EMSegment Tutorial

How to Define and Fine-Tune Automatic Brain Compartment Segmentation and the Detection of White Matter Hyperintensities

This documentation serves as a tutorial to learn to customize the `3D Slicer EMSegment` module for `White Matter Hyperintensity` and `Deep Gray Matter` segmentation based on a sample case. The strategies described here can also be applied to different segmentation problems.

Istvan Csapo, Kilian Pohl, Charles R.G. Guttmann

Center for Neurological Imaging and Surgical Planning Laboratory
Brigham & Women's Hospital
Harvard Medical School
icsapo@bwh.harvard.edu
pohl@csail.mit.edu
guttmann@bwh.harvard.edu

Version 1
(3D Slicer 2.6)

Last update: January 4, 2007

Table of Contents

How to Use This Tutorial

General Usage

This guide assumes a basic knowledge of `3D Slicer`, including, but not limited to loading volumes and label maps; loading scenes; saving volumes; and registering images. For further information on these topics, please refer to the `3D Slicer` website (http://www.slicer.org).

Glossary

The terminology used in reference to general `Slicer` components is the same as found on the `Slicer` website.

CGM – cortical gray matter
CSF – cerebrospinal fluid
DEEPGM – deep gray matter (caudate, putamen, thalamus)
ECC – extra cranial content (inverse of ICC: skull, eyes, skin, background, etc.)
GM – gray matter (including CGM and DEEPGM)
ICC – intra-cranial cavity (GM, WM, and CSF)
NAWM – normal appearing white matter
WM – white matter (including both NAWM and WMH)
WMH – white matter hyperintensity

Sample Case

The tutorial describes general strategies for customizing the `EMSegment` module. However, it also includes the specifics for a sample case. The two input channels for the sample case are MPRAGE and FLAIR. Different channels may be used, but the strategies described here are specific to the aforementioned two channels.

All the necessary files for the sample case are included in `EMSegmentTutorial_v1.zip` and can be download from http://www.na-mic.org/Wiki/images/6/6e/EMSegmentTutorial.zip.

```
Instructions and parameter settings for the sample case appear in this textbox style throughout the tutorial.

Note: The parameters used for the sample case produce good results; however, these might not be the best results.
```

Sample Case Files

The `EMSegmentTutorial_v1.zip` contains the following files:

- Subject Files:
  - subject001_MPRAGE.nii.gz – MPRAGE (176x256 in plane resolution, 176 slices, $1x1x1mm^3$ voxel)
  - subject001_rFLAIR.nii.gz – FLAIR registered to MPRAGE
- Atlas Files (registered to MPRAGE):
  - subject001_icbmCSF.nii.gz – probabilistic atlas of the CSF
  - subject001_icbmDEEPGM.nii.gz – probabilistic atlas of the deep GM structures (caudate, putamen, thalamus)
  - subject001_icbmGRAY.nii.gz – probabilistic atlas of the GM
  - subject001_icbmWHITE.nii.gz – probabilistic atlas of the WM
- Label Maps:
  - subject001_ECC.nii.gz – ECC label map (inverse of ICC)
- XML Files:
  - subject001_scene01_Images.xml – scene file containing all the images
  - subject001_scene01.xml – scene file with all the images and settings for GM, WM, and CSF segmentation
- Results:
  - subject001_scene01_Seg.nii.gz

Atlas

The tutorial uses the `ICBM` probabilistic atlases (available at http://www.loni.ucla.edu/Atlases). The atlases are registered to the MPRAGE image using the FMRIB Linear Registration Tool (FLIRT, http://www.fmrib.ox.ac.uk/fsl/flirt/index.html), applied with 9 degrees of freedom.

ECC

The ECC label map was generated by inverting the ICC label map. The ICC was automatically generated from the subject's T2 image and then manually corrected by an expert user.

File Formats

All images and label maps are in NifTI-1 format (http://nifti.nimh.nih.gov). Use `Slicer`'s `Menu Window -> Data Panel -> Add Volume -> Generic Readers` to read NifTI images.

Loading Scenes

There are some scene files included with the sample case. Opening these files takes care of loading the images, setting the `EMSegment` parameters, and setting the labels/colors in one easy step. It is recommended to start with these scene files when starting a new segmentation problem and modify the parameters as seen necessary.

Segmentation

`EMSegment` combines information from the input channels, and optionally the probabilistic atlases, to create a segmentation label map that delineates the subclasses defined in the class hierarchy. For a more detailed description of the `EMSegment` algorithm, refer to [1].

1 GRAY MATTER, WHITE MATTER, AND CSF SEGMENTATION

This section describes the process of segmenting the intra-cranial cavity (ICC) into gray matter (GM), white matter (WM), and cerebrospinal fluid (CSF).

1.1 General Strategy

1. Load the subject and the atlas images.
2. Set the `EMSegment` parameters.
3. Set up the class hierarchy (see Figure 1.1).
4. Build the intensity distribution for each subclass.
5. Run `EMSegment`.
6. Refine the parameters and the class distributions and rerun `EMSegment` until satisfactory segmentation is achieved.
7. Save the parameters into a scene file – this file can then be used to segment datasets with the same characteristics.

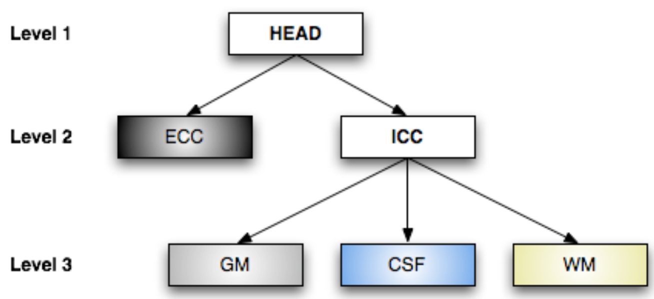

Figure 1.1: The class hierarchy for GM, WM, and CSF segmentation (HEAD and ICC are super classes).

1.2 Files Used

- Subject Files:
    - subject001_MPRAGE.nii.gz
    - subject001_rFLAIR.nii.gz
- Atlas Files (registered to MPRAGE):
    - subject001_icbmCSF.nii.gz
    - subject001_icbmGRAY.nii.gz
    - subject001_icbmWHITE.nii.gz
- Label Maps:
    - subject001_ECC.nii.gz

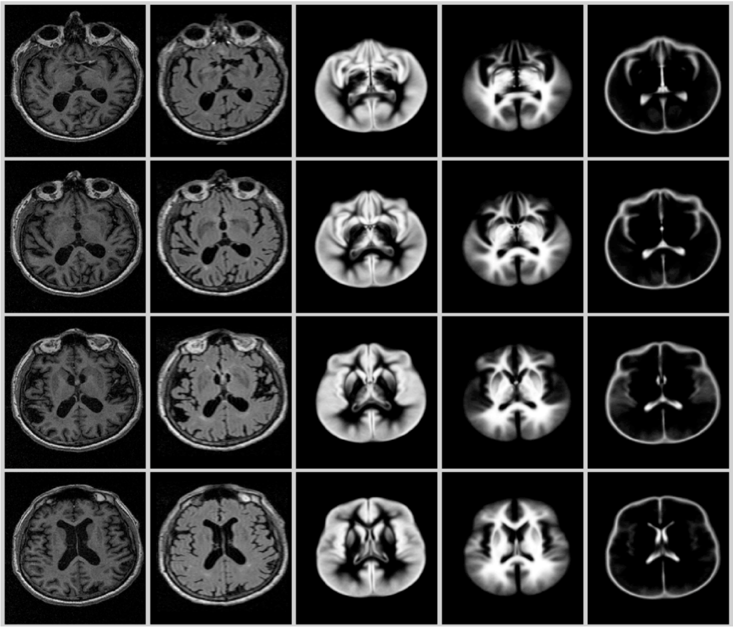

Figure 1.2: Sample slices from the MPRAGE and FLAIR subject images and the GM, WM, and CSF probabilistic atlases.

1.3 Loading the Subject and Atlas Images

Load the input channels (subject images) and the probabilistic atlas images into `Slicer`.

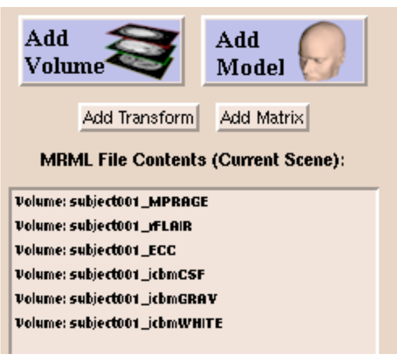

Figure 1.3: Contents of the Current Scene window after loading the images.

```
There are two ways to load the sample images into Slicer:

1. Automatically via Menu Window -> File Menu -> Open Scene:
      subject001_scene01_Images.xml

2. Manually via Menu Window -> Data Panel -> Add Volume:
      subject001_MPRAGE.nii.gz
      subject001_rFLAIR.nii.gz
      subject001_ECC.nii.gz
      subject001_icbmCSF.nii.gz
      subject001_icbmGRAY.nii.gz
      subject001_icbmWHITE.nii.gz

Note: if you load subject001_scene01.xml, these and all other parameters for the GM, WM, and CSF segmentation will be
set automatically. Refer to section 1.10 for saving the parameters into an XML file.
```

1.4 Setting the EMSegment Parameters

Go to the `EMSegment` module: `Menu Window -> Modules -> Segmentation -> EMSegment` or `Menu Window -> More -> EMSegment`.

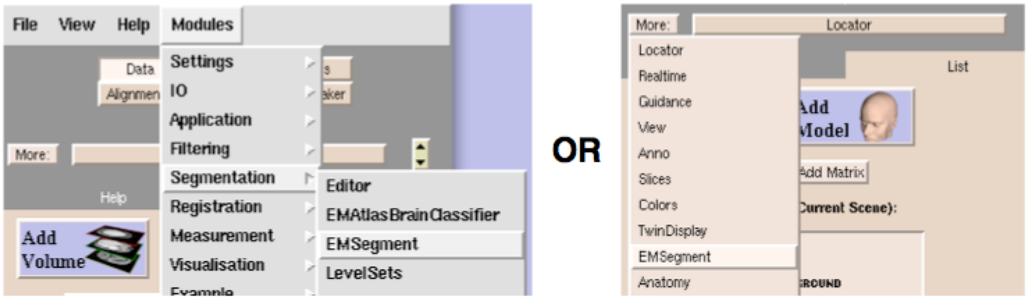

Figure 1.4: Opening the EMSegment module.

Overview of the EMSegment Tabs:

- `Help` – Brief description of the module.
- `EM` – Guides the user through the general steps of the segmentation. This tab can also be used to set up the class hierarchy, although, the Class tab provides access to more advanced setups.
- `Class` – Parameters of the class hierarchy.
- `CIM` – Class Interaction Matrix (not modified in this tutorial).
- `Setting` – Parameters of the EM algorithm.

Adding Input Channels (EM -> Step1 Tab)

The intensity information from the input channels, combined with the optional probability atlases, is used to classify the image pixels into different classes. Only two channels are used in this tutorial; however, any number of channels can be used.

To add the input channels go to the `EM` panel -> `Step1` tab. The `Volume List` window shows the available loaded volumes.

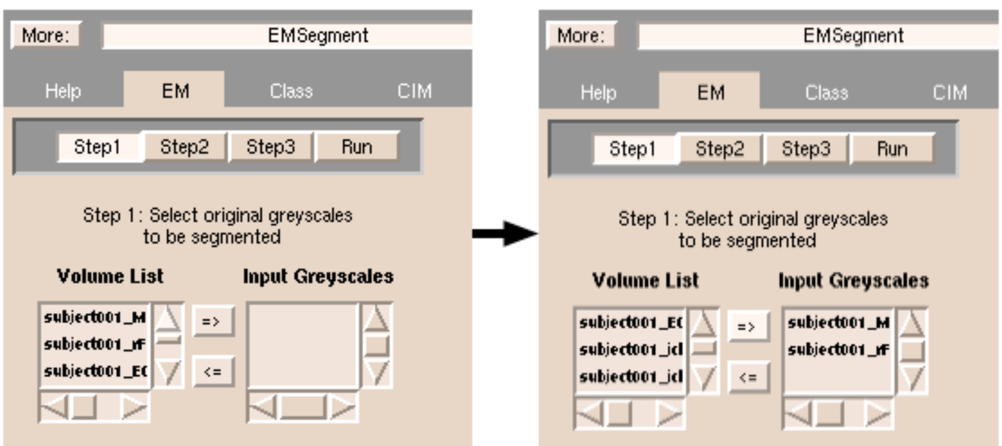

Figure 1.5: Adding MPRAGE and FLAIR as input channels. Highlight the image to be added in the Volume List window then click on the right arrow to add. Use the left arrow to remove images from the Input Greyscales window. Note: the input channels are referenced by the order they appear in (MPRAGE becomes channel 0, FLAIR becomes channel 1).

Add subject001_MPRAGE and subject001_rFLAIR to the **Input Greyscales** window. **MPRAGE** should be added first and **FLAIR** should be added second.

Note: Keep in mind the order of the input channels: **MPRAGE** becomes channel 0, **FLAIR** becomes channel 1.

Tip: `Slicer` may sometimes become unstable and crash without a warning if certain parameters are not properly set. To avoid retyping the parameter values, periodically save the settings as described in section 1.10.

Setting the Number of Iterations, Class Settings, and Boundary Settings (EM -> Step2 Tab)

Parameters:

- `No. of classes` – Defines the number of subclasses under the current super class. The classes are hierarchically organized. The top super class, by default, is the `Head` super class.
- `Iterations` – The number of maximum iterations the EM algorithm is to perform.
- `Boundary Min, Max` – Define the area of the image to be segmented.

Tip: Increasing the number of iterations increases the running time and might not have much affect on the segmentation quality. Set the number of iterations to the lowest that produces good results.

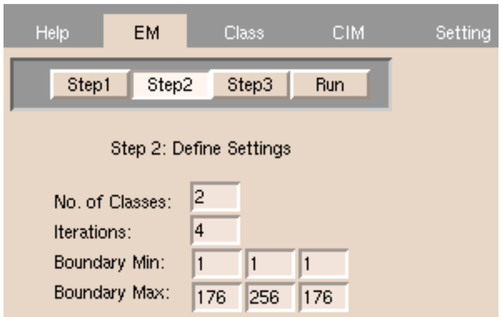

Figure 1.6: Define general parameters.

```
Set the following parameters:

No. of Classes:         2
Iterations:             4
Boundary Min:           1, 1, 1
Boundary Max:           176, 256, 176
```

1.5 Setting Up the Class Hierarchy (Class Tab)

The class hierarchy determines the relationship between the different classes. The structure of the class hierarchy and the parameters of each class can be changed under the `Class` tab. See Figure 1.1 for a general overview of the hierarchy.

Head Super Class

The current class can be selected with the top button of the `Class` tab. First, choose the `Head` super class:

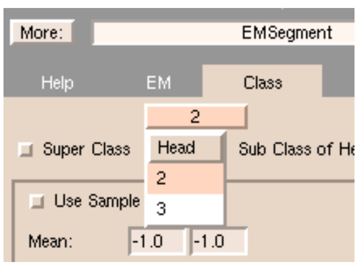

Figure 1.7: Choosing the current class. Click on the top button in the class tab and select the desired class from the pull-down menu.

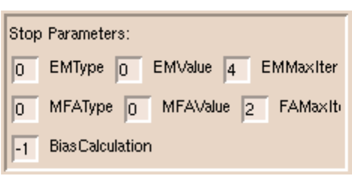

Figure 1.8: `Stop Parameters` for the `Super Class` structure.

Description of parameters for the `Super Class` structure:

- `Name` – Name of the super class.
- `Class Probability` – Expected proportion of the classes at the given level. The class probabilities for a level have to add up to 1. (e.g. Figure 1, level 3: GM: 0.6; WM: 0.3; CSF: 0.1).
- `Number of Classes` – Number of subclasses.
- `Prob Data Weight` – Weight of the probability atlas.
- `Input Channel Weights` – Weights of the input channels.
- `Print Parameters` – Intermediate results, such as the `LabelMap`, can be printed out at each iteration. `Frequency` defines how often these results will be printed out (-1 = only the last iteration; 0 = never; >0 = at each iteration).
- `Stop Parameters`
  - `EMType` – Defines after which criteria should the algorithm halt. 0 = fixed number of iterations (defined by EMMaxIter); 1 = “when label map converges” measure (calculate difference measure by comparing labelmaps between iterations); 2 = “when weights calculated in the E-Step converge” measure (calculate difference measure by comparing weights between iterations).
  - `EMValue` – Defines the threshold for stopping the algorithm based on the average difference (DifferenceMeasure/Number of Voxels) (e.g. if the algorithm should stop if the difference between

iterations is below 1%, then set EMValue to 0.01). Note: the algorithm halts if the number of iterations goes beyond EMMaxIter; EMValue is only activated if EMType >0.

- `EMMaxIter` – The number of maximum iterations of the algorithm.
- `MFAType` – Defined as EMType, but for MFA.
- `MFAValue` – Defined as EMValue, but for MFA.
- `FAMaxIter` – Should be MFAMaxIter but tcl displays it wrong. Definition as for EMMaxIter just applied to MFA.
- `BiasCalculation` – At each EM iteration the algorithm not only computes a MFA, but also an inhomogeneity correction. This parameter defines after how many EM iterations should the inhomogeneity correction stop (default = -1, which means never stop).

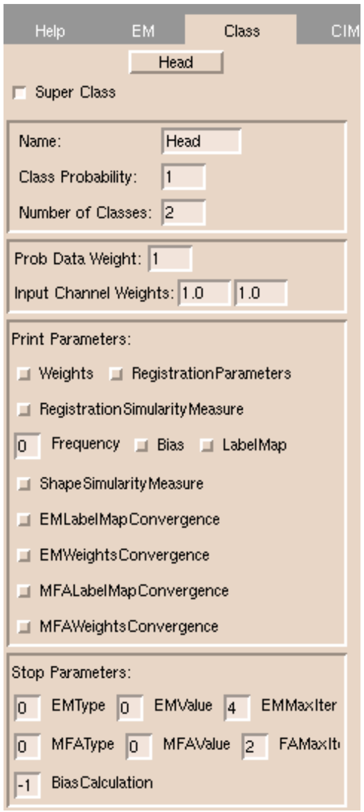

Figure 1.9: Parameters for the `Head` super class.

```
Set the following parameters:

      Name:                     Head
      Class Probability:        1.0
      Number of Classes:        2
      Prob Data Weight:         1.0
      Input Channel Weights:    1.0, 1.0
      Stop Parameters
              EMMaxIter:        4
              FAMaxIter:        2

The rest of the parameters can be left at their default settings.
```

Extra Cranial Content (ECC) Subclass

Change the current class to the first subclass of the `Head` super class. This subclass will represent the ECC.

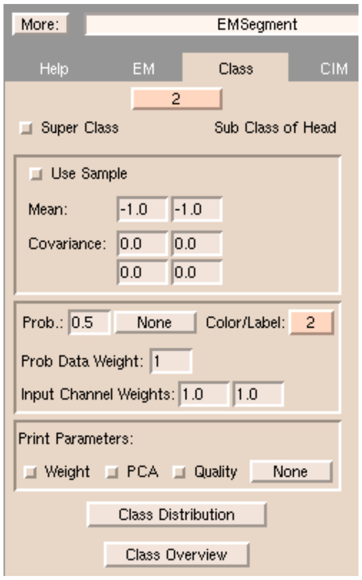

Figure 1.10: Parameters for the `Sub Class` structure.

Description of parameters for the `Sub Class` structure:

- `Mean` – Mean intensity value of the sample points.
- `Covariance` – Covariance matrix.
- `Prob.` – Expected proportion of the classes at the given level. The class probabilities for a level have to add up to 1. (e.g. Figure 1, level 2: ECC: 0.15; ICC: 0.85).
- `Prob. Atlas` – The probabilistic atlas to be used for the current subclass.
- `Color/Label` – Color and label of the current subclass in the resulting `LabelMap`.
- `Prob Data Weight` – Weight of the probabilistic atlas.
- `Input Channel Weights` – Weights of the input channels (channel 0, channel 1, etc.). The order of the channels here is the same as in the Input Greyscales

There are two ways of sampling a distribution:

- Manually: To set the class distribution (Mean and Covariance), click on the `Use Sample` button (so it is lowered) and pick a few (5-20) samples (that represent the ECC) from the lower image panel of the `Viewer` window by pressing `CTRL-Left Mouse Button` over the desired voxels. The `Mean` and `Covariance` matrix should change with each sample. [**Note**: this can also be done from the `EM Tab -> Step3 -> StepD` panel by clicking on the `Manual` button and selecting sample points the same way.] **Tip:** See section 1.6 on how to visualize where the pixel under the cursor falls on the intensity distribution graph.

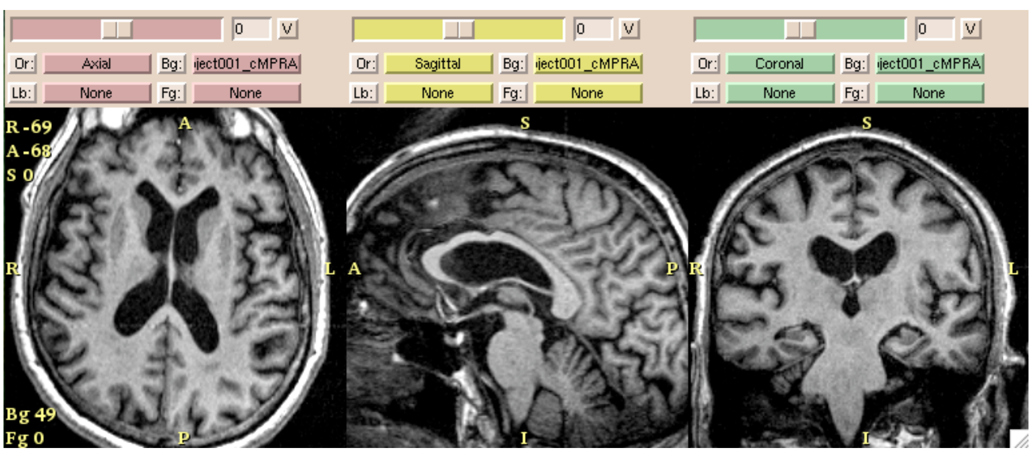

Figure 1.11: Lower Image Panel of the `Viewer` window.

- Automatically: By clicking on the `Auto` button in the `EM Tab -> Step3 -> StepD` panel. [**Note**: not recommended.]

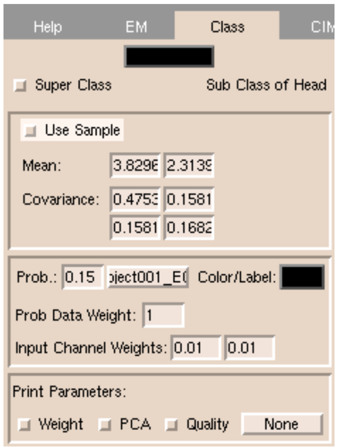

Figure 1.12: Parameters for the ECC subclass.

```
Set the following parameters:

       Mean:                      3.8296, 2.3139
       Covariance:                0.4753, 0.1581
                                  0.1581, 0.1682
       Prob:                      0.15
       Prob Atlas:                subject001_ECC
       Color/Label:               Black=0
       Prob Data Weight:          1
       Input Channel Weight:      0.01, 0.01

Note: The first channel (channel 0) is the MPRAGE, the second (channel 1) is the FLAIR.

Note: To exclude (set to label 0) all of the ECC, we want to put maximum weight on the ECC label map and little weight
on the input channels (0 weight will give an error, thus we set it to 0.01).
```

To set the label and color of the class click on the `Color/Label` button. A pop-up window will appear, where you can pick a color/label or define your own.

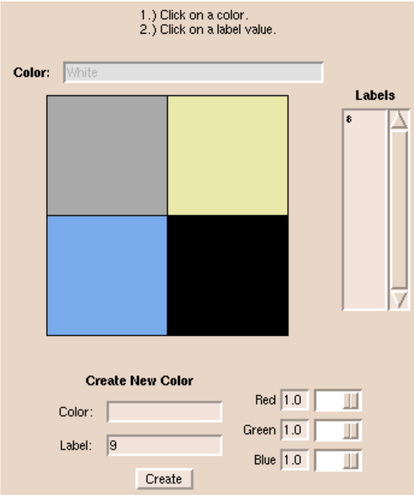

Figure 1.13: 'Select a Color' window with the predefined colors for the sample case. The default colors and labels are different.

```
The colors for the sample case are the following:

       Color = Gray      Label = 4       RGB = 0.6 0.6 0.6
       Color = White     Label = 8       RGB = 0.9 0.9 0.6
       Color = Csf       Label = 5       RGB = 0.4 0.6 0.9
       Color = Black     Label = 0       RGB = 0.0 0.0 0.0

Note: If you loaded the scene file subject001_scene01.xml, these colors will be predefined.
```

ICC Super Class

Change the current class to the second subclass. Click on the `Super Class` button (so it is lowered) to make it a super class. This super class will include the GM, WM, and CSF subclasses. For a description of the parameters refer back to the `Head` super class section.

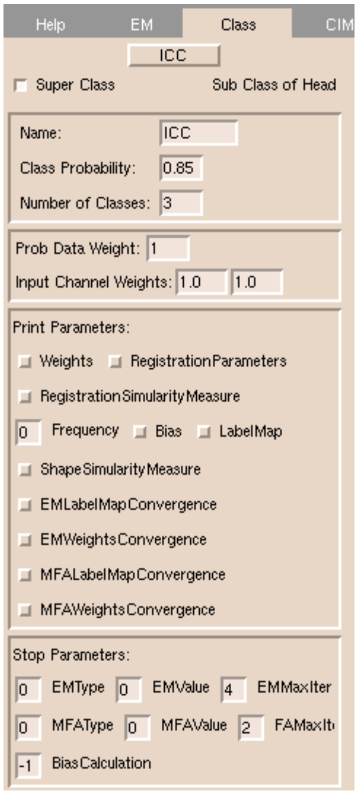

Figure 1.14: Parameters for the ICC super class.

```
Set the following parameters:

       Name:                     ICC
       Class Probability:        0.85
       Number of Classes:        3
       Prob Data Weight:         1.0
       Input Channel Weights:    1.0, 1.0
       Stop Parameters
               EMMaxIter:        4
               FAMaxIter:        2
```

GM Subclass

Change the current class to the first subclass under ICC. Using the method described in the ECC section, pick sample voxels to set the `Mean` and `Covariance`.

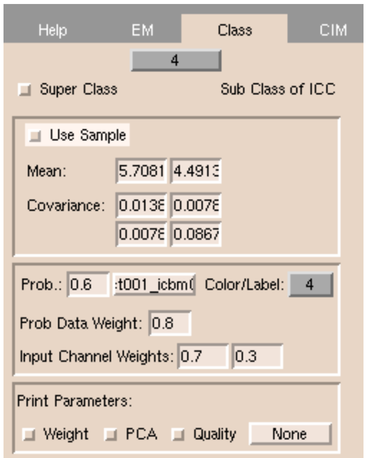

Figure 1.15: Parameters for the GM subclass.

```
Set the following parameters:

        Mean:                       5.7081, 4.4913
        Covariance:                 0.0138, 0.0078
                                    0.0078, 0.0867
        Prob:                       0.6
        Prob Atlas:                 subject001_icbmGRAY
        Color/Label:                Gray=4
        Prob Data Weight:           0.8
        Input Channel Weight:       0.7, 0.3
```

WM Subclass

Change the current class to the second subclass under ICC. Using the method described in the ECC section, pick sample voxels to set the `Mean` and `Covariance`.

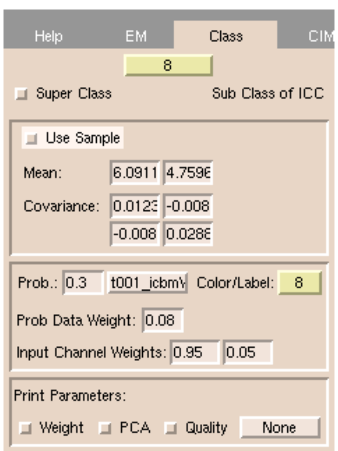

Figure 1.16: Parameters for the WM subclass.

```
Set the following parameters:

        Mean:                       6.0911, 4.7596
        Covariance:                 0.0123, -0.0084
                                    -0.0084, 0.0288
        Prob:                       0.3
        Prob Atlas:                 subject001_icbmWHITE
        Color/Label:                White=8
        Prob Data Weight:           0.08
        Input Channel Weight:       0.95, 0.05
```

CSF Subclass

Change the current class to the third subclass under ICC. Using the method described in the ECC section, pick sample voxels to set the `Mean` and `Covariance`.

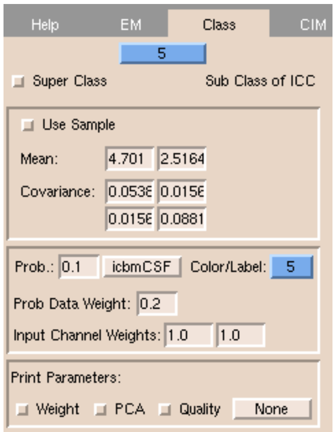

Figure 1.17: Parameters for the CSF subclass.

```
Set the following parameters:

       Mean:                     4.7010, 2.5164
       Covariance:               0.0538, 0.0156
                                 0.0156, 0.0881
       Prob:                     0.1
       Prob Atlas:               subject001_icbmCSF
       Color/Label:              Csf=5
       Prob Data Weight:         0.2
       Input Channel Weight:     1.0, 1.0
```

Subclass Probabilities

The probabilities of subclasses within a super class should add up to 1. To get an overview of the subclasses of a certain super class, click on the `Class Overview` button at the bottom of the `Class` tab. A pop-up window will display some of the subclass parameters. These parameters can also be changed in this window.

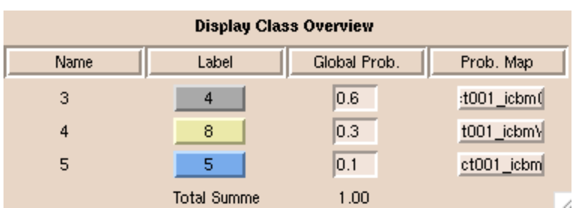

Figure 1.18: Class Overview window for the ICC super class.

1.6 Class Distributions

The class distributions of all the subclasses can be displayed by clicking on the `Class Distribution` button at the bottom of the `Class` tab. In the pop-up window click on the buttons on the top and bottom left corners to set the input channels and click on the labeled buttons to display the distributions.

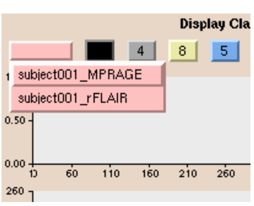

Figure 1.19: Setting the Input Channels in the Class Distribution window. Click on the top left and bottom left buttons to set the input channels.

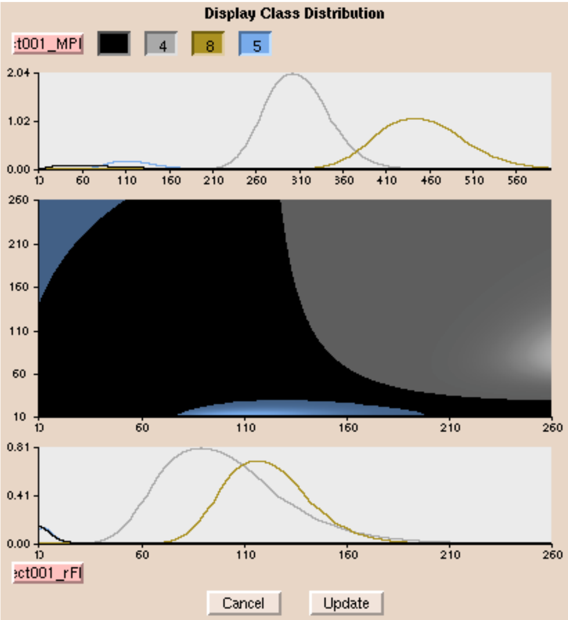

Figure 1.20: Class Distributions for the Sample Case.

To change the graph settings, such as the range of the axis click on the graph and change the settings in the pop-up window.

Tip: When picking the sample points for the distributions, a red line indicates in the `Display Class Distribution` window where the value of the pixel under the cursor falls. This can be a useful indicator of how the distribution would be affected by that pixel value.

1.7 EM Algorithm Settings

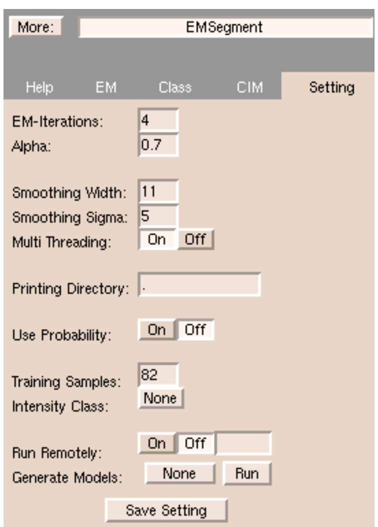

Figure 1.21: `Setting` panel with default values.

Description of parameters for the `Setting` tab:

- `EM-Iterations` – Same as `EMMaxIter` (obsolete field).
- `Alpha` – Influence of MFA on the definition of weights [0 = MFA is deactivated (good for high signal-to-noise ratio); >0 = importance of MFA, 1 = maximum importance (good for noisy images).
- `Smoothing Width` – When computing image inhomogeneity, a Gaussian filter is used to guarantee smoothness constraints of the field. This parameter determines the size of the filter (e.g. width = 20 means a 20x20x20 Gaussian filter).
- `Smoothing Sigma` – The variance of the filter. If sigma is large, the filter is similar to a box filter. If sigma is small, the filter relates to a median filter.
- `Multi Threading` – Use multiple threads.
- `Printing Directory` – The directory where intermediate results, as activated by `Print Parameters`, should be printed out.

- `Use Probability` – 1 = incorporate the global probabilities (`Prob.`) when plotting distributions in the "`Class Distribution`" window; 0 = plot the distributions only considering mean and covariance.
- `Training Samples` – Maximum value of the atlas images.
- ~~`Intensity Class`~~ – Deactivated field.
- `Run Remotely` – Run on a remote machine.
- `Generate Models` – Generate 3D models from the subclasses.

```
Set the following parameters:

      EM-Iterations:            4
      Alpha:                    0.3
      Smoothing Width:          19
      Smoothing Sigma:          9
      Multi threading:          On
      Use Probability:          On
      Training Samples:         100
      Intensity Class:          None
```

1.8 Running EMSegment

Once all the necessary parameters are set, run `EMSegment` by clicking on the `Run` button under the `EM` tab.

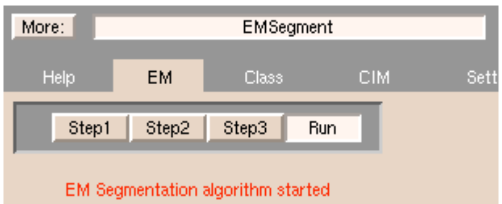

Figure 1.22: Running EMSegment. Click on the `Run` button in the `EM` tab.

The progress of the segmentation can be followed in the terminal window of `Slicer`.

```
vtkImageEMLocalAlgorithm: 2. E-Step
EMLocalAlgorithm: 1. EM - MF Iteration
EMLocalAlgorithm: 2. EM - MF Iteration
vtkImageEMLocalAlgorithm: M-Step

vtkImageEMLocalAlgorithm: 3. E-Step
EMLocalAlgorithm: 1. EM - MF Iteration
EMLocalAlgorithm: 2. EM - MF Iteration
vtkImageEMLocalAlgorithm: M-Step

vtkImageEMLocalAlgorithm: 4. E-Step
EMLocalAlgorithm: 1. EM - MF Iteration
EMLocalAlgorithm: 2. EM - MF Iteration
Elapsed time: 56.1925
EMLocalAlgorithm::RunAlgorithm: Finished
End vtkImageEMLocalSegmenter::HierachicalSegmentation
End vtkImageEMLocalSegmenter::HierachicalSegmentation
End vtkImageEMLocalSegmenter::HierachicalSegmentation
End vtkImageEMLocalSegmenter::HierachicalSegmentation
End vtkImageEMLocalSegmenterExecute
```

Figure 1.23: Slicer Terminal window messages.

Upon successful segmentation the following message is displayed in the `EM` tab.

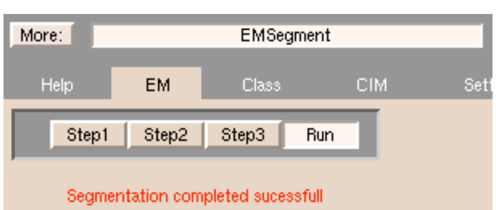

Figure 1.24: Completion message.

1.9 Results

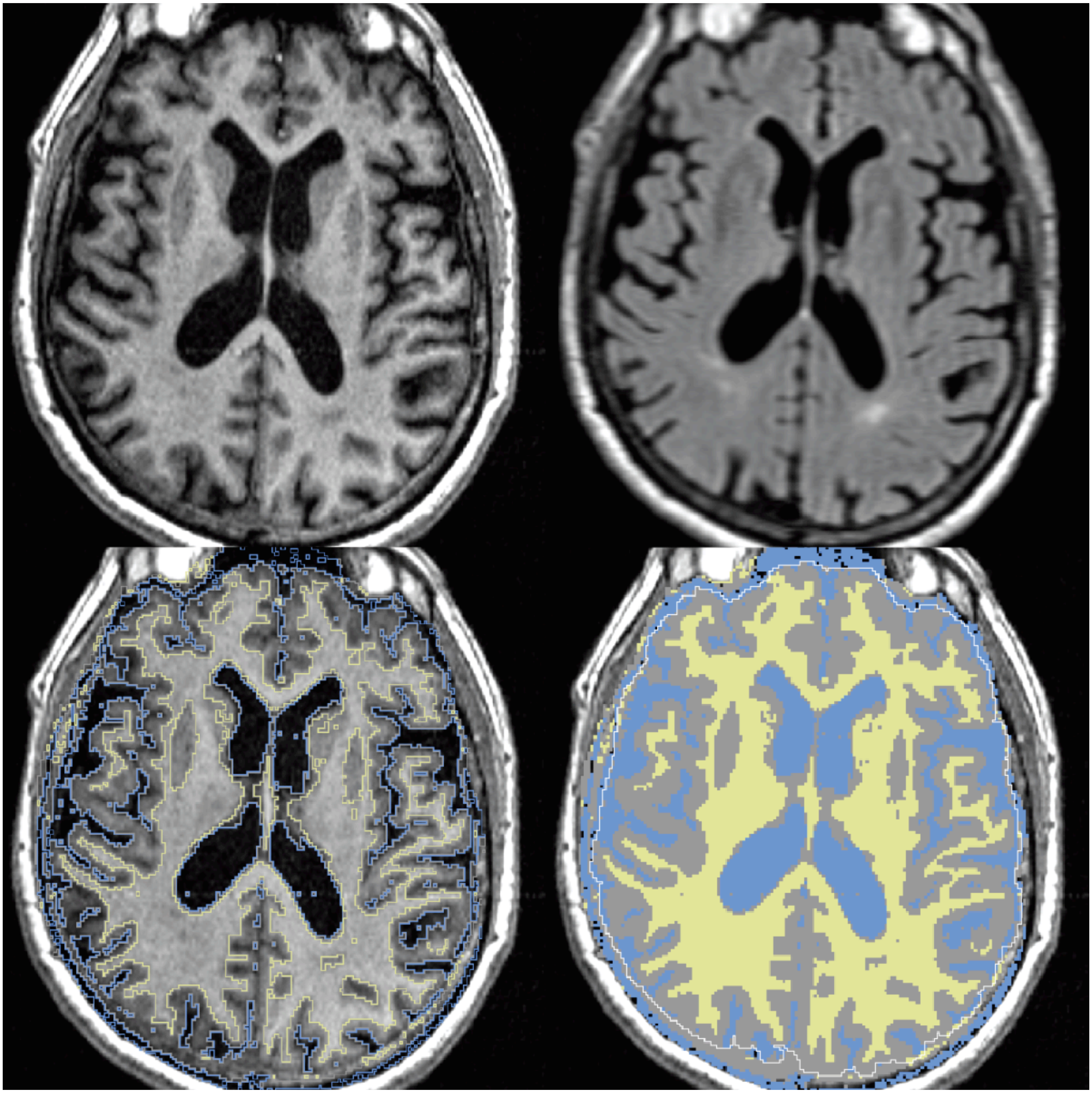

Figure 1.25: Segmentation results – MPRAGE, FLAIR, LabelMap and MPRAGE overlay, LabelMap with ICC outlined (white line).

Although, the current settings provide good results the following problems are apparent:

- Deep gray matter structures, such as the caudate and the thalamus are not properly segmented.
- CSF is slightly underestimated.
- There is brain matter outside of the ICC (white line).

Note: We can only define ECC as a probabilistic atlas and not as a mask. Therefore, some parts of the ECC are still classified as brain matter. These extra pixels have to be masked out with the ECC. Refer to the `Slicer` website on masking.

In the next chapter we describe how to add a deep gray matter subclass to improve the segmentation of the caudate and the thalamus.

Ways to improve the segmentation:

- Depending on the weight of the probabilistic atlases, the class distributions have the largest impact on the outcome of the segmentation. If one of the classes is way over or underestimated, changing the class distributions should be first step. A particular distribution can be broadened, for example, by picking additional sample points that lie towards the edge of the distribution. Use the `Display Class Distribution` window (section 1.6) to see where the current pixel under the cursor falls and how the distributions look like.
- The weights of the input channels also have a large impact. Put more weights on the input channel that has the largest contrast for the particular subclass.
- Changing the relative probabilities of the subclasses within a super class has little effect, thus these probabilities should be used for fine-tuning.

1.10 Saving the Results and EMSegment Parameters

See the `Slicer` tutorial on saving the segmentation results.

To save the EMSegment parameters, go to the `Setting` tab and click on the `Save Setting` button. The XML file can then be reloaded to restore the saved parameters (`Menu Window -> File Menu -> Open Scene`).

2 ADDING THE DEEP GRAY MATTER SUBCLASS

In this section we'll describe the strategy to further refine the segmentation by introducing the Deep Gray Matter subclass. Deep gray matter structures (caudate, putamen, thalamus) have different intensity distributions than cortical gray matter. These differences can be modeled by further dividing GM into subclasses and building unique distributions for each of these classes.

> **Note:** This section gives only an overview of the general strategy and it is up to the reader to refine the segmentation; therefore, we do not provide specific parameters.

2.1 General Strategy

1. Continue from the previous segmentation.
2. Set up the class hierarchy (add GM super class, and CGM and DEEPGM subclasses, see Figure 2.1).
3. Build the intensity distribution for the new subclasses.
4. Run `EMSegment`.
5. Refine the parameters and the class distributions and rerun `EMSegment` until satisfactory segmentation is achieved.
6. Save the parameters into a scene file.

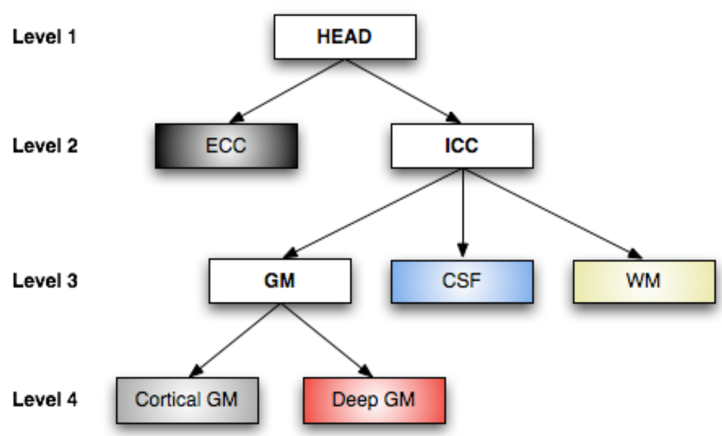

Figure 2.1: Class hierarchy for CGM, DEEPGM, WM, and CSF segmentation (HEAD, ICC, and GM are super classes).

2.2 Files Used

- Subject Files:
  - subject001_MPRAGE.nii.gz
  - subject001_rFLAIR.nii.gz
- Atlas Files (registered to MPRAGE):
  - subject001_icbmCSF.nii.gz
  - subject001_icbmGRAY.nii.gz
  - subject001_icbmDEEPGM.nii.gz
  - subject001_icbmWHITE.nii.gz

- Label Maps:
    - subject001_ECC.nii.gz

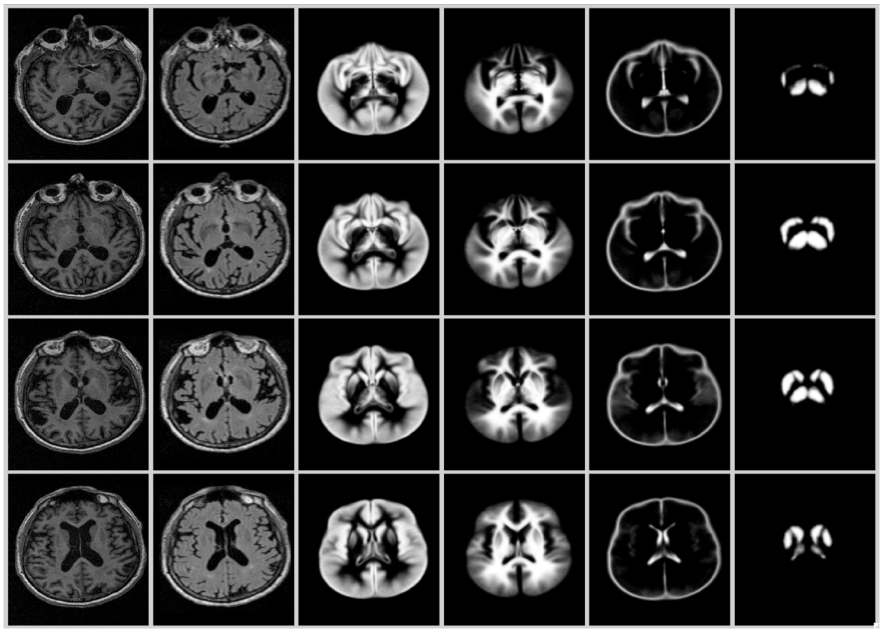

Figure 2.2: Sample slices from the MPRAGE and FLAIR subject images and the GM, WM, CSF, and DEEPGM probabilistic atlases.

2.3 Loading the DEEP GM Atlas Image

Load the DEEP GM probabilistic atlas image into `Slicer`.

```
Manually via Data Panel -> Add Volume:
      subject001_icbmDEEPGM.nii.gz
```

2.4 Modifying the Class Hierarchy (Class Tab)

GM Super Class

Change the current class to the GM subclass. Click on the `Super Class` button (so it is lowered) to make it a super class. This super class will include the CGM and DEEPGM subclasses. For a description of the parameters refer back to the `Head` super class under section 1.5.

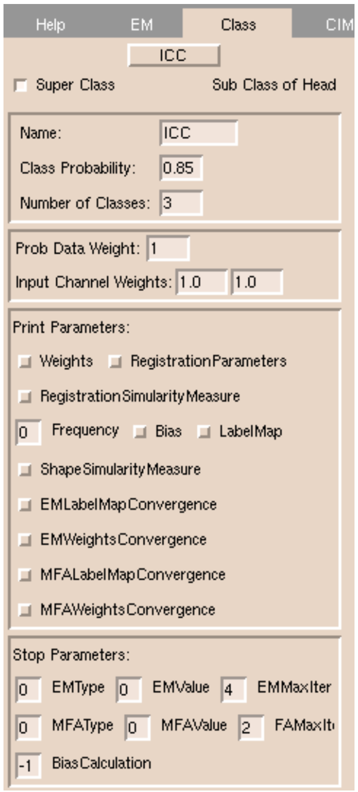

Figure 2.3: Parameters for the GM super class.

CGM Subclass

Change the current class to the first subclass under ICC. Using the method described in the ECC section, pick sample voxels to set the `Mean` and `Covariance`.

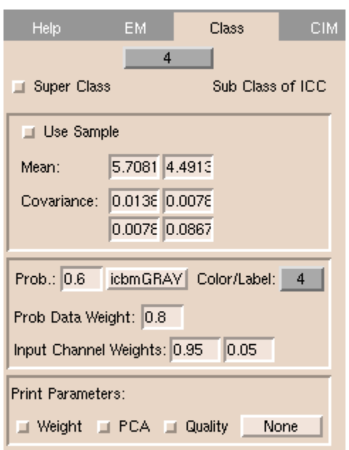

Figure 2.4: Parameters for the CGM subclass.

DEEPGM Subclass

Change the current class to the first subclass under ICC. Using the method described in the ECC section, pick sample voxels to set the `Mean` and `Covariance`.

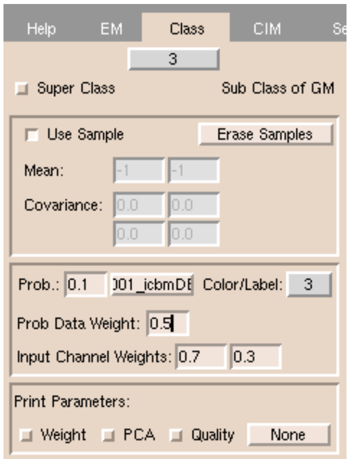

Figure 2.5: Setting the parameters for the DEEPGM subclass.

2.5 Running EMSegment

Run `EMSegment` by clicking on the `Run` button under the `EM` tab. Refine the parameters until the segmentation quality is satisfactory.

3 ADDING THE WHITE MATTER HYPERINTENSITY SEGMENTATION

In this section we'll further refine the segmentation by dividing the White Matter into Normal Appearing White Matter (NAWM) and White Matter Hyperintensities (WMH). WMH is segmented in a separate step and then the results are combined with the previous segmentation.

> **Note:** This section gives only an overview of the general strategy and it is up to the reader to refine the segmentation; therefore, we do not provide specific parameters.

3.1 General Strategy

Adding in the NAWM and WMH subclasses under the hierarchy used so far does not produce satisfactory results. Segmenting WMH and GM together in the same segmentation step tends to result in a considerable amount of GM misclassifications as WMH. To circumvent such misclassifications, the following strategy can be used:

1. Load the FLAIR image.
2. Set up the class hierarchy for the separate WM segmentation (WM super class, and NAWM and WMH subclasses, see Figure 3.1).
3. Build the intensity distribution for the new subclasses.
4. Run `EMSegment`.
5. Refine the parameters and the class distributions and rerun `EMSegment` until satisfactory segmentation is achieved.
6. Merge the WMH segmentation and the previous result to get the final segmentation that includes all tissue types.

Note: This strategy works well for the sample dataset used; however, it might not be suitable for different datasets. It is used to demonstrate an approach that might lead to satisfactory results.

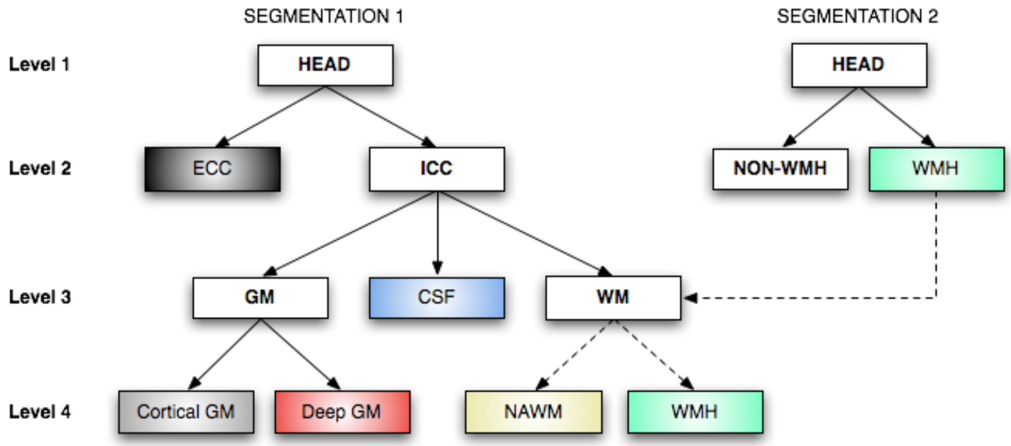

Figure 3.1: Class hierarchy for NAWM and WMH segmentation (SEGMENTATION 2) and merging the results with the previous segmentation.

R e f e r e n c e s

1. K.M. Pohl , S. Bouix, R. Kikinis, W.E.L. Grimson, “Anatomical Guided Segmentation with Non-Stationary Tissue Class Distributions in an Expectation-Maximization Framework.” In Proc. ISBI 2004: IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Arlington, VA, USA, pp. 81-84, 2004.
http://people.csail.mit.edu/pohl/publications/pohl-isbi-2004.pdf