

Final Draft

EM Segmentation of the Phalanx Bones of the Hand Tutorial

September 12, 2007

Austin J. Ramme, B.S. ¹
Vincent A. Magnotta, Ph.D. ²
Nicole Grosland, Ph.D. ³

[1]University of Iowa Carver College of Medicine, [2] University of Iowa Department of Radiology,
& [3] University of Iowa Department of Biomedical Engineering

A special thanks to Kilian Pohl, Ph.D. of Massachusetts Institute of Technology for his assistance.

Goal of Tutorial:

To demonstrate the application of the EMSegment module in Slicer2.7 for segmentation of the phalanx bones of the hand.

Table of Contents:

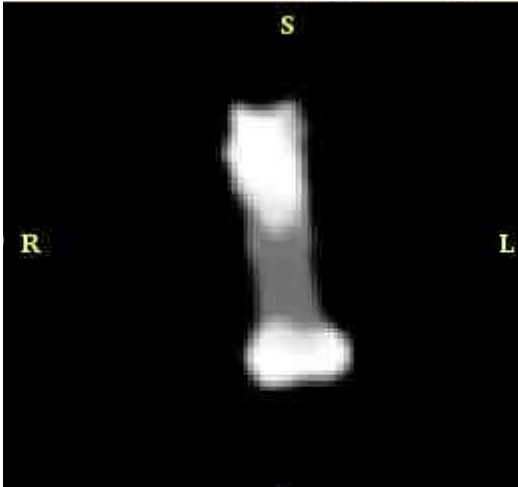
Step 1: Selection of Volumes.....	2
Step 2: Registration of Atlas to Volumes.....	3
Step 3: Load Volumes into Slicer2.7.....	
Step 4: Developing EMSegment Intensity Parameters.....	
Step 5: Optimize EMSegment Parameters.....	
Step 6: Saving EMSegment Results.....	
Step 7: Generate Models.....	
Step 8: Apply to Remaining Phalanx Bones.....	
Other Applications.....	
Appendix.....	

Step 1: Selection of Volumes

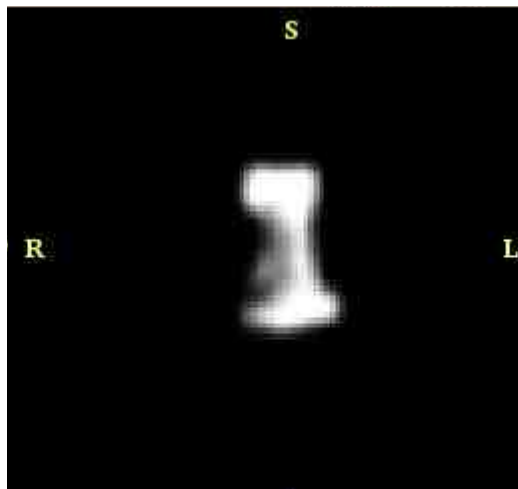
The following volumes were selected to segment the three phalanx bones of the index finger:



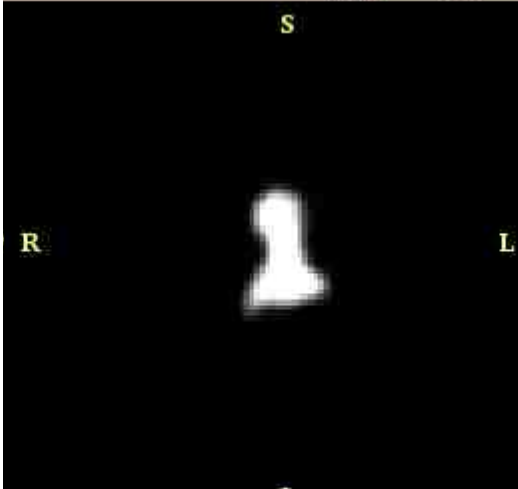
Title: CT Subject Volume
File Type: Unsigned 8-bit .hdr
Purpose: The image to segment



Title: Proximal Phalanx Probability Map
File Type: Unsigned 8-bit .hdr
Purpose: Probability map
Generation: This image was generated from a manual trace defined on the atlas image. The manual tracing was performed using BRAINS2. The ROI was converted to a mask and subsequently an image. The resulting image was filtered using the itkDiscreteGaussian utility with a Variance and Width both set to 4.0 and a data type of float-single. The resulting smoothed image was saved as an unsigned-8bit image for use with the EM Segmentation algorithm.



Title: Medial Phalanx Probability Map
File Type: Unsigned 8-bit .hdr
Purpose: Probability map
Generation: This image was generated from a manual trace defined on the atlas image. The manual tracing was performed using BRAINS2. The ROI was converted to a mask and subsequently an image. The resulting image was filtered using the itkDiscreteGaussian utility with a Variance and Width both set to 4.0 and a data type of float-single. The resulting smoothed image was saved as an unsigned-8bit image for use with the EM Segmentation algorithm.



Title: Distal Phalanx Probability Map

File Type: Unsigned 8-bit .hdr

Purpose: Probability map

Generation: This image was generated from a manual trace defined on the atlas image. The manual tracing was performed using BRAINS2. The ROI was converted to a mask and subsequently an image. The resulting image was filtered using the itkDiscreteGaussian utility with a Variance and Width both set to 4.0 and a data type of float-single. The resulting smoothed image was saved as an unsigned-8bit image for use with the EM Segmentation algorithm.

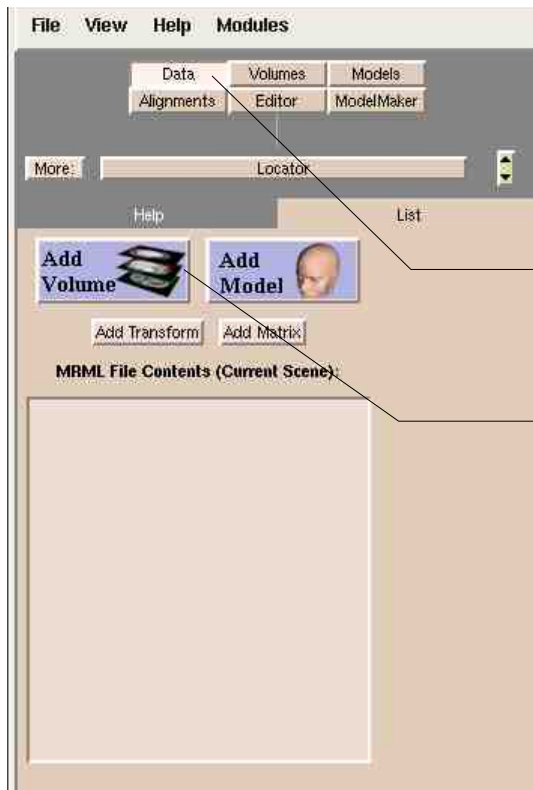
Note: To segment the remaining phalanx bones of the hand, additional probability maps for each phalanx bone in each finger should be included. An additional nine probability maps would be necessary.

Step 2: Registration of Atlas to Volumes

A three stage registration process was used to co-register the atlas image to the subject image. This was done using itk registration procedures using a landmark identification, thin plate spline registration, and finally a higher order Thirion Registration. These tools are in two programs developed as part of IA-FEmesh package. This technique worked well for our applications; however, other registration techniques may also work equally as well. To see the command line code used for our registration, please refer to the appendix.

Step 3: Load Volumes into Slicer2.7

Volumes need to be loaded in Slicer2.7 to perform the EMSegment module. To load the images, follow the following steps:



1. Click the Data button

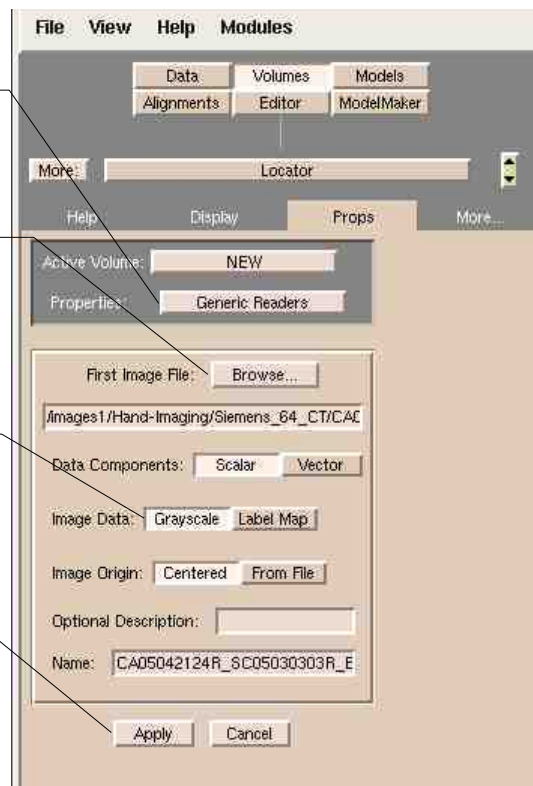
2. Click the Add Volume button

3. Select Generic Reader

4. Select the desired file

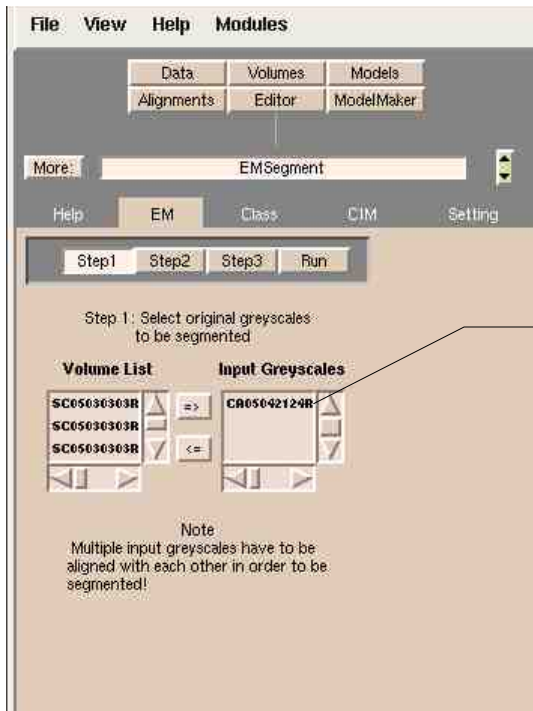
5. Select Grayscale (Label Map should be selected for resulting saved EMSegment Label Maps)

6. Click the Apply button



Step 4: Determining EMSegment Intensity Parameters

After all of the necessary volumes have been loaded, the EMSegment module should be initiated. To reach the EMSegment module select Modules=>Segmentation=>EMSegment. The following steps describe our parameters for this procedure:



1. Move the subject volume to the Input Greyscales list using the provided arrow button

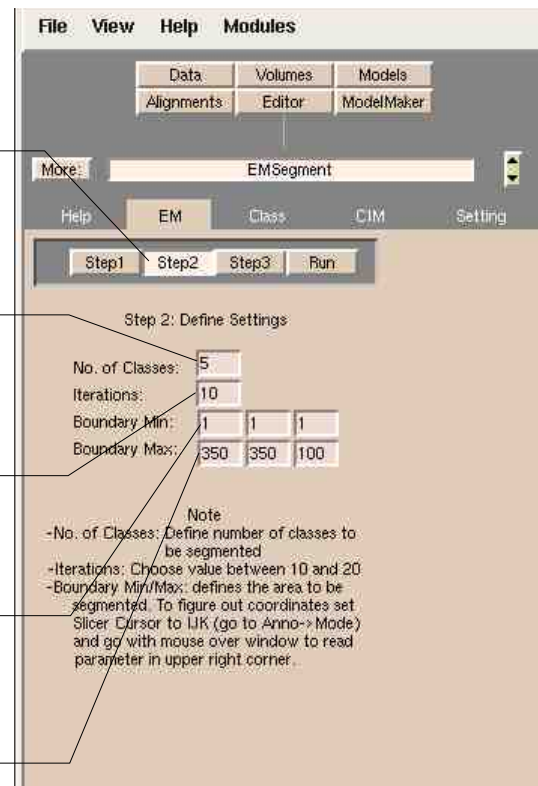
2. Click Step2 button

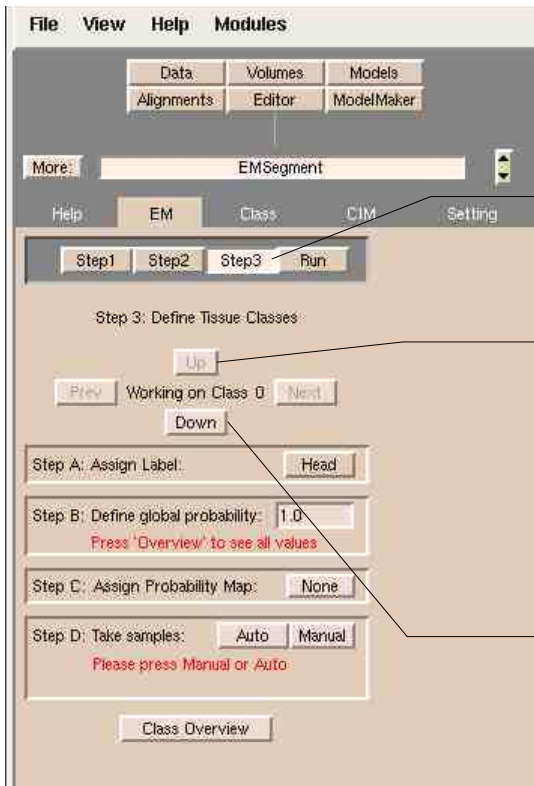
3. Input the number of classes. In our case, we have five: background, soft tissue, proximal phalanx, medial phalanx, and distal phalanx.

4. Input the number of iterations; we use 10.

5. Input the minimum boundary values for the image.

6. Input the maximum boundary values for the image.





7. Click Step3

8. If Up is pressed, then this takes you to the Super Class.

The default values of the Super Class do not need to be altered for our purposes with the exception of adding a value of "0" to the InhomogeneityInitialDataNames entry field.

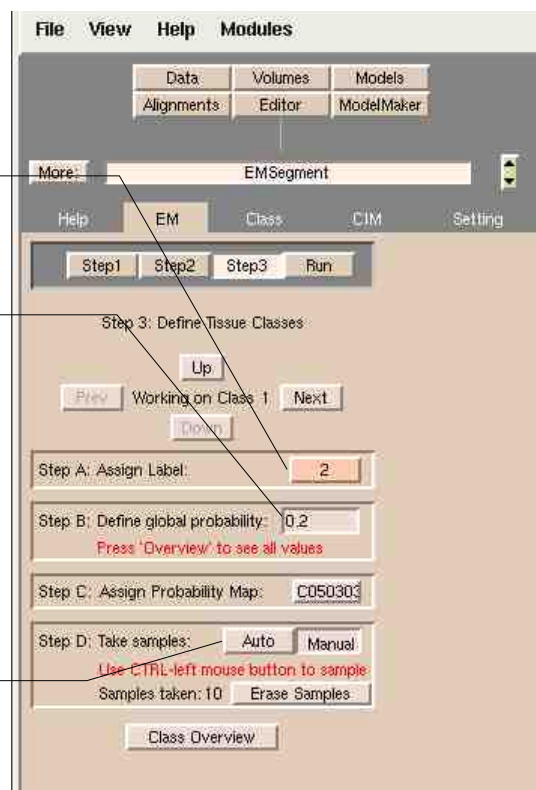
9. The down button will return you to the individual class areas.

10. For each class, assign a label.

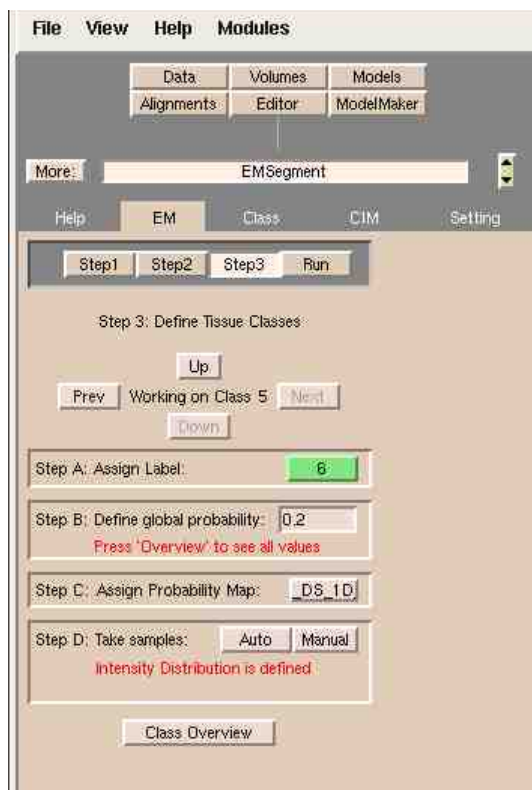
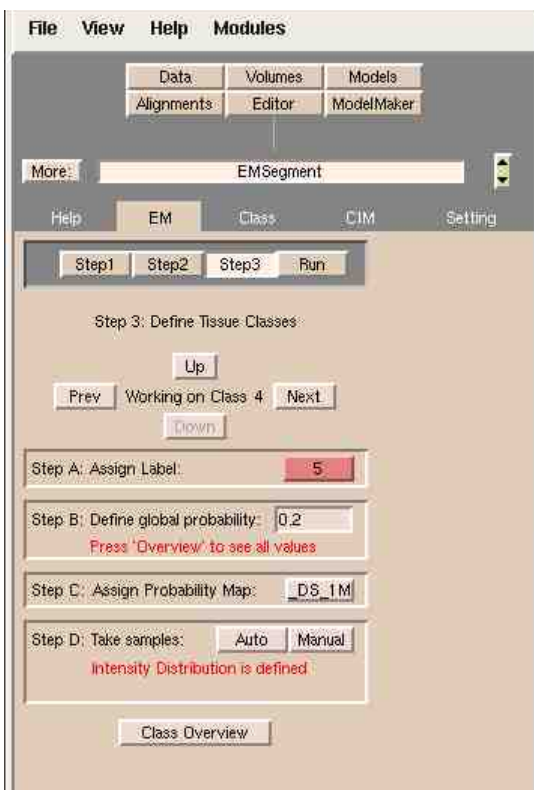
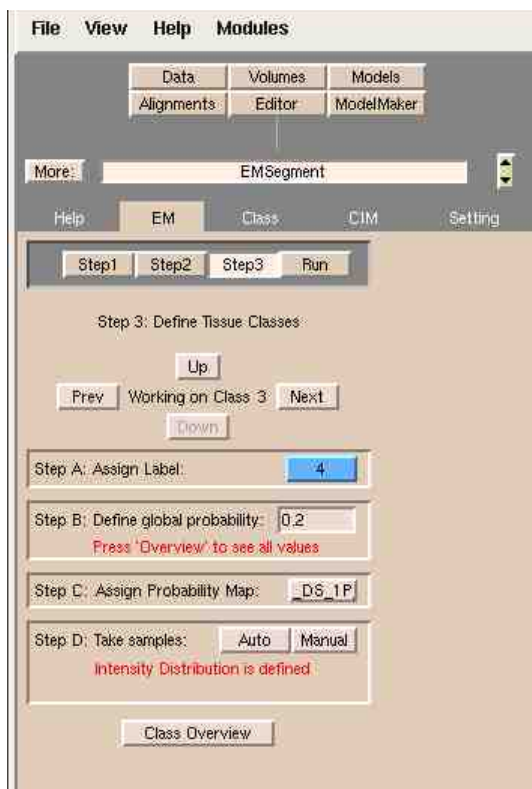
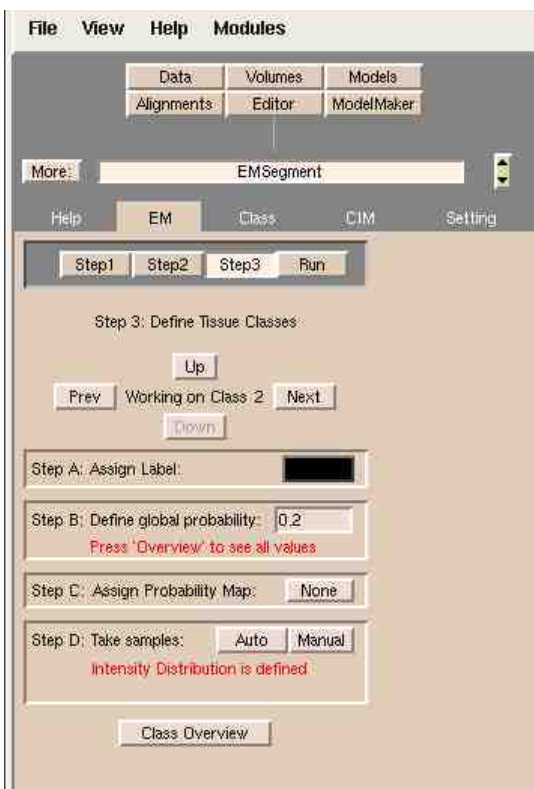
11. For each class, input a value for the global probability. The sum of the classes for the global probability should be one; we choose 0.2 since we have five classes.

12. For each class, perform either an auto sampling or a manual sampling of the class's region of interest to develop a set of intensity values for the EMSegment process. For manual sampling, hold the control button down as you select voxels of interest with the mouse.

To proceed to the next class click the Next button.



The following screen shots show the values entered for the remaining classes:

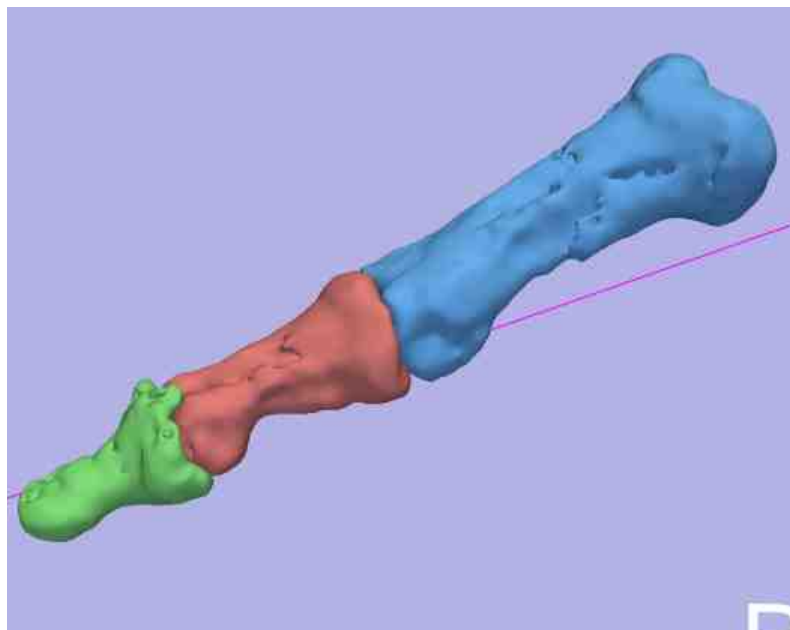


13. Press the Run button to perform the EMSegment process after all values have been entered.

The following images represent the raw output from the EMSegment process. Note the visible gaps and holes within the label map. Optimizing a few parameters can remove most of these issues.



EMSegment Label Map; Note the irregularities present in the distal phalanx.

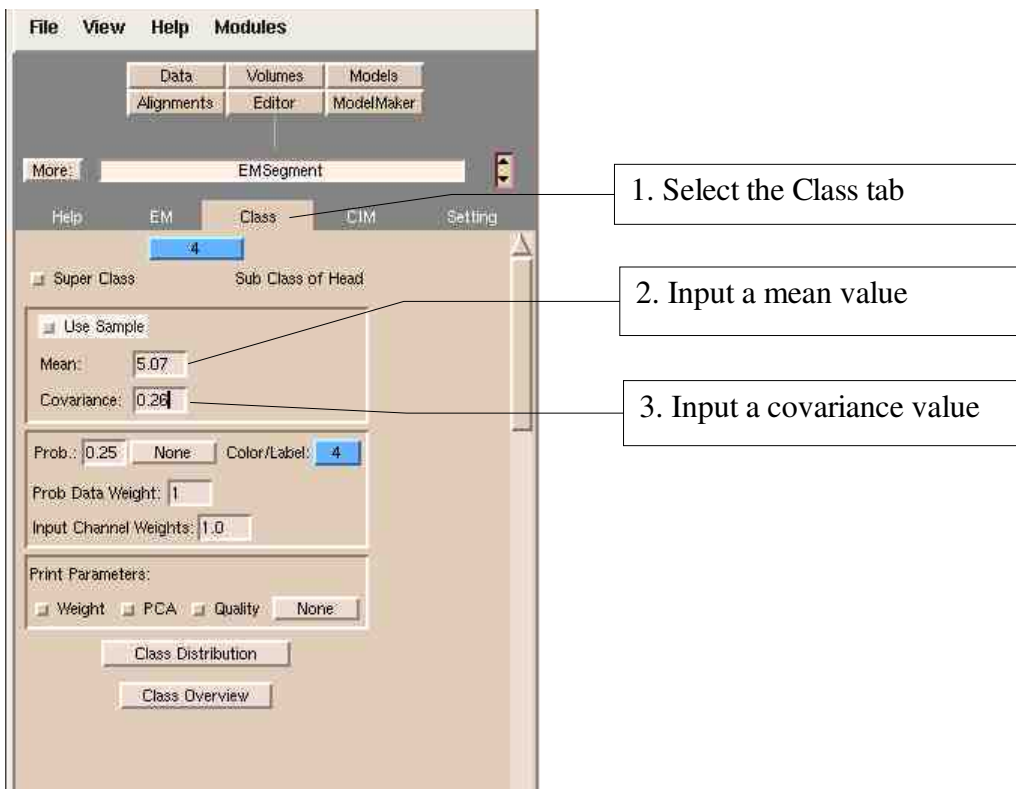


3-D Model of the raw EMSegment results. Note the surface irregularities.

Step 5: Optimize EMSegment Parameters

I. Streamlining the Intensity Sampling Procedure

The auto and manual sampling process can become a very time consuming step in the EMSegment module and also creates difficulty in obtaining consistent results. To bypass this step, click on the “Class Tab” and manually input the mean and covariance values. Mean and covariance values must be entered for each class. A starting point for obtaining ballpark values would be to run either the manual or auto sampling process and then check the output in this tab. After ballpark values have been established, incremental changes in these values can be used to optimize the resulting Label Map. Sample values and an optimized map can be seen below. The steps for this type of optimization are as follows:



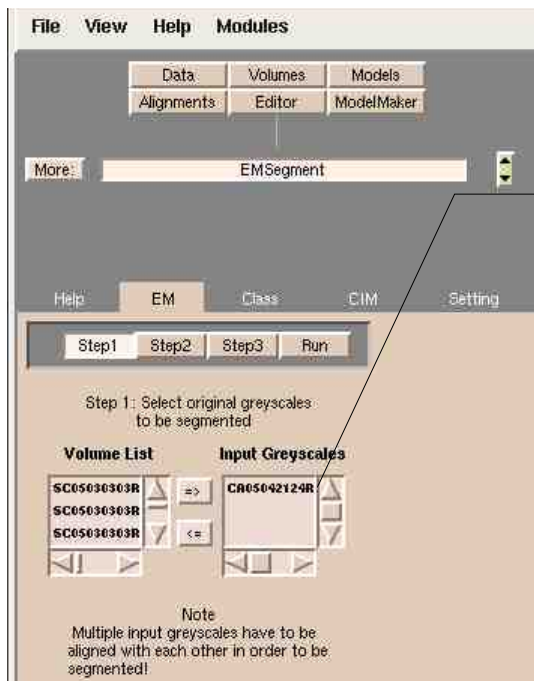
The following table displays the optimized values for the bones of the index finger, which were obtained using an iterative process:

<u>Label Map Class</u>	<u>Optimized Mean Value</u>	<u>Optimized Covariance Value</u>
Soft Tissue	4.71	1.76
Background	1.46	0.94

Proximal Phalanx - 1 st Finger	5.5	0.25
Medial Phalanx - 2 nd Finger	5.5	0.25
Distal Phalanx - 3 rd Finger	5.5	0.25

II. Ensuring that Both Intensity and Atlas Information are Utilized

The following steps should be used to ensure that both intensity and label map information are utilized. In Slicer2.7, the user is able to assign a probability map to a super class. By assigning the probability map to the super class, the EMSegment module identifies the specified region and then uses intensity information to determine whether the region is bone or other tissue.



1. Move the subject volume to the Input Greyscales list using the provided arrow button

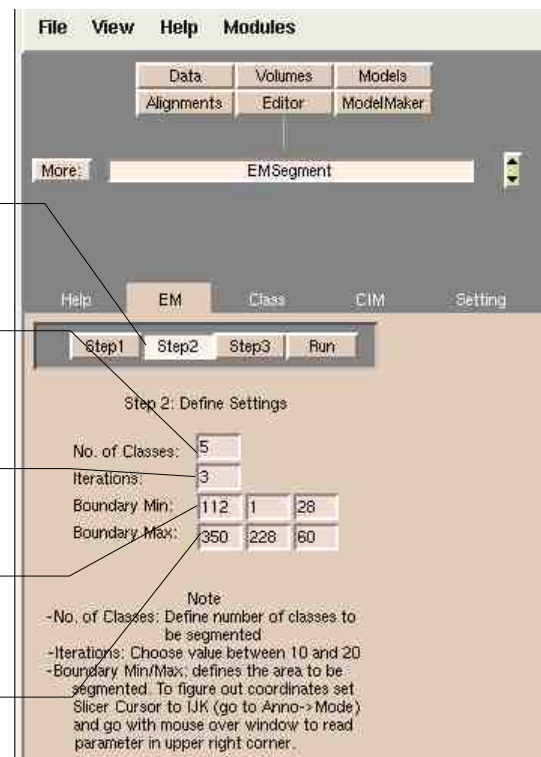
2. Click Step2 button

3. Input the number of classes. In our case, we have five: background, soft tissue, proximal phalanx, medial phalanx, and distal phalanx.

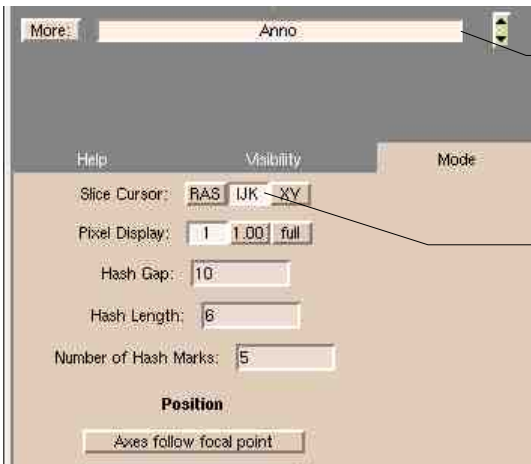
4. Input the number of iterations; we use 3.

5. Input the minimum boundary values for the image (see below).

6. Input the maximum boundary values for the image (see below).



To maximize the efficiency of the EMSegment process, establishing the boundary values for the segmentation is essential. The following shows how to determine the IJK coordinates for each image that is to be segmented.

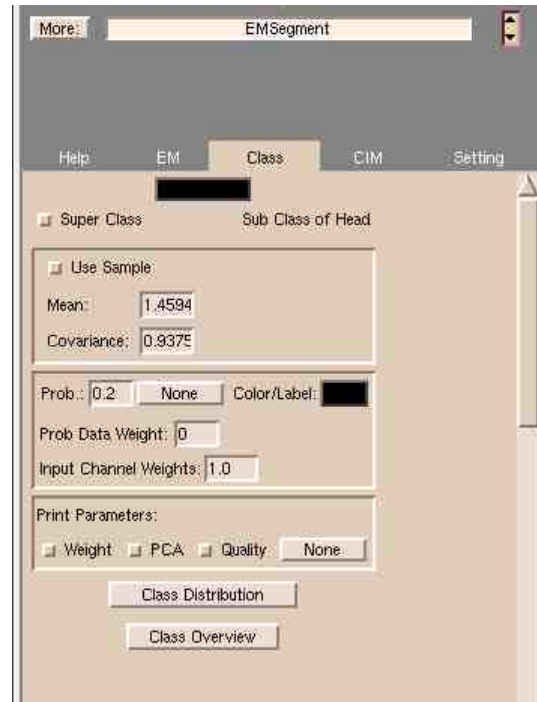
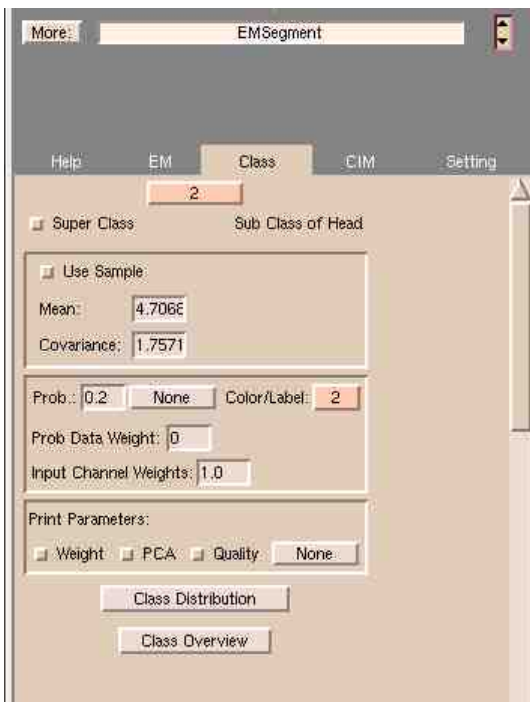


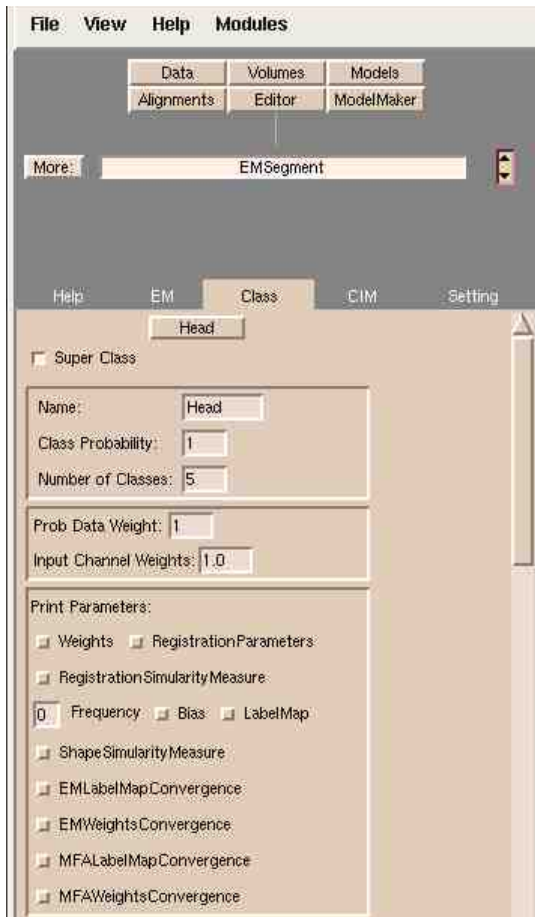
Select the Anno option to determine the max and min boundaries for segmentation.

Select the IJK coordinate system. Then determine your dimensions using the image displays.

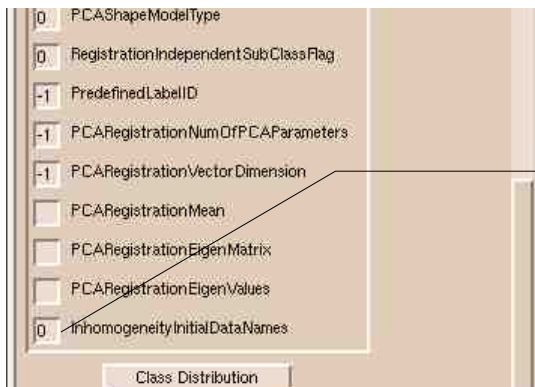
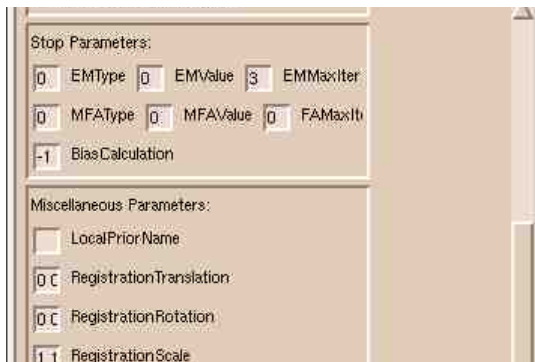
The first two classes (background and soft tissue) are assembled in a manner identical to the previous example. The Class tab screens for each class are shown below.

7. Enter values for the soft tissue and background classes as shown below.

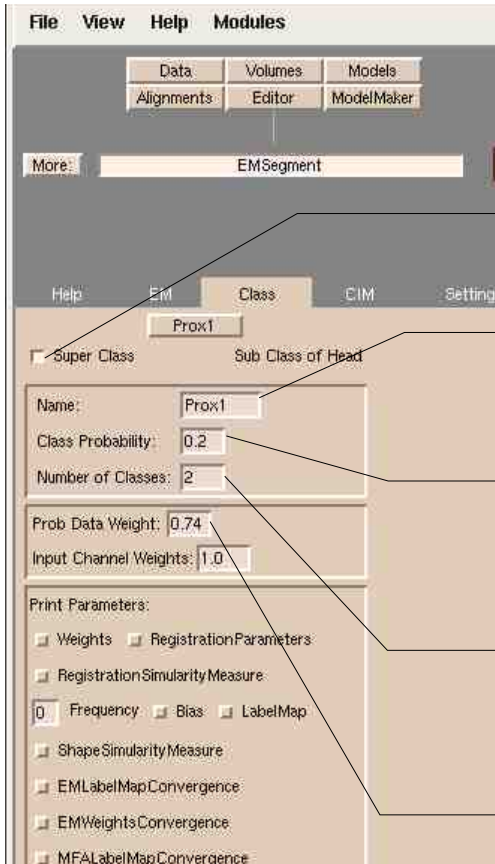




Under the Class tab, select the Head superclass. All of the default settings should be sufficient for our segmentation with the exception of the InhomogeneityInitialDataNames. Change this value as shown below.



8. Enter a zero in this data field.



Each phalanx bone requires its own superclass. The following shows how to setup the superclass for each phalanx bone.

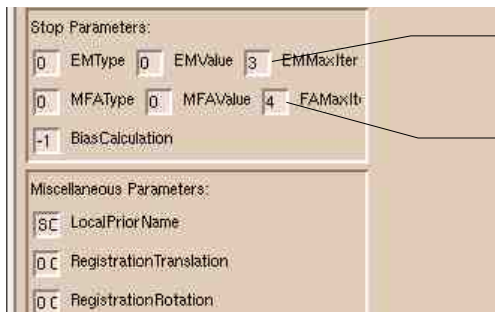
9. Click on the Super Class button.

10. Give the Super Class a name. This super class is for the proximal phalanx of the the index finger.

11. Set the class probability to 0.2 since we are using five classes.

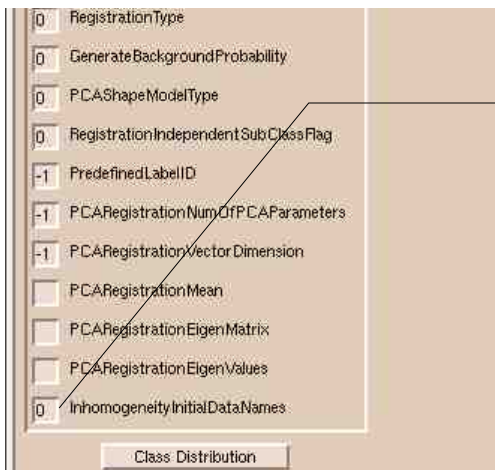
12. For each phalanx super class, create two subclasses. The subclasses represent a bony region and non-bony region.

13. Use a value of 0.74 for the probability weight data.



14. Use three iterations here.

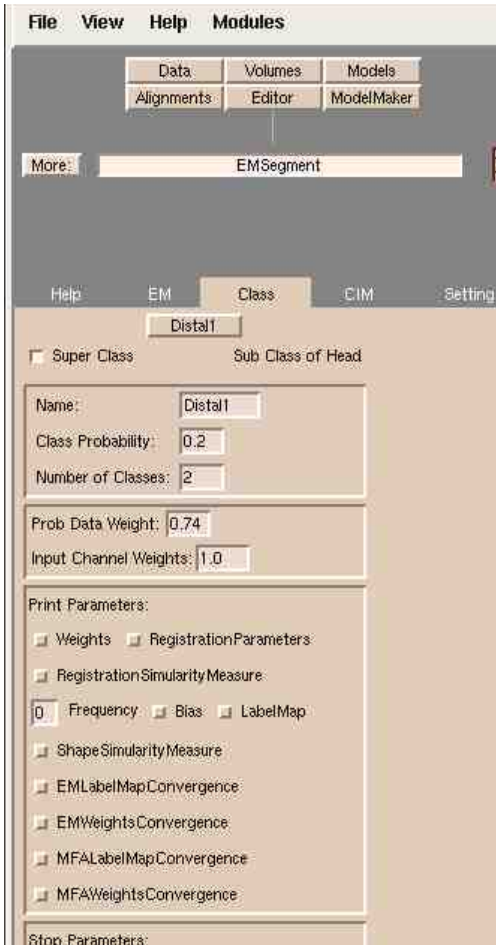
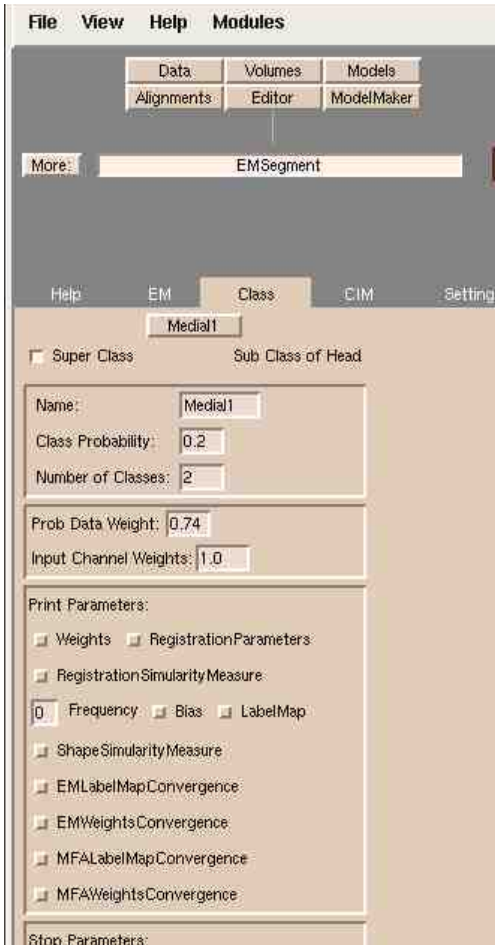
15. Use four iterations here.



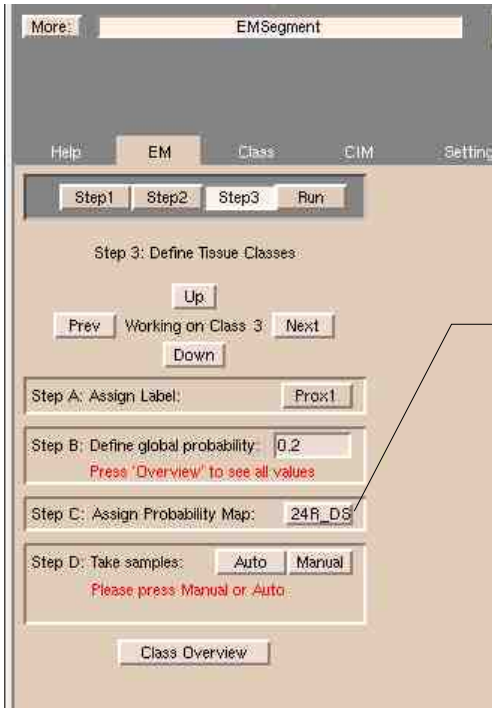
16. Enter a zero in this field.

The method to set up the proximal phalanx super class was demonstrated above. Use the same values to set up the super classes for the medial and distal phalanx bones. The following images show these super classes.

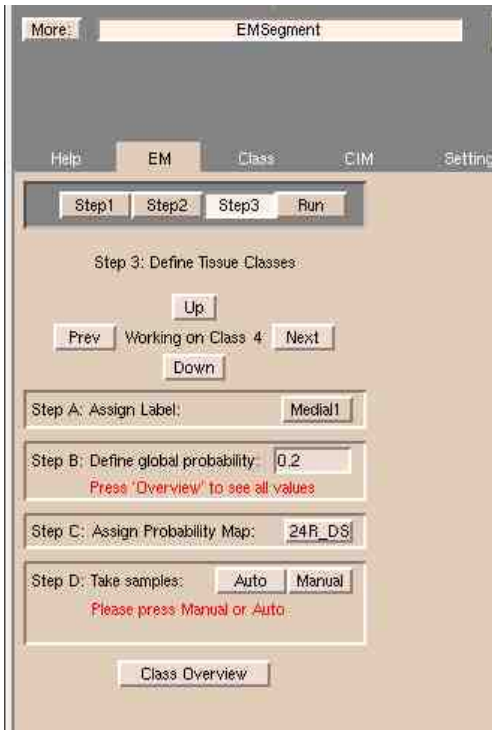
17. Create super classes for the medial and distal phalanx bones using the same methods as described for the proximal phalanx bone.



After the super classes have been created, return to the Step3 portion of the EMSegment module and assign the probability map for each phalanx bone superclass (prox1, medial1, and distal1).



18. Assign a probability map to each phalanx bone superclass (prox1, medial1, and distal1). The other two superclass entries can be seen below.



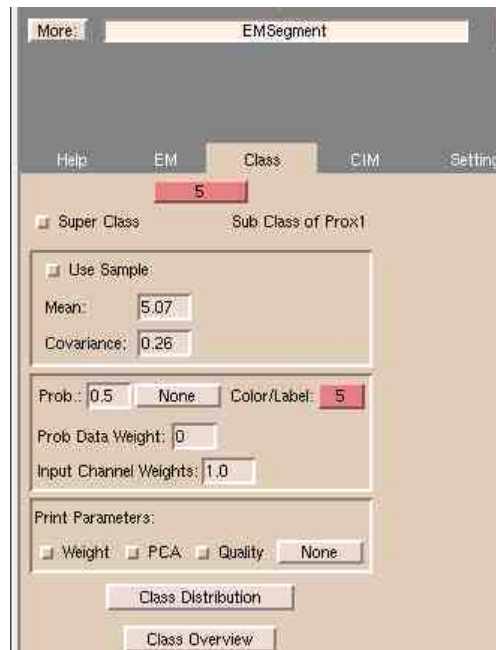
Within each phalanx bone super class, one subclass should be defined using predetermined values for bone. The three bone subclasses that were created are shown below.

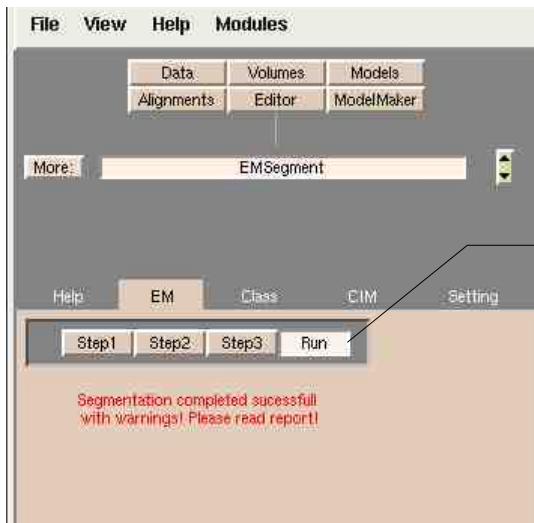
19. Create a bone subclass within each of the phalanx bone super classes. Use the values shown below.



The remaining subclass within each phalanx bone superclass should be reserved for information that does not reflect the intensity values of bone. An example is shown below:

20. Create a subclass for each phalanx bone superclass that is reserved for information that is not bone.





21. Select run.

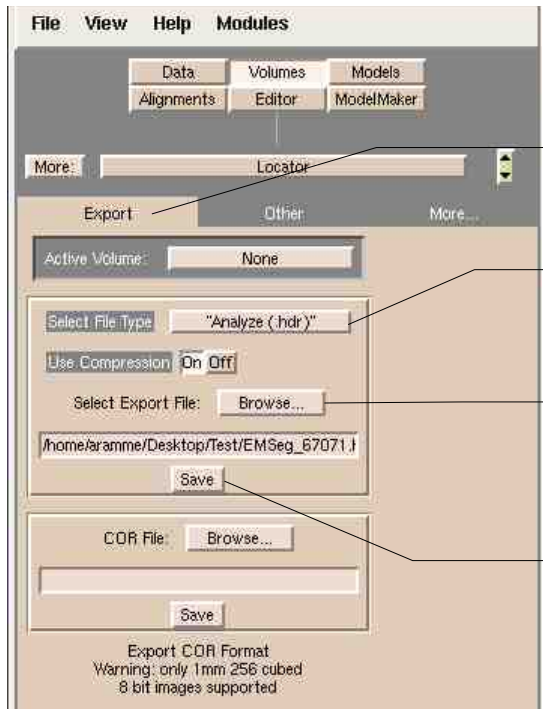
The following images show the resulting label map that is generated. Note that the trabecular bone within each phalanx bone is not labeled as bone. This issue can be resolved by exporting the label map to the BRAINS2 software and performing a Mask to ROI conversion followed by a ROI to Mask conversion. This process will fill the regions to give a final product of solid bone.



Final label map after the EMSegment module. The hollow regions in the trabecular bone can be filled using the BRAINS2 software as described above.

Step 6: Saving EMSegment Results

After the EMSegment module has completed, the EMSegment label map should be saved. The following steps outline this process.



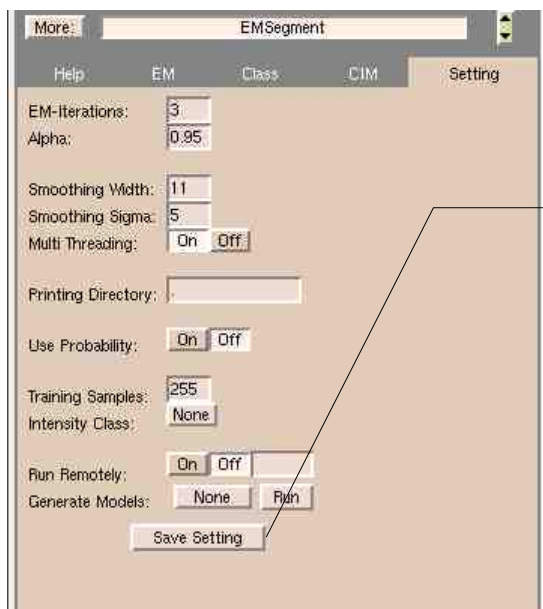
1. Click on the Export tab

2. Select the .hdr file type

3. Select the directory and file name

4. **Important!** Be sure to click this Save button or your results will not be saved

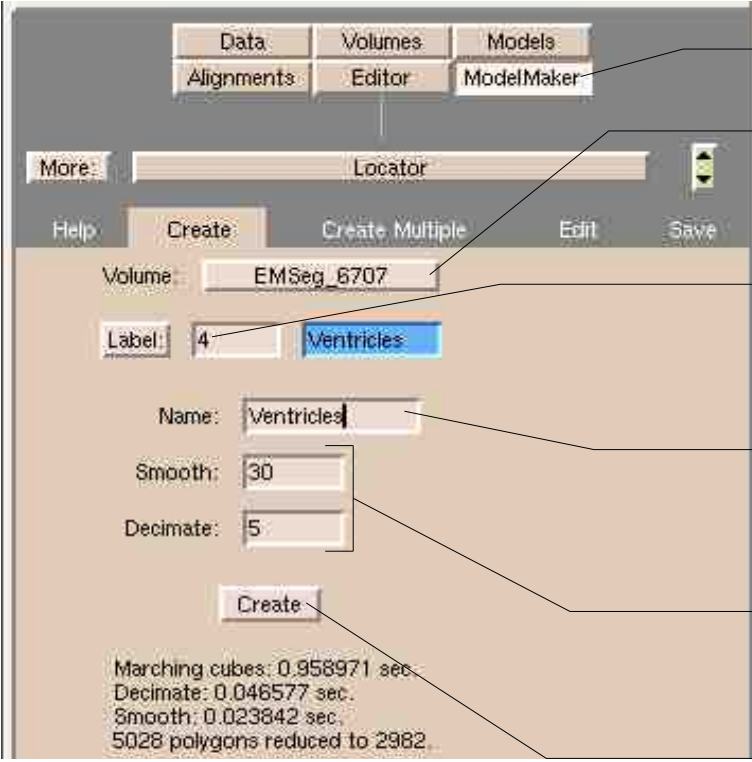
To simplify future EMSegment procedures, a file containing all of the necessary parameters can be saved as follows:



To save all of the loaded files and EMSegment parameters click the Save Setting button.

Step 7: Generate Models

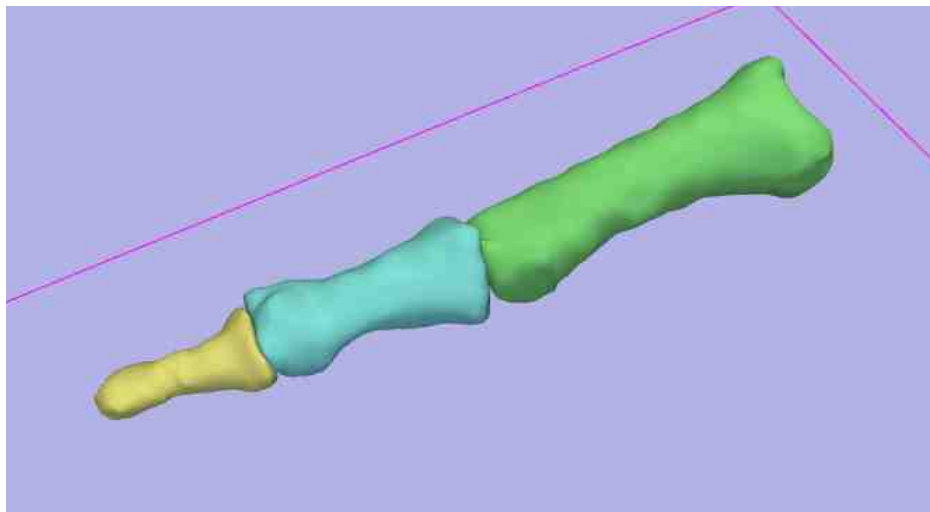
To generate models from an image that has undergone the EMSegment process simply follow the instructions below.



The screenshot shows the EMSegment software interface. At the top, there are tabs for 'Data', 'Volumes', and 'Models'. Below these are 'Alignments', 'Editor', and 'ModelMaker'. A 'Locator' bar is visible. The 'ModelMaker' tab is active, showing a 'Volume' dropdown set to 'EMSeg_6707'. Below this, there is a 'Label' dropdown set to '4' and a 'Name' field containing 'Ventricles'. There are also 'Smooth' and 'Decimate' sliders set to 30 and 5 respectively. A 'Create' button is at the bottom. At the very bottom, performance statistics are displayed: 'Marching cubes: 0.958971 sec.', 'Decimate: 0.046577 sec.', 'Smooth: 0.023842 sec.', and '5028 polygons reduced to 2982'.

1. Click on the ModelMaker tab
2. Select the EMSegment image
3. Each bone is stored under a different label and the model for each label must be generated separately. Select the label that you would like to model.
4. Ignore the default names or rename them if you wish.
5. Smooth = 30 and Decimate = 5 were the optimum values for our purposes. You may need to optimize them for your project.
6. Click Create

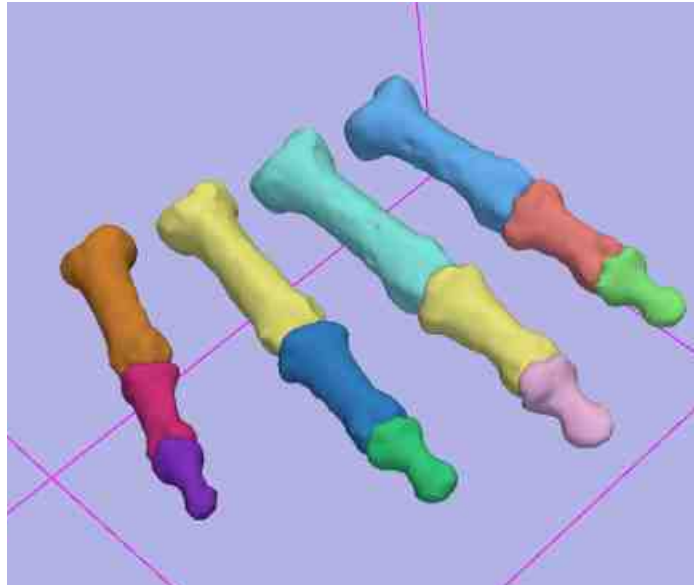
The resulting image is as follows:



Index finger 3-D model using the EMSegment module

Step 8: Apply to Remaining Phalanx Bones

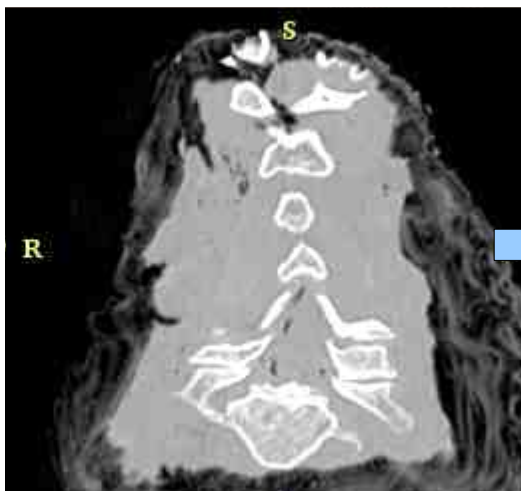
To segment the remaining phalanx bones of the hand, a similar process can be followed as outlined above. Generate a separate super class for each phalanx bone and optimize the parameters for that particular bone. A probability map of each of the bones is necessary to obtain a clean image. These probability maps should be similar in format to those shown in Step 1 of this tutorial. The resulting image of all of the phalanx bones is shown below.



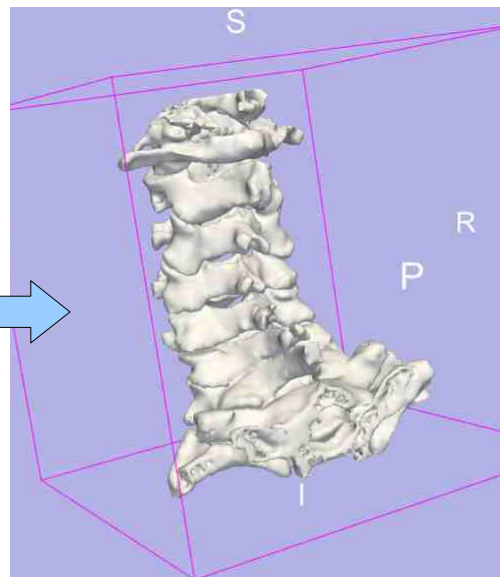
Phalanx bones 3-D model using the EMSegment module

Other Applications

We also used a similar process to segment a section of the human cervical spinal column. In this example we used four classes that represent the four distinct regions of background and tissue present in the image. Optimization of parameters as shown above results in a fairly clean model of the cervical spinal column as shown below.



CT image of the cervical spine



Cervical spinal column 3-D model using the EMSegment module

Appendix

Place new registration code here - old code is listed below....

```
mimxCoRegisterRigid -f subjectImage.hdr \  
-m atlasImage.hdr -s 500000 \  
-o atlasToSubjectRigid.xfm
```

```
mimxCoRegisterBSpline -f subjectImage.hdr \  
-m atlasImage.hdr -n 100 -g 8 8 4 -b 3 3 3 \  
-t atlasToSubjectRigid.xfm \  
-o atlasToSubjectSpline.xfm
```

```
mimxResampleImage -f subjectImage.hdr \  
-m atlasImage.hdr -o atlasToSubject.hdr \  
-t 2 -x atlasToSubjectSpline.xfm
```

It appears that the number of points to be used in the B-Spline registration is resolution dependent and will need to be adjusted for particular applications. The above worked for 0.5mm isotropic resolution, but a larger grid size of 12 12 12 appeared to work for 0.2 mm resolution. The complete command line for the program is as follows:

Usage : mimxCoRegisterRigid

System tags:

```
[ -v ] or [ -h ]  
= List options in short format  
[ -V ] or [ -H ]  
= List options in long format  
[ -vxml ] or [ -hxml ] or [ -exportXML ]  
= List options in xml format for BatchMake  
[ -vgad ] or [ -hgad ] or [ -exportGAD ]  
= List options in Grid Application Description format  
[ -version ]  
= return the version number  
[ -date ]  
= return the cvs checkout date
```

Command tags:

```
[ -m < filename > ]  
= Moving Image  
[ -f < filename > ]  
= Fixed Image  
[ -o < filename > ]  
= Output Transform  
[ -n [ iterations ] ]  
= # of iterations [1000]
```

```

    With: iterations (Default = 1000)
[ -s [ samples ] ]
  = # of samples for MI computation [100000]
    With: samples (Default = 100000)
[ -r [ relaxation-factor ] ]
  = Relaxation Factor [0.5]
    With: relaxation-factor (Default = 0.5)
[ -max [ max ] ]
  = Maximum step size [0.2]
    With: max (Default = 0.2)
[ -min [ min ] ]
  = Minimum step size [0.0001]
    With: min (Default = 0.0001)
[ -ts [ translation-scale ] ]
  = Translation scale [1000]
    With: translation-scale (Default = 1000)
[ -r [ option ] ]
  = Reorient images before registration [0=Off, 1=On]
    With: option (Default = 0)

```

Usage : mimxCoRegisterBSpline

System tags:

```

[ -v ] or [ -h ]
  = List options in short format
[ -V ] or [ -H ]
  = List options in long format
[ -vxml ] or [ -hxml ] or [ -exportXML ]
  = List options in xml format for BatchMake
[ -vgad ] or [ -hgad ] or [ -exportGAD ]
  = List options in Grid Application Description format
[ -version ]
  = return the version number
[ -date ]
  = return the cvs checkout date

```

Command tags:

```

[ -m < filename > ]
  = Moving Image
[ -f < filename > ]
  = Fixed Image
[ -t [ filename ] ]
  = Starting Rigid Transform
[ -o < filename > ]
  = Output Transform
[ -n [ iterations ] ]
  = # of iterations [1000]
    With: iterations (Default = 1000)
[ -g [ xsize ] [ ysize ] [ zsize ] ]
  = Grid Size [12 12 12]
    With: xsize (Default = 12)
    With: ysize (Default = 12)
    With: zsize (Default = 12)
[ -b [ xsize ] [ ysize ] [ zsize ] ]
  = Border Size [3 3 3]
    With: xsize (Default = 3)

```

```

    With: ysize (Default = 3)
    With: zsize (Default = 3)
[ -bx [ type ] ]
    = Bound X dimension deformations [0=unbounded, 1=use lower bound, 2=use
lower and upper bounds, 3=use upper bound
    With: type (Default = 0)
[ -by [ type ] ]
    = Bound Y dimension deformations [0=unbounded, 1=use lower bound, 2=use
lower and upper bounds, 3=use upper bound
    With: type (Default = 0)
[ -bz [ type ] ]
    = Bound Z dimension deformations [0=unbounded, 1=use lower bound, 2=use
lower and upper bounds, 3=use upper bound
    With: type (Default = 0)
[ -xlb [ bound ] ]
    = Lower Bound X dimension deformations
    With: bound (Default = -12)
[ -xub [ bound ] ]
    = Upper Bound X dimension deformations
    With: bound (Default = 12)
[ -ylb [ bound ] ]
    = Lower Bound Y dimension deformations
    With: bound (Default = -12)
[ -yub [ bound ] ]
    = Upper Bound Y dimension deformations
    With: bound (Default = 12)
[ -zlb [ bound ] ]
    = Lower Bound Z dimension deformations
    With: bound (Default = -12)
[ -zub [ bound ] ]
    = Upper Bound Z dimension deformations
    With: bound (Default = 12)
[ -c [ number ] ]
    = Number of Corrections [12]
    With: number (Default = 12)
[ -e [ number ] ]
    = Number of Evaluations [500]
    With: number (Default = 500)
[ -h [ bins ] ]
    = Number of Histogram Bins [50]
    With: bins (Default = 50)
[ -s [ scale ] ]
    = Spatial Sample Scale [100]
    With: scale (Default = 100)
[ -cf [ factor ] ]
    = Convergence Factor [10000000]
    With: factor (Default = 10000000)
[ -gt [ tolerance ] ]
    = Gradient Tolerance [0.0001]
    With: tolerance (Default = 0.0001)
[ -z [ index ] ]
    = Moving Image 4D Index [0]
    With: index (Default = 0)
[ -r [ option ] ]
    = Reorient anatomical image [0=Off, 1=On]

```

With: option (Default = 0)

Usage : mimxResampleImage

System tags:

- [-v] or [-h]
= List options in short format
- [-V] or [-H]
= List options in long format
- [-vxml] or [-hxml] or [-exportXML]
= List options in xml format for BatchMake
- [-vgad] or [-hgad] or [-exportGAD]
= List options in Grid Application Description format
- [-version]
= return the version number
- [-date]
= return the cvs checkout date

Command tags:

- [-m < filename >]
= Moving Image Filename
- [-f < filename >]
= Fixed Image Filename
- [-o < filename >]
= Output Image Filename
- [-x < filename >]
= Input Transform Filename
- [-t [option]]
= Transform type: 1=Rigid, 2=B-Spline
With: option (Default = 1)
- [-r [option]]
= Reorient anatomical image [0=Off, 1=On]
With: option (Default = 0)