

MIT OpenCourseWare  
<http://ocw.mit.edu>

HST.582J / 6.555J / 16.456J Biomedical Signal and Image Processing  
Spring 2007

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.

## HST-582J/6.555J/16.456J Biomedical Signal and Image Processing Spring 2007

### Laboratory Project 3 Post Processing Brain MRI Images

**DUE: 4/24/07 by the end of lecture**

## 1 Introduction

It is frequently useful to segment, or determine the spatial extent of, anatomical structures that appear in medical images. This capability facilitates 3D visualization of the structures, as well as quantitative measurements of anatomy.

This lab will explore some of the common techniques that are used for post-processing MRI images. We will work at preprocessing, tissue classification, and brain isolation from clinical MRI. The resulting brain segmentation will be compared to an expert's version, and a simple 3D visualization will be generated.

### 1.1 Segmentation

Brain tissue comes in two major categories that may be visually distinguished, white matter and gray matter. They may also be distinguished in MRI. An example of an MRI cross sectional image of a brain appears in Figure 1.

The simplest sort of segmentation is obtained by thresholding the intensity values in the images. This method can be used when there is strong contrast between the desired structure and everything else. It can work reasonably well for isolating hard bone in CT images, and for separating soft tissues from air in MRI.

### 1.2 Noise Reduction

One difficulty that arises in segmenting MRI is due to the inevitable thermal noise in the images. The signal to noise ratio is limited in practice by the magnetic field strength and the time available for examinations. While this noise does not interfere very much with diagnostic reading of the images, it can be an annoyance in applications that depend on segmentation, by causing fragmentation of the segmented structures. This fragmentation will confuse a 3D rendering of a structure by causing a cloud of debris around the *real* object.



Image courtesy of the project, "Retrospective Image Registration Evaluation", National Institutes of Health, 8R01EB002124-03, Principal Investigator, J. Michael Fitzpatrick, Vanderbilt University, Nashville, TN.

Figure 1: An annotated Gradient Echo MR slice (Air, CSF and Cranium are dark in these images)

We will explore linear and non-linear filtering methods for noise reduction.

### 1.3 Statistical Classification

Statistical classification is a commonly-used method for segmenting MRI, and the theory will be covered in the lectures. Briefly, in a training phase, models of the intensity properties of the various tissues are constructed from labeled data. Each pixel of the image to be segmented is compared against the tissue models and assigned a most probable or likely tissue category.

We will build and use a *simple* classifier based on the assumption that image intensities are modeled as Gaussian random variables whose parameters depend on the tissue class (i.e. the class-conditional densities are Gaussian).

The following is a very brief exposition of statistical classification using Gaussian models, as is (mostly!) implemented in the supplied code.

The probability of pixel intensities, conditioned on knowing the tissue class at the pixel, will be modeled by a Gaussian probability density,

$$G(\mu, \sigma, x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$p(x_i \mid \text{pixel } i \text{ is tissue class } j) = G(\mu_j, \sigma_j, x_i) \quad .$$

In order to make useful tissue-class conditional models on intensity, we will need to find suitable parameters, the mean and variance, of the Gaussian density for each tissue class.

One approach to this problem is to use the maximum-likelihood method of estimating these parameters from labeled training data, as follows. Let  $Y_j$  be the set of training pixels for tissue class  $j$ , then

$$\mu_j = \frac{1}{|Y_j|} \sum_{y \in Y_j} y$$

$$\sigma_j^2 = \frac{1}{|Y_j|} \sum_{y \in Y_j} (y - \mu_j)^2$$

where  $|Y_j|$  is the number of elements in the set  $Y_j$ .

If  $P_j$  is the prior probability of tissue class  $j$  then the Bayes' classification of pixel  $i$  is obtained by choosing the  $j$  which maximizes

$$P_j p(x_i \mid \text{Pixel } i \text{ is tissue class } j).$$

If all tissue classes have equal prior probability then this rule simplifies to choosing the  $j$  which maximizes

$$p(x_i \mid \text{Pixel } i \text{ is tissue class } j).$$

## 1.4 Brain Isolation

While pixel-intensity based segmentation methods, like the method described above, are a good start on anatomical labeling from MRI, they usually produce noticeably imperfect results due to contrast ambiguities among some of the tissues. In brain segmentation, a common failing is the appearance of spurious bits of brain tissue in the soft tissues of the head. Problems of this sort may be attacked by using methods that pay attention to spatial or structural aspects of the segmentation results.

In this lab, we will investigate the use of the image processing tools that are collectively referred to as *mathematical morphology* (erosion, dilation, opening, closing, etc.). These methods, in conjunction with *connected components* operations, are frequently used to repair problems left over from intensity-based segmentation methods.

## 1.5 Validation

Validation of the results of MRI segmentation is an important, but problematic area, because it is hard to obtain ground truth for testing purposes. A commonly used method is to use manual segmentations performed by expert human *raters* as the standard, but this has several problems. It is not easy to convince a real expert to manually label the several megavoxels in a brain MRI. Also, the human raters introduce variability by not agreeing terribly well with each other.

You will investigate some simple approaches to validation that are based on a supplied gold standard.

## 2 Lab Exercises

The supplied code and data for this laboratory project are located in `/mit/6.555/matlab/seg/` and `/mit/6.555/data/seg`, respectively. \* All of the coding is done in MATLAB. The data contains slices from two individuals and an expert's segmentation of one of the data sets.

### 2.1 Pre-Processing

Try not to spend too much time on this section of the lab, because more involved and interesting stuff comes later.

1. Load and display the `sw/spgr/I.082` MRI slice by using `mri_read.m`.
2. Find a threshold that does a reasonably good job of separating white and gray matter.

**Question 1** *What intensity value did you use for the threshold? In your lab report, include pictures of the original slice and the thresholded image.*

3. Notice that the segmentation results from above contain some fragmentation due to contamination by white noise. A standard method for noise reduction is linear filtering. Use a lowpass filter to reduce the noise-induced fragmentation. Find a smoothing filter with a standard deviation that is a reasonable compromise between structural faithfulness and noise reduction. You can use the MATLAB command `fspecial` to create a Gaussian filter.

**Question 2** *Briefly describe your smoothing filter. In your lab report, include a picture of the slice **after smoothing** and the new thresholded image.*

\* For OCW users: these data and code files are supplied in the supporting ZIP archive.

4. Non-linear noise reduction methods, for example median filtering, are often the choice for MRI. Find a combination of threshold and median filter neighborhood size that is a reasonable compromise between structural faithfulness and noise reduction. You can use the MATLAB command **medfilt2**.

**Question 3** *State the threshold and median filter neighborhood size that you used. In your lab report, include a picture of the slice **after median filtering** and the new thresholded image.*

## 2.2 Statistical Classification

This next section uses a statistical classifier to segment brain MRI. We start by exploring histogramming and the construction of Gaussian intensity models for each tissue based on training data. The training data provided consists of a segmentation that was performed by an expert. For this part of the lab, copy and modify the function **classify.m** based upon the instructions below.

5. Create a subdirectory *output/classification* in your own working directory to hold the output of the following operations. Specify its exact location in the variable **CLASSIFICATION\_PREFIX**.

6. View the preselected MRI slice, **img**, and the corresponding expert label map, **labels**, that are read in at the beginning of **classify.m**.

### 2.2.1 Histograms

Histogramming is one approach to estimating probability density functions (PDFs). Start by inspecting the histogram generated by the **hist.m** function in the early part of **classify.m**. Notice that it is not a legitimate PDF, because, when viewed as a function over the real line, it does not integrate to one.

Your task in this section is to implement a histogram based estimate of the PDF for the intensity of white matter in the training data. Your implementation will need to construct a MATLAB function that has two parts: the first constructs the state of the histogram from the training data by, among other things, establishing an array of *bins* and counting the number of training data points that fall into each bin. The second part uses this histogram to generate a PDF, as a function from real-valued intensities to probability density. In this situation, each bin will correspond to a range of intensity values. Keep in mind that PDF must be non-negative and integrate to one.

7. Copy and complete the MATLAB file template **HistPDF.m** to generate a valid histogram-based PDF. (You may also want to look at the documentation for the MATLAB function **hist**).

8. There is some freedom available in making histograms, for example the choice of the number of bins. Experiment with the number of bins and determine a “good” choice of bin count for generating your histogram-based PDF.

**Question 4** *How many bins did you use? Briefly explain why you think it is a good choice and turn in the corresponding plot as well your MATLAB code.*

### 2.2.2 Gaussian Models

Next we will estimate PDFs using Gaussian class conditional models that capture the way the intensities depend on tissue class. Your task is to implement the maximum likelihood method for estimating the parameters of Gaussian class conditional PDFs.

A Gaussian is defined by its mean and variance. Two vectors, **tissue\_means** and **tissue\_vars**, will eventually hold the estimated tissue class mean and variance parameters for the estimated Gaussian PDFs, indexed by the tissue class which can be white, gray, or csf.

9. Set the two vectors defined above with three values each that are derived from the observed class conditional intensities: `intensity_white`, `intensity_gray`, and `intensity_csf`.

**Question 5** *Turn in your code fragment that sets values for **tissue\_means** and **tissue\_vars**.*

The code defines the Gaussian density function, which we can use to estimate class conditional PDFs on intensity by the use of Gaussian models. This was done by using the ML method to estimate the mean and variance parameters for each class, based on the intensities from the training data.

10. View the Gaussian PDFs by plotting, for example, `plot(gauss([0:500], tissue_means(1), sqrt(tissue_vars(1))), 'r')` Generate a plot that, for white matter, shows the Gaussian-based PDFs along with the histogram-based PDF generated earlier.

**Question 6** *Include this plot in your lab report. What differences do you see between the two models?*

Next you will implement a Bayes’ classifier for the image intensities (assuming the *prior* probability of each class is equal) using the Gaussian models that we have constructed. In **classify.m**, the code loops over the test slices and performs classification. After reading each slide, the code displays three images showing the likelihoods for each tissue. After you supply the missing code, the tissue class is determined for each pixel. The resulting classifications are then written out for subsequent processing later in the lab.

11. Complete the MATLAB code to fill in the matrix **classification** with the ML classification of the image. **classification** has the same dimensions as **img**, and should contain the predefined tissue labels LABEL\_WHITE, LABEL\_GRAY, LABEL\_CSF to indicate the classification of the image. Try to do this the MATLAB way, and avoid looping over pixels!

**Question 7** *As the classifier runs, it displays the tissue likelihoods as well as the resulting classification. What different information do the likelihoods and classification contain? Turn in your MATLAB code fragment that fills in the **classification** matrix.*

## 2.3 Brain Isolation

The next part of the lab concerns an attempt at isolating the brain structure. The corresponding code is located in **morphology.m**. It uses the *conditional morphology / largest connected component* strategy. The code expects input slices that are labeled as white matter, gray matter, and other stuff.

12. Fill in the **CLASSIFICATION\_PREFIX** and the **BRAIN\_PREFIX** variables and run the code.

After completion, you will (hopefully) notice that this brain isolation algorithm failed on slice 102, slice 110, and a few other slices as well.

**Question 8** *Why did the isolation fail on these parts? Specifically compare your results on slice 102 to slice 110. Describe in what sense the isolation failed and possible reasons for the failure. Describe (briefly) how you might modify or change the isolation procedure to fix these problems.*

## 2.4 Validation

The penultimate part of the lab involves a validation of your brain segmentation against that of the expert. The supplied code in **validation.m** loads and displays your version and the expert's for slice 130.

14. Fill in the **CLASSIFICATION\_PREFIX** and the **BRAIN\_PREFIX** variables and run the code.

You will notice that your result is not the same as the *gold standard*. The *gold standard* is the result of a more sophisticated algorithm which uses anatomical information in addition to the intensity models to determine tissue type.

**Question 9** *Quantify the agreement of your segmentation with the expert with respect to white matter and gray matter in slice 130. That is, create a table which contains the prob-*



ability that your classifier made decision  $i$  conditioned on the **expert** making decision  $j$ . Additionally, in a few sentences suggest possible qualitative reasons for the differences between your results and the gold standard (e.g. what types of information are you ignoring and how might you incorporate it statistically).

## 2.5 3D Rendering

The last, and easiest, part of the lab involves an attempt at generating a simple 3D surface rendering of a portion of the white matter structure (that was computed in the previous exercises). The file **render.m** contains a quick hack implementing this task.

16. Fill in the **CLASSIFICATION\_PREFIX** and the **BRAIN\_PREFIX** variables and run the code.

17. After you get a display up, flip it around and try to get the most informative view of the surface.

**Question 10** *Include a picture of that view in your lab report.*

## 3 Conclusions

**Question 11** *What did you find most challenging about this lab exercise?*

**Question 12** *What is the most important thing that you learned from this lab exercise?*

**Question 13** *What did you like/dislike the most about this lab exercise?*