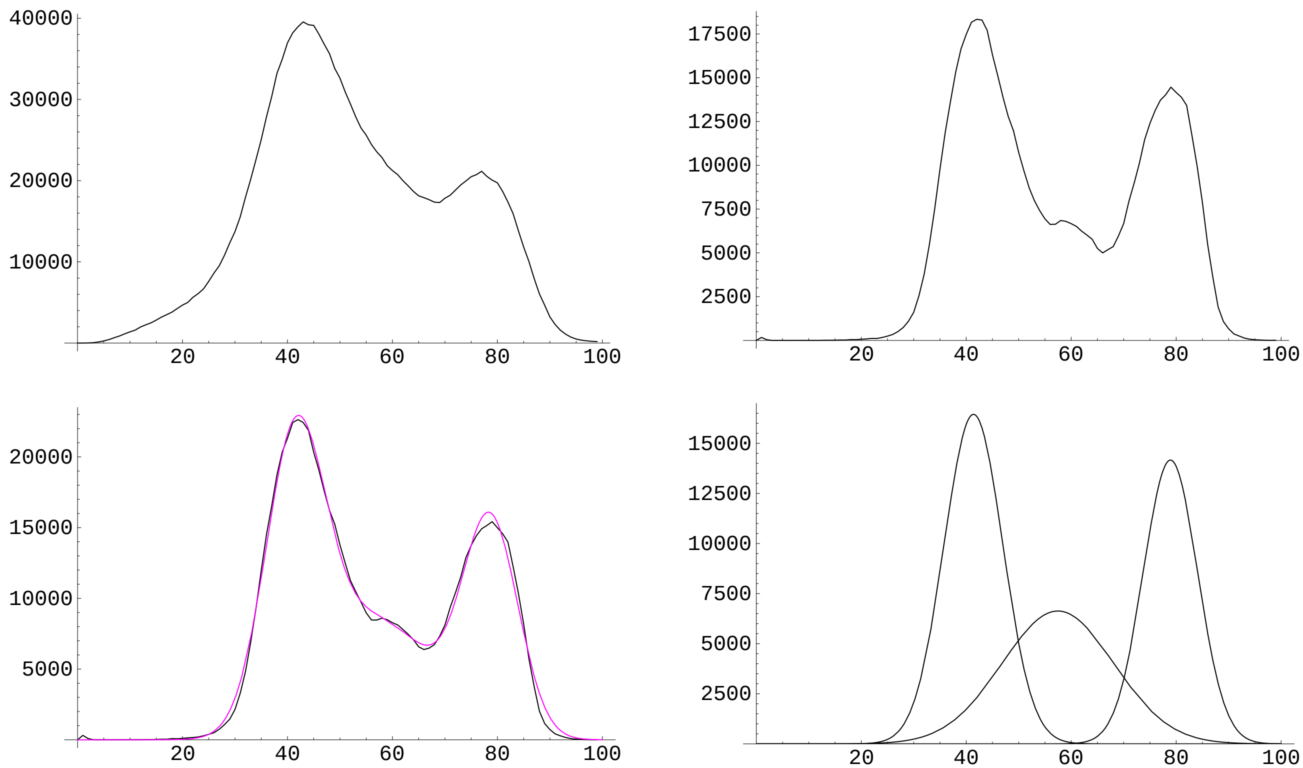
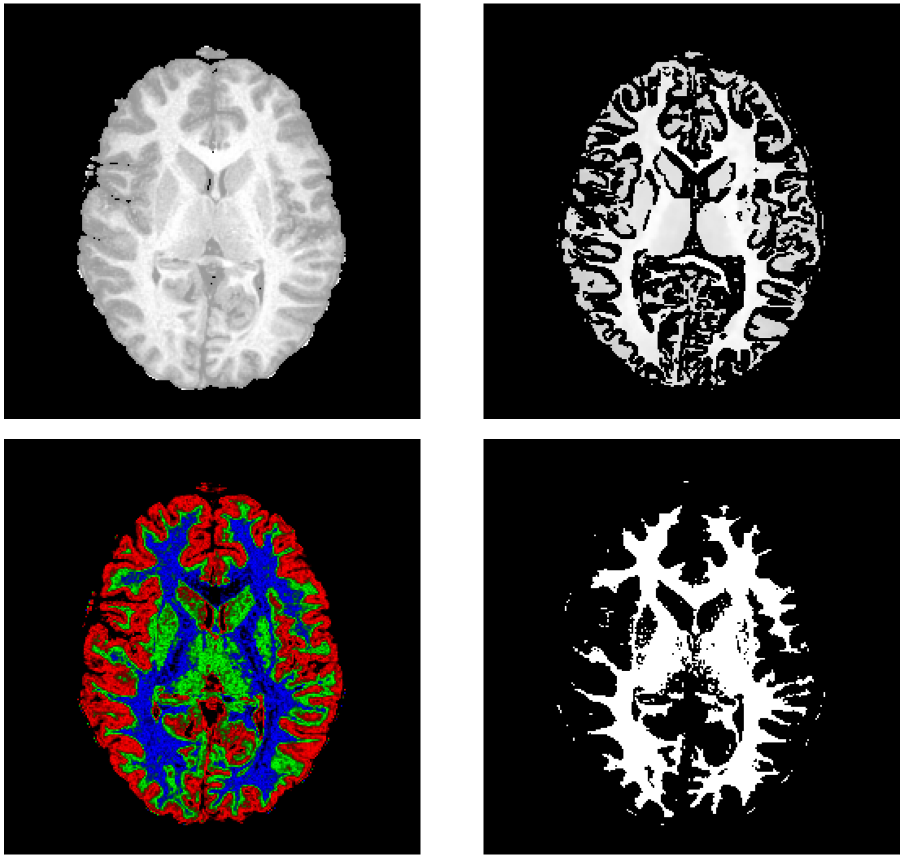


Image segmentation



Tests performed

1. Segmentation of the 27 datasets
   - Based on the full brain.
   - Based on the front half of the brain.
   - Based on the top half of the brain.
   - After bias correction.
2. Comparison on 12 datasets
3. Registration for overlap estimation

Problems

- 4 datasets were difficult to mask.
- 3 datasets were difficult to segment (problem with the fit).
- 1 segmentation failed.

Results : Volume estimation

Comparison of the Supervised & Automatic segmentations.

Error in the volume estimation in % :

	Full Brain		Front Half		Top Half	
	ES	LM	ES	LM	ES	LM
Total Volume	2.58	0.80	3.17	3.94	1.49	1.08
White Matter	28.12	20.94	16.50	13.33	25.59	13.51
Gray Matter	18.90	11.73	16.50	12.91	13.74	8.67

Results : Overlap between the segmentations

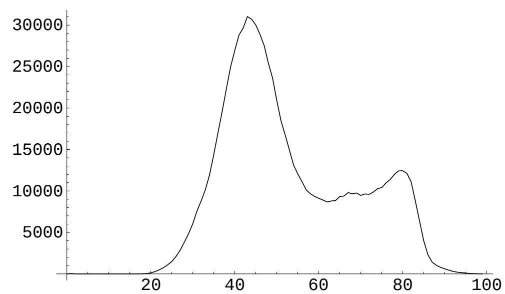
Comparison of the Supervised & Automatic segmentations.

Misclassified pixels in % :

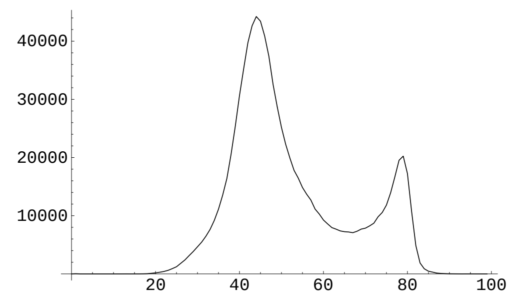
	Full Brain	Front Half	Top Half
White Matter	30 to 45	26 to 48	24 to 50
Gray Matter	11 to 37	13 to 33	16 to 31

Bias correction

There is a difference of contrast between front and back.



Before Bias correction

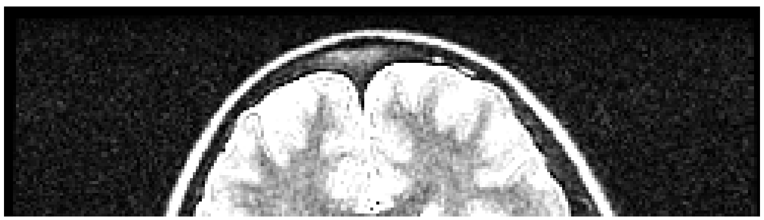


After Bias correction

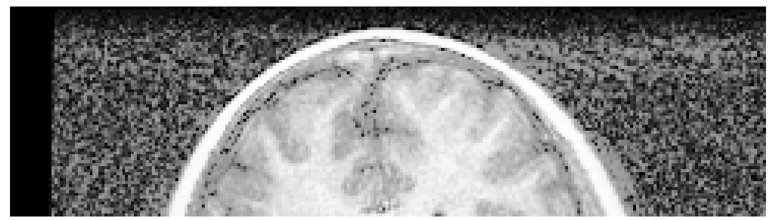
This correction improves the Gray matter volume estimation.

2 different acquisitions

2 different acquisitions $\Longrightarrow$ 2 visual segmentations



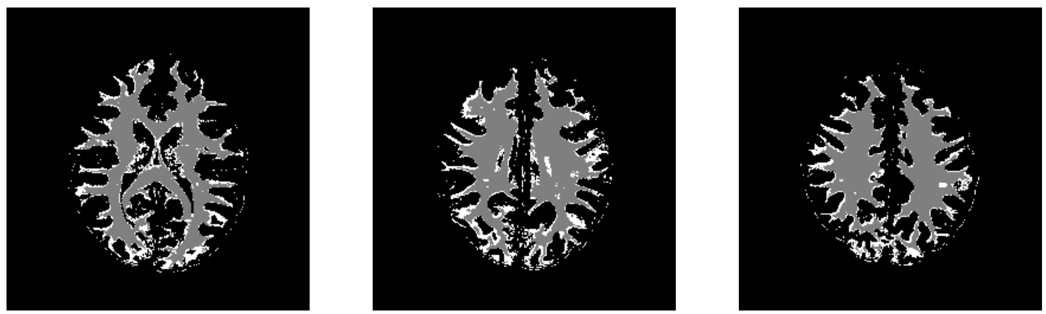
2^{nd} Echo



Gradient Echo

Partial voluming classification

The misclassification is especially due to the Partial voluming :



Misclassified pixels in white.

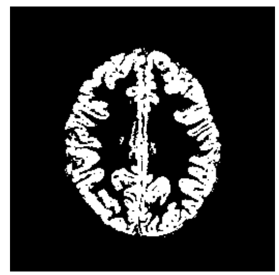
Pixels of Partial voluming are a mixture of tissues, they do not belong only to White or Gray matter.

Gray matter thicker

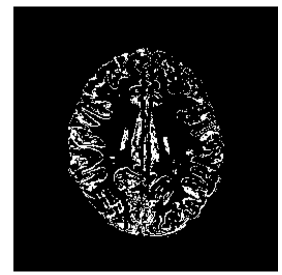
The Automatic segmentation results in a thicker Gray matter :

Gradient Echo image

Supervised segmentation



Automatic segmentation



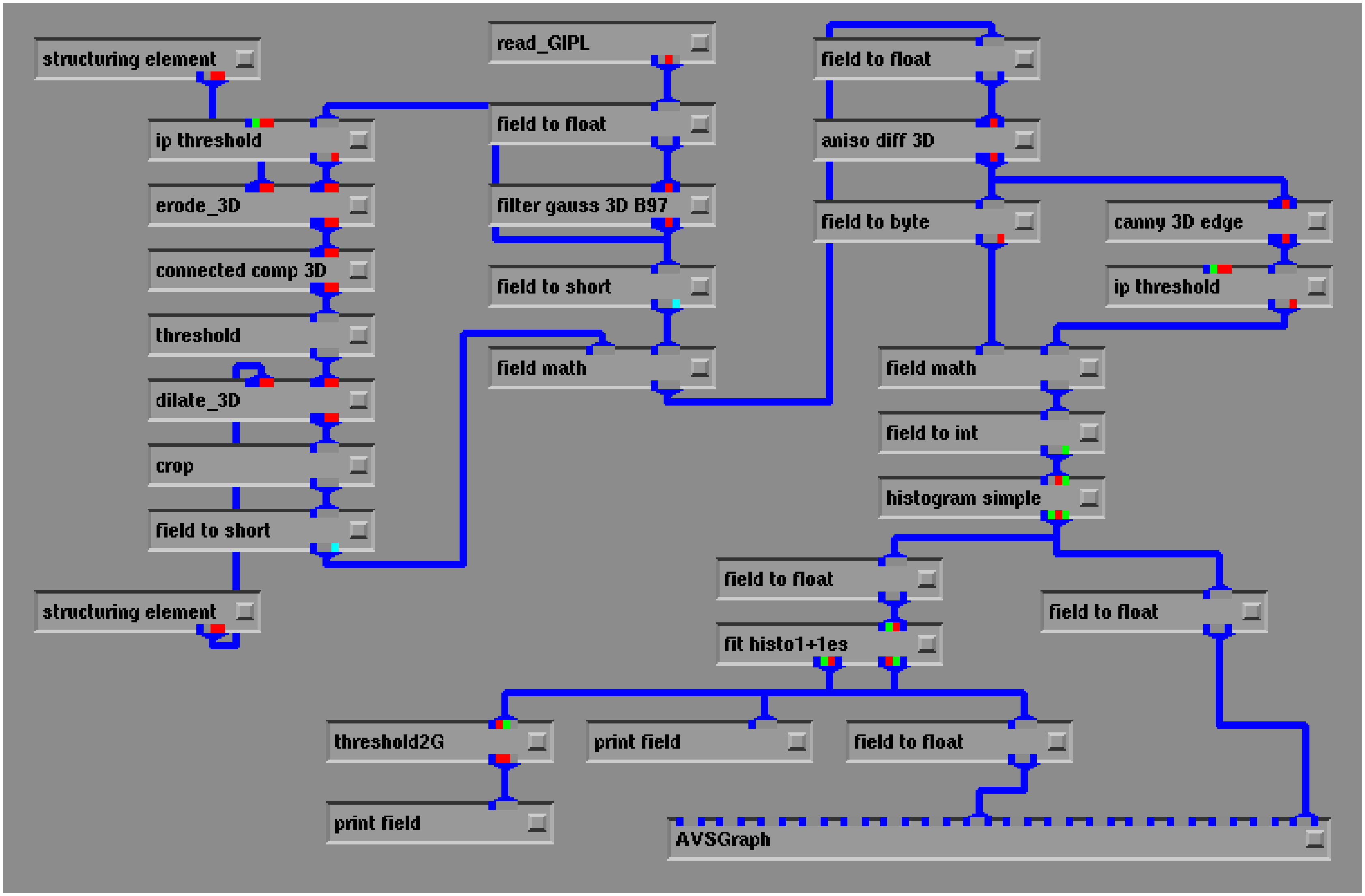
Differences

Conclusion

- Accuracy : Low because of the partial voluming pixels
- Reproducibility : Good, comes only from the appreciation a the “good” fit
- Efficiency : Good, it takes a few minutes to run the AVS Network

Further research is needed to classify the partial voluming.

AVS Network



structuring element
ip threshold
erode_3D
connected comp 3D
threshold
dilate_3D
crop
field to short
structuring element
read_GIPL
field to float
filter gauss 3D B97
field to short
field math
field to float
aniso diff 3D
field to byte
canny 3D edge
ip threshold
field math
field to int
histogram simple
field to float
fit histo1+1es
field to float
threshold2G
print field
print field
field to float
AVSGraph