Intracranial Segmentation

B. Douglas Ward
Biophysics Research Institute
Medical College of Wisconsin
email: ward@post.its.mcw.edu

June 4, 1999

1 Program 3dIntracranial

1.1 Purpose

Program 3dIntracranial provides automatic segmentation of the intracranial region. Since this program does not require operator interaction, it can be used for batch processing of a number of images. The output segmented images can be in one of two formats: (1) a segmented anatomical dataset, where brain voxels retain their original gray-scale intensities, and non-brain voxels are set to zero, or (2) a functional dataset "mask", consisting of zero's (corresponding to brain voxels) and one's (corresponding to non-brain voxels).

The anatomical dataset output can be used directly as the underlay for display of functional activation maps. Also, the segmented anatomical dataset can be used as input to the volume rendering program which is included with the *AFNI* package.

The functional dataset mask output can be used as an overlay on top of the original anatomical dataset. This allows for convenient visual inspection of the automatic segmentation results, and manual editing of the output if necessary (see Draw Dataset plugin). Assuming that the original anatomical dataset will be retained, conversion of the functional mask into a segmented anatomical dataset is trivial. Since the functional mask dataset can be compressed to a tiny fraction of the disk space required by the anatomical dataset, this might be another advantage of the functional dataset output over the anatomical dataset output.

1.2 Theory

1.2.1 Probability Density Estimation

One way to distinguish brain from non-brain voxels is by their gray-scale intensities. A plot of the histogram of gray-scale intensities for a typical anatomical image shows, in addition to the enormous peak near zero corresponding to voxels outside the brain, two small peaks at higher intensities, corresponding to the gray matter and white matter voxels. The distribution of gray matter and white matter voxel intensities will be modeled using the normal PDF:

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

Also, for greater modeling accuracy, a third normal distribution, corresponding to "background" voxels, will be used.

Therefore, *in the neighborhood of the gray matter and the white matter peaks*, the distribution of voxel intensities will be modeled as the weighted sum of three normal distributions:

$$f(x) = k_b * n(x;\mu_b,\sigma_b^2) + k_g * n(x;\mu_g,\sigma_g^2) + k_w * n(x;\mu_w,\sigma_w^2)$$

where
k_b = coefficient for background distribution
μ_b = mean for background distribution
σ_b = std. dev. for background distribution
k_g = coefficient for gray matter distribution
μ_g = mean for gray matter distribution
σ_g = std. dev. for gray matter distribution
k_w = coefficient for white matter distribution
μ_w = mean for white matter distribution
σ_w = std. dev. for white matter distribution

The above set of 9 parameters is estimated using a nonlinear optimization method (the Simplex algorithm), which seeks a least squares fit of the above PDF to the empirical PDF obtained from the given anatomical dataset. (Technical note: since the background distribution is usually truncated, it is seldom true that $k_b + k_g + k_w = 1$. Hence, all 9 parameters must be estimated.) The large variation in the values of these parameters from one image to another makes it necessary to estimate these parameters separately for each image.

Now, we want to set the lower and upper bounds so that most of the voxels containing gray matter or white matter are included, and other voxels are excluded. The bounds calculated by program 3dIntracranial are as follows:

lower bound: $\mu_g - 2\sigma_g$
upper bound: $\mu_w + 2\sigma_w$

(Note: these bounds can be manually overridden by the user.) Assuming that the above physical model is correct, and that the PDF parameter estimates are accurate, the intensity bounds should contain approximately 98% of the gray matter and the white matter voxels.

1.2.2 Slice-by-slice Segmentation

Starting with the middle axial slice, voxels are preliminarily classified as belonging to the brain if their intensities are within the above bounds. The result is the set of (in-brain) voxels R. Note that R is usually a disconnected set. Since we would like the brain to be connected (and ignoring, for now, the fact that what appears disconnected for a slice may actually be connected in 3 dimensions), we form the set S which is the largest subset of R, starting from the center, which is connected. Now, although S is connected, it may contain holes. To eliminate the holes, we form the set T which is the largest connected set, starting from outside the brain, which excludes S. Therefore, the complement of T, $U \equiv T^C$, contains S. Moreover, U is connected, and does not contain holes. We will take U as the estimate of the intracranial region for the current slice.

Note that we start with the middle axial slice (more precisely, we start with the axial slice which contains the center of mass for the image). This is done for two reasons: First, the middle axial slice usually allows a clean separation between brain and non-brain voxels based upon image intensity, as described in the previous paragraph. For superior, and especially for inferior, slices, the picture becomes more complicated. Eventually, everything is connected to everything else. Allowing an unrestricted growth in the set U from slice to slice would result in "flooding" of non-brain areas. Second, the middle axial slice contains close to the maximum number of brain voxels per slice. In general, the set of brain voxels shrinks with each slice as we get farther away from the middle slice.

Therefore, to restrict the growth in the set U, and since we expect the set of brain voxels to shrink as we move away from the middle slice, the set U for the current slice is used as a mask for the next slice which is farther away from the middle. (Actually, we use the set U', which is a slightly enlarged version of U, to allow for possible growth.) In this way, we restrict the growth of U from slice to slice, and prevent flooding of non-brain areas.

Now, one problem, alluded to above, is that we have, by design, required U for each slice to be a connected set. However, particularly for inferior slices, an axial slice of the brain is not necessarily a connected set. Therefore, to compensate for this limitation, we repeat the entire above procedure using sagittal slices, followed by coronal slices. This allows the set of "brain" voxels to grow into regions which were previously excluded due to the connectivity requirement.

1.2.3 3D Envelope

In order to help smooth the brain image, an envelope is constructed in three dimensions, and all voxels inside this envelope are classified as brain voxels. The envelope is constructed as follows: A two-dimensional array $R(\theta, \phi)$ is calculated for discrete values of "latitude" θ and "longitude" ϕ. $R(\theta, \phi)$ is the maximum distance from the center of the brain to any voxel inside the brain at angle (θ, ϕ) from the center. The array $R(\theta, \phi)$ is smoothed, by local averaging, producing an array $R^S(\theta, \phi)$.

The distance r and direction (θ, ϕ) of each voxel from the center of the brain is calculated. If

$$r < R^S(\theta, \phi)$$

then the voxel in question is reclassified as being part of the brain, regardless of the previous classification.

1.2.4 Connectivity

Another factor that can be exploited for automatic image segmentation is the fact that anatomical structures tend to be strongly connected. That is, weakly connected brain voxels probably do not correspond to actual anatomical structures. It is also unlikely that weakly connected non-brain voxels would correspond to actual anatomy.

Therefore, after the above slice-by-slice segmentation, and envelope construction, we have a set of voxels which is tentatively labeled as belonging to the brain structure. Each voxel in this set is examined to determine if it is weakly connected to the rest of the brain structure. Weakly connected brain voxels are removed from the brain structure, and weakly connected holes in the brain structure are filled in.

Each voxel has 6 nearest neighbors in the 3 dimensional image. Thus, for any voxel, the number of neighboring voxels which have been labeled as belonging to the brain structure can vary from 0 to 6. The operator has the option of specifying the *Maximum Connectivity to Leave*, i.e., the number n is input such that any voxel in the brain structure which has $\leq n$ neighboring voxels (out of 6) in the brain structure will be removed from the brain structure.

Similarly, the operator has the option of specifying the *Minimum Connectivity to Enter*, i.e., the number m is input such that any voxel not in the brain structure which has $\geq m$ voxels (out of 6) in the brain structure will be added to the brain structure.

1.3 Usage

The syntax for execution of program 3dIntracranial is as follows:

3dIntracranial **-anat** *filename* [**-min_val** *a*] [**-max_val** *b*]
[**-max_conn** *n*] [**-min_conn** *m*] [**-mask**] [**-quiet**]
-prefix *pname*

The different command line options are explained below.

1.4 Options

-anat *fname*

The mandatory -anat command specifies that $fname$ is the filename of the *AFNI* 3d anatomical dataset to be segmented.

-min_val *a*

The optional -min_val command specifies that a is the lower bound gray-scale intensity for a voxel to be initially classified as belonging to the brain. Voxels having gray-scale intensity less than a will be ignored during the preliminary classification phase of the segmentation algorithm. (However, it is quite possible for such voxels to be added to the brain structure at a later stage in the process.)

Default: The program will calculate the lower bound a using the internal PDF estimate obtained for that particular dataset.

-max_val *b*

The optional -max_val command specifies that b is the upper bound gray-scale intensity for a voxel to be initially classified as belonging to the brain. Voxels having gray-scale intensity greater than b will be ignored during the preliminary classification phase of the segmentation algorithm. (However, it is quite possible for such voxels to be added to the brain structure at a later stage in the process.)

Default: The program will calculate the upper bound b using the internal PDF estimate obtained for that particular dataset.

-max_conn *n*

The optional -max_conn command is used to remove isolated voxels from the brain structure. For each voxel that has been classified as belonging to the brain structure, the number of its six nearest neighbors which also belong to the brain structure is determined. If this number is $\leq n$, then this voxel is removed from the brain structure. Only voxels with $> n$ neighbors in the brain structure will remain in the brain structure. The acceptable values for the integer n are: $-1 \leq n \leq 5$. Setting $n = -1$ effectively disables this connectivity test, since no voxels will be removed from the brain structure.

Note that this "maximum connectivity to leave" test is always performed *before* the "minimum connectivity to enter" test, regardless of the order in which the commands are entered.

Default: $n = 2$.

-min_conn *m*

The optional -min_conn command is used to remove isolated holes from the brain structure. For each voxel that has been classified as *not* belonging to the brain structure, the number of its six nearest neighbors which belong to the brain structure is determined. If this number is $\geq m$, then this voxel is added to the brain structure. The acceptable values for the integer m are: $1 \leq m \leq 7$. Setting $m = 7$ effectively disables this connectivity test, since no voxels will be added to the brain structure.

Note that this "minimum connectivity to enter" test is always performed *after* the "maximum connectivity to leave" test, regardless of the order in which the commands are entered.

Default: $m = 4$.

-mask

The optional -mask command specifies that the output will be a functional (*AFNI* 'fim') dataset, consisting of zero's (corresponding to brain voxels) and one's (corresponding to non-brain voxels).

Default: The output will be a segmented anatomical dataset, where brain voxels retain their original gray-scale intensities, and non-brain voxels are set to zero.

-quiet
The optional -quiet command is used to suppress output to the screen.

-prefix *pname*
The mandatory -prefix command specifies that *pname* is the prefix of the filename of the output *AFNI* 3d dataset to contain the segmented volume. Note that, since this program will not overwrite a previously existing file, the specified output file must *not* exist prior to program execution.

1.5 Example

Suppose that a number of anatomical images require intracranial segmentation. For convenience, the names of the subjects of the anatomical images are listed in a separate ASCII file Subjects.lst, e.g.:

Contents of Subjects.lst:

```
adam
baker
charley
⋮
etc.
```

The files adam+orig (.HEAD and .BRIK), baker+orig (.HEAD and .BRIK), etc., are stored in the current directory. In order to segment the intracranial region for each of these datasets, the following script file might be used:

Script for program 3dIntracranial:

```
foreach   name (`cat Subjects.lst`)
          echo Processing: $name

          3dIntracranial \
          -anat $name+orig \
          -prefix $name.brain

end
```

This script file extracts each of the subject names from file Subjects.lst, one by one, and builds the name of the corresponding anatomical dataset. The program 3dIntracranial performs automatic segmentation of the intracranial region, using the algorithm outlined above. Finally, the segmented anatomical images are saved into the datasets:
adam.brain+orig, baker.brain+orig, etc.

An alternative is to produce the functional image masks. This is done using the optional -mask command:

Script for program 3dIntracranial:

```
foreach   name (‘cat Subjects.lst‘)
          echo Processing: $name

          3dIntracranial \
          -anat $name+orig \
          -mask \
          -prefix $name.mask

end
```

The functional image mask files are saved into the *AFNI* ‘fim’ datasets: adam.mask+orig, baker.mask+orig, etc. These mask files may be used to overlay the original anatomical images. To convert a mask file to the corresponding segmented anatomical image, the following commands may be used:

Converting functional mask to anatomical image

```
3dcalc \
-a adam+orig \
-b adam.mask+orig \
-expr “a*(1-b)” \
-prefix adam.brain
```

Note that the mask file contains zeros at brain locations, and ones at non-brain locations. Therefore, the above expression multiplies the original anatomical image not by $mask$, but by $1 - mask$.

1.6 Notes

- At present, the input anatomical dataset, as well as the output segmented image dataset (whether anatomical or functional), must be stored in the short integer format.
- It is assumed that all voxel intensities are nonnegative.
- In general, the segmentation algorithm seems to work better with .orig files than with .tlrc files. Therefore, it might be better to first segment the original data, and then to transform the segmented anatomical image into Talairach coordinates.