

CUBE ANALYZER
A Medical Image Analysis Program
User's Manual

*John Hoford, Howie Choset,
Stane Kovacic, Jim Gee
and Ruzena Bajcsy*

MS-CIS-88-101
GRASP LAB 170

Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104

December 1988

Acknowledgements: This research was supported in part by Air Force AFOSR F49620-85-K-0018, U.S. Army grants DAA29-84-K-0061, DAA29-84-9-0027, NOO14-85-K-0807, NSF grants MCS-8219196-CER, IRI84-10413-AO2, INT85-14199, DMC85-17315, NIH NS-10939-11 as part of the Cerebro Vascular Research Center, NIH 1-RO1-NS-23636-01, NATO grant 0224/85, NASA NAG5-1045, ONR SB-35923-0, DARPA grant NOO014-85-K-0018, and by DEC Corporation, IBM Corporation and LORD Corporation.

CUBE ANALYZER

A Medical Image Analysis Program

User's Manual

*John Hoford, Howie Choset, Stane Kovacic,
Jim Gee, and Ruzena Bajcsy*

General Robotics and Active Sensory Perception Laboratory
UNIVERSITY OF PENNSYLVANIA
Department of Computer and Information Sciences
Philadelphia, Pennsylvania 19104-6389

Abstract

The Cube Analyzer is a group of programs designed to analyze three-dimensional tomographic data such as CT, PET and MRI scans. It is designed to work on a Unix workstation running X-windows, and was programmed in C. Cube Analyzer provides the user with an interactive system by which he can view the scans, define regions of interest (ROI), and extract data. The system can be used alone to analyze images, or combined with the elastic matching software package to analyze computer-generated ROI's.

Contents

1	Introduction	1
1.1	An Overview of The System	1
1.1.1	The Main Program	1
1.1.2	The Supporting Programs	2
2	The Main Program in Detail	4
2.1	The Windows	4
2.1.1	The Information Window	5
2.1.2	The Contrast Enhancement Window	6
2.1.3	The Original Image Windows	6
2.1.4	The Zoomed Image Window	6
2.1.5	The One-Dimensional Statistics Display Window . . .	6
2.1.6	The Two-Dimensional Statistics Display Window . . .	7
2.1.7	The Three-Dimensional Display Window	7
2.2	Commands	7
2.2.1	The Information Window	8
2.2.2	The Contrast Enhancement Window	8
2.2.3	The Original Image Windows	8
2.2.4	The One-Dimensional Statistics Display Window . . .	9
2.2.5	The Zoomed Picture Window	9
2.2.6	The Two-Dimensional Statistics Display Window . . .	10
2.2.7	The Three-Dimensional Display Window	10
3	The Supporting Programs In Detail	11
3.1	The 3-D statistics program	11
3.2	The Atlas Generation Program	12

4	Appendix	13
4.1	Sample Run	13
4.2	Regional Statistics	17
4.2.1	The Match3 Program	17
4.2.2	The Superstat Program	22
4.2.3	The Stat2d2 Program	24

Chapter 1

Introduction

The Cube Analyzer is a group of programs designed to analyze three-dimensional tomographic data such as CT, PET and MRI scans. It is designed to work on a Unix workstation running X-windows, and was programmed in C. Cube Analyzer provides the user with an interactive system by which he can view the scans, define regions of interest (ROI), and extract data. The system can be used alone to analyze images, or combined with the elastic matching software package to analyze computer-generated ROI's.

1.1 An Overview of The System

The system consists of a main program and several support programs. The main program displays the three-dimensional data sets and is used to draw and edit ROI's. One of the main reasons for the existence of Cube Analyzer is to display results from the elastic matching software package, which automatically generates ROI's. A support program generates comparative three-dimensional statistics from ROI's, either those drawn with the main program or obtained from the elastic matcher.

1.1.1 The Main Program

In many ways this program is like a word processor program, but it is used to generate and edit regions and statistics, rather than text. Like most users of word processor programs, it is not expected that the users of this program will learn every command, but will use those that are needed and disregard the others until they are required.

One must also bear in mind that this is an on-going project and, like a word processor program, is never really complete as new features can be continuously added.

The main program (d3d4) has the following features:

- Displays several voxel (cubic) represented data sets.

- Shows any slice from the data set (the cube).

- Zooms on the displayed slice, magnifying it if necessary.

- Draws regions of interest (ROI) on zoomed slices and maintains a data structure that can link the two dimensional ROIs to form a three dimensional ROI.

- Saves and recalls the ROI to files and edits the ROI.

- Generates information about a point, line, 2-D region, (3-D done by a separate batch type program).

- Adjusts the contrast and the color mapping of the pictures.

- Displays defined three-dimensional regions in perspective.

1.1.2 The Supporting Programs

The supporting programs are used to modify or analyze the ROI files created by the main program.

The Three-Dimensional Statistics Program

The statistics program generates comparative three-dimensional statistics on regions. Given a set of ROI studies, the program generates statistics on each of the three- dimensional ROI's defined in each study. The program also generates statistics for the union and intersection of the same ROI across all given studies. It also generates another image file which contains the original data cube plus the ROI's being compared.

The Atlas Generation Program

This program is used in the generation of atlases from ROI data. The atlas generated is a voxel representation with the same data structure used to store all voxel data sets. Atlases can be colored to suit the particular modality that one wishes to simulate. For more details see the Elastic Matcher manual.

The Elastic Matcher

The Elastic Matcher was the main reason for creating the Cube Analyzer program. There is a separate manual for that package, so most of the descriptions of elastic matching routines are included in that manual. Editing of ROI's generated by the elastic matcher, particularly the tasks done with the Cube Analyzer, is discussed in both manuals.

Chapter 2

The Main Program in Detail

2.1 The Windows

The Cube Analyzer is made up of several different windows. Different windows have particular functions and look distinct. The system uses multiple windows in order to allow the user to position them in a manner that suits him best. It also allows the user to hide or put aside information that he is presently not interested in. For example, if he does not wish to see a patient's data window he can Iconify. Multiple windows also allow the program to have the same commands represent different things in different windows.

The windows are standard X windows and, as such, the window will accept all the commands that the user's window manager provides, such as Move and Iconify. This helps the user, as he does not have to learn a new set of commands. The basics of using X windows are introduced only when needed. For a more complete tutorial of X, the user should consult the X reference manuals.

A list of the windows follows:

The Information Window.

The Contrast Enhancement Window.

The Original Image Windows (1 or 2).

The Zoomed Image Window.

The One-Dimensional Statistics Display Window.

The Two-Dimensional Statistics Display Window.

The Three-Dimensional Display Window.

2.1.1 The Information Window

The information window provides the user with textual information concerning the images being displayed. Most of this is provided in the header information of the cubic data sets.

The information includes:

The patient's name.

The patient's run number.

The study type.

The date of the study.

The number of slices in the cubic data set.

The dimensions of the image.

The slice thickness in millimeters.

The pixel size in millimeters.

The slice presently being displayed.

Various warning or notifying messages.

This window is also used to send prompts, warnings, and messages to the window. A few examples of which would include:

calculating new zoom to explain the delay in bringing up a new zoomed slice.

finished new zoom to notify that it is now done.

You must begin a region first to notify the user what he tried could not be done.

2.1.2 The Contrast Enhancement Window

The full scale of image intensities are displayed in a 0-255 gray scale. For many medical applications, it is of interest to examine only a small range of intensities and to display the image with greater contrast over this range. This is done by setting all intensities below a selected value to 0, all intensities above another selected value to 255, and displaying this limited intensity range in the full 0-255 gray scale. Clinical investigators may also wish to set an intensity window, and then adjust its location while maintaining a constant contrast window. These manipulations are performed in the contrast enhancement window. This window displays a color bar showing the colors and their associated values. It also shows what range of numbers the original number range was mapped to (shows the contrast window) and allows adjustment of the contrast settings.

2.1.3 The Original Image Windows

The original image window displays the picture in its original size and allows one to quickly step through the slices. It also displays any regions drawn on that particular slice as outlines. Currently, there can be up to two image windows displayed concurrently.

2.1.4 The Zoomed Image Window

The zoomed window displays an image which was copied from one of the original image windows. If the original is small, it is magnified by two or four. This is the window that is used to draw two dimensional regions which can be linked to define a 3-dimensional region. Due to the size of a magnified image, only a portion of the total image may be displayed in this window; the rest may be displayed by scrolling the image. The window size can be adjusted by using the normal X window manager commands and the proportion shown will adjust accordingly.

2.1.5 The One-Dimensional Statistics Display Window

The one-dimensional statistics window is used to display cross sectional profile information from lines drawn through any two arbitrarily selected points in the original image window. This is referred to as a temporary window because it is not displayed at all times. It is not shown until the user generates a cross section in the original image window. Currently the display

only shows the intensity cross section along the line drawn, along with the coordinates of the two points which defined the line.

2.1.6 The Two-Dimensional Statistics Display Window

The two-dimensional statistics window displays data extracted by ROI's. When an ROI is selected, this window appears and contains the following:

- The number of pixels in the ROI.

- The centroid of the region (x and y coordinates).

- The mean intensity.

- The standard deviation of the intensity.

- The median intensity.

- A histogram of the intensity of the pixels in the region.

2.1.7 The Three-Dimensional Display Window

This window displays three-dimensional outlines in perspective. A two-dimensional region that belongs to the 3-dimensional object wanted, is selected. All outlines with the same region number are then displayed. The objects can be rotated about the Z axis and the current X axis.

2.2 Commands

Commands are entered with a keyboard and mouse. Each window accepts different commands, and the user sends commands to that window if, and only if, the mouse's cursor is in that window. For example, in one window the + and - keys change the slice number, while in another window they change the region being selected. At first one would think it might be confusing to have a key mean different things in different windows, but since the functions are relative to that window the use of the same keys helps reduce the number of keys you have to remember.

There is one command that works in all the windows – the [stop] key, which ends the execution of the program.

The following are all the commands that can be issued to the system, organized by window.

2.2.1 The Information Window

In the information window, clicking either mouse button at any point along the line defining the current slice changes the current slice to one approximately proportional to the position of the mouse cursor. This is used to randomly access individual slices.

2.2.2 The Contrast Enhancement Window

The commands in the contrast enhancement window are:

left mouse button sets the lower value in the contrast range.

right mouse button sets the higher value in the contrast range.

‘+’ key slides the contrast window up.

‘-’ key slides the contrast window down.

f1, f2, f3, and f4 keys change the color map to get various types of color scales.

2.2.3 The Original Image Windows

The commands in the original image window are:

‘+’ key steps one slice forward in the sequence of slices.

‘-’ key steps one slice backward in the sequence of slices.

‘c’ key is used to change slices. The user is prompted for a slice number and that slice will then be displayed.

‘z’ key causes a zoomed image of the slice being displayed to be moved to the zoom window.

mouse - Upon clicking either mouse button at any point, the X and Y coordinates of that point and the intensity of the pixel underneath that point are displayed in the information window.

‘[’ key specifies the starting point for the line upon which the 1-D statistics will be based.

‘]’ key marks the end of the line upon which the 1-D statistics will be based. After pressing this key, the 1-D statistics window appears, displaying the intensity cross section of the line drawn.

2.2.4 The One-Dimensional Statistics Display Window

Pressing either mouse button in this window hides it from display.

2.2.5 The Zoomed Picture Window

The commands in the zoomed picture window are:

cursor keys scroll the underlying picture to enable the user to see the entire picture. The window appears as a hole through which a portion of a much larger object can be seen. When one of the cursor keys is pressed, the “hole” moves in that direction, allowing another part of the image to be seen.

‘b’ key marks the beginning of a 2-dimensional polygon used to define an ROI.

mouse buttons mark points along the polygon.

‘n’ key closes the polygon. The user is then prompted to input the region and frame index and the region and frame parent. This is used to describe the 3-Dimensional data structure.

‘s’ key is used to save the regions that have been outlined. The user is then prompted for the name of the file where these regions will be saved.

‘r’ key is used to recall regions previously saved. The user is prompted for the name of the file to recall. The user can recall several files and they will all be appended to one another automatically. The user will be prompted for the color the regions are to be drawn. Although a green option is available, only use it when necessary since the program uses green to indicate ROI selection.

‘k’ key is used to delete a series of regions by color.

[prev] and [next] keys are used to select a particular ROI. Upon pressing these keys, one of the ROIs is changed from its usual color to a green color. Further pressing of the keys steps one through all ROI’s on the screen, changing each one individually to green as one steps through them.

[select] key is used to generate the 2-dimensional statistics from the highlighted region. This brings up the 2- Dimensional Statistics Window.

[del line] key deletes the highlighted region from the lists of regions. If you hit del line by accident and immediately hit it again, the region will be restored.

‘3’ key displays the 3-Dimensional Display Window, showing the object of which the selected region is a part.

2.2.6 The Two-Dimensional Statistics Display Window

The commands in the two-dimensional statistics display window are:

‘+’ key increases the size of the scale of the histogram.

‘-’ key decreases the size of the scale of the histogram.

‘a’ key automatically adjusts the contrast settings to the mean +/- two standard deviations.

2.2.7 The Three-Dimensional Display Window

The commands in the three-dimensional display win window are:

mouse buttons adjust the viewing angle of the three dimensional image.

‘k’ key removes this window from being displayed.

Chapter 3

The Supporting Programs In Detail

3.1 The 3-D statistics program

The program is executed by the command:

```
match3 <control_file> <outfile>
```

The control file must be structured in the format illustrated in Table 3.1.

The ROI files are in the format that is saved by the main program. The 3-D region numbers are numbers that have been previously agreed upon to represent the various 3-D regions within the scan that is being studied. For each 3-D region in each ROI file, the output file contains:

- The number of pixels in the region.
- The centroid of the 3-D region.
- The mean intensity.
- The standard deviation in the intensity.
- The median intensity.

If, instead of running “match3”, you run “match4”, the computer will produce a file that includes the original images with all of the defined outlines, which may then be displayed with the main program. For details see the Appendix.

image	image location and filename
n	the number of ROI files in the study
file 1	the first ROI file in the series
.	
.	
.	
file n	the n'th ROI file ($n < 7$)
k	the number of different 3-D regions in study
r_1	the first 3-D region number
.	
.	
.	
r_k	the k'th 3-D region number

Table 3.1: Control file for the 3-D statistics program.

n	number of regions
R_1C_1	region number , color of region
.	
.	
.	
R_nC_n	region number , color of region

Table 3.2: Control file for the atlas generation program.

3.2 The Atlas Generation Program

The user generates a set of three dimensional regions in the form of the atlas that he wishes to create. The program `gen_atlas` is then run with the command “`gen_atlas <ROI file> n`” where `n` is the number of slices. The program reads the information on how to color the atlas and what regions to put into the atlas from a file named “`rlist`” in the current directory. The format of the “`rlist`” file is illustrated in Table 3.2.

For more details see the Elastic Matcher manual.

Chapter 4

Appendix

4.1 Sample Run

This is a sample run on the GEORGES machine only, with a description of how to do all functions of the cube analyzer. All commands that the user must enter are surrounded by asterisks, while all computer responses are surrounded by + signs. <CR> indicates carriage return.

The image is brought into memory and the middle slice is displayed by typing the command “d3d4 filename.”

```
***d3d4 /pic/pics/ct/lincoln/lincoln.ct<CR>***
```

The computer will take several seconds to create the windows and bring this file into memory. It will then display slice 14 in the lower left (Original Image) and lower right (Zoomed Image) windows, and will provide information about the subject in the upper left window (Information), and the window from which d3d4 was invoked will tell you that the image is being displayed by printing:

```
+++getting /pic/pics/ct/lincoln/lincoln.ct+++  
+++in mapcolors+++  
+++after dispic+++  
+++no. of rns 0+++  
+++after makezoom+++  
+++C+++
```

In the Information window, there will be the phrase +++calculating new zoom+++ highlighted in red. This will change shortly to +++finished new zoom+++ when the image in the Zoom window is created. The file is then ready for display manipulations and analysis.

The Contrast Enhancement window initially overlaps part of the original image. To move it out of the way, place the cursor in the window by rolling the mouse appropriately, press the “Extend char” button down and press the left mouse button. While holding both buttons down, roll the mouse slowly upwards and to the right. The contrast window will also move in this direction, uncovering the remainder of the original image. Release all buttons and the window will stay in its new position.

To change the contrast, again position the cursor within the window. Position the cursor along the right color bar near the number 64, and then press the left mouse button. The image on the left should become dark. Then move the cursor near the number 128 and press the right mouse button. The original image should appear within increased contrast, and the numbers along the left color bar should have changed from 0, 64, 128, 192, and 256 to something like 63, 79, 95, 111, and 127, indicating the contrast range that you have selected. Type the letter “z”, with the cursor in any position at all, and this enhanced image will shortly appear in the Zoom window.

If you would like to remove the contrast window entirely, place the cursor in the window, hold down the “Extend char” button, and press both mouse buttons simultaneously. The window will disappear, and will be replaced by a small box with the word “color” inside. Repeating the process on this window will restore the original contrast window, if desired.

Different color scales may be selected by putting the cursor in the Contrast window and typing the f2, f3, or f4 keys. f3 results in a reverse gray scale, while the other two give different color scales, neither of which is currently well suited to any particular modality image. Typing f1 returns to the original gray scale.

Different slices may be selected in two ways, using the “+” and “-” keys while the mouse is within the original image window, or using the slice bar in the information window.

Position the cursor within the original image window and then press the “+” key. The image should change quickly, maintaining the contrast settings previously set. The information window will now inform you that slice 15 is being displayed, while the 14 in parentheses indicates that slice 14 is currently in the Zoom window. You may step through the slices using the “+” and “-” keys in this manner.

Slices may also be selected randomly by positioning the cursor at the appropriate position on the slice selection bar in the Information window and then clicking either mouse button. Slice number 1 may be selected by placing the cursor to the left of the scale, and the highest slice by placing it to the right of the scale. With experience, you will probably be able to get within one slice of the slice you want using the bar, and then make final corrections by moving the cursor into the original window and using “+” and “-” as described above.

Another random selection method is to press the “c” key (change slices) in the original image window. A window pops up and prompts for the slice number of the slice to be displayed.

In all cases, the zoomed image does not change until and unless you type “z”.

Zero-dimensional analysis is done in the original image window. With slice 14 displayed, place the cursor at any point in the image and press either mouse button. In the information window, just above the image, the coordinates of this point and the gray level of the image at this point will be displayed with a statement like `+++pressed at x 122 y 105 col=123+++`. Try this at several points in the image. You should get values of 60-80 inside the ventricles, 255 in bone, 100-120 elsewhere in the brain. The coordinates, as you will see, are with respect to 0,0 being in the upper left corner and 512,512 the lower right.

One-dimensional analysis is also done in the original image window. Position the cursor somewhere in the skull on the left side of the image at the level of the ventricles and press the “[” key, then move the cursor to the skull on the right side and press the “]” key. A line will appear between the two points, and the 1-D statistics window will appear. This window will contain the cross-sectional profile defined by that line, with values as you saw by the point by point analysis you just did above. The left and right coordinates also appear in this window.

To remove the 1-D statistics window, place the cursor within the window and press either mouse button.

Continue your analysis of slice 14 by drawing 2-dimensional contours of the right head of the caudate nucleus. (Remember that all images are in CT format, so patient left is image right). The zoomed image window is used for 2-D and 3-D analysis. Place the cursor at some point on the contour of the structure and type “b”. Define the contour by repositioning the cursor and pressing either mouse button at various points. When ready to close the region, type “n”, and the computer will automatically connect the last

point you defined with the beginning point.

A window will then pop up and display the message, `+++input region index+++`. Type the number `"132"<CR>`, which is the predefined code number for the right head of the caudate. The computer will then say `+++input frame index+++`, to which you respond with `"1"<CR>`, because this is the first region drawn in this frame. The computer will then say `+++input region parent index+++`, to which you respond `"0"<CR>`, since no other region surrounds this one, and the same response is given to `+++input frame parent index+++` for the same reason. The window then disappears and you may proceed to draw additional regions, analyze regions, etc.

To extract statistics from the region just drawn, move the cursor to the zoom window and press the "select" key. The 2-D statistics window appears, containing the following information: number of pixels, mean, standard deviation, and median gray values within the region, and the coordinates of the centroid (center of gravity). A histogram of the gray values is also displayed.

To remove this window, place the cursor within it and press either mouse button.

Now go to slice 13 and type `"z"`. You have not yet drawn any regions on this slice, so none are present, but the one you just drew on slice 14 is still in memory. Now outline the same structure in this slice, assign the same input values, look at the statistics, and remove the window.

By defining the same object in more than one slice, you have created a 3-D object. You may display this object by first pressing the "next" key to highlight the ROI in green. Then press the `"3"` key and a larger 3-D display window will appear. The 2 contours will be drawn along with a set of axes, and the statement `+++Displaying region 132 (found in 2 slices)+++`.

Iconify this window, and then save the set of regions by placing the cursor in the zoom window and typing `"s"`. A window will pop up and display the message, `+++Input file name to save+++`. Give this file a logical name, like `rhcn.roi` (better yet, instead of the `roi` extension, use your initials).

Now press the "stop" key to end execution of the program.

You may now restart the program by going to the Text Input window and typing `"!d"<CR>`, which will tell the computer to do the same thing as before when you typed `***d3d4 /pic/pics/ct/lincoln/lincoln.ct***`. This time, when the images appear you may recall the previously saved set of ROI's by placing the cursor in the Zoom window and typing `"r"`. The computer will ask you to `+++input file name to recall+++`, which in this case will be `"rhcn.roi"`. Look at slices 13 and 14 to verify that your original

ROI's were indeed saved and recalled.

You may extract statistics from this recalled ROI set by pressing the “prev” or “next” keys until the desired ROI is highlighted. Then pressing either “select” or “3” will again produce the 2-D or 3-D statistics for that region.

If you choose to add new regions now, when you save the set it will contain both the old and new ROI's. It is also possible to merge several set of regions by recalling them one after the other. Try this by recalling the following files:

```
/pic/pics/ct/lincoln/experts/lincoln.wa  
/pic/pics/ct/lincoln/experts/lincoln.rwd  
/pic/pics/ct/lincoln/computer/lincolnxxx.r40
```

This will result in the display of two set of regions defined by different expert readers, one set defined by the elastic matcher, and whatever regions you have drawn yourself. With all these regions in place, try moving from slice to slice and region to region. Display both the 2-D and 3-D statistics for several regions until you understand exactly what is required to select the ROI you want. You may occasionally have to hold down the “prev” or “next” keys for several seconds until any region is highlighted in green. Then step back in the opposite direction to reach the desired ROI.

The next section will show you how to extract 3-D statistical data from your ROI data sets.

4.2 Regional Statistics

4.2.1 The Match3 Program

It is possible to obtain statistics on 3-D regions from any number of readers and for any number of structures. When these data are generated, the computer automatically compares the degree of overlap among all of the readers, so you may need to run the statistics program several times, depending on exactly what data are required. 3-D statistics are generated using the programs “match3” or “match4”. Each of these requires the presence of another file, usually defined as “filename.stat”. The data should be written into a file, usually “filename.dat”, which may then be displayed or printed. “Match4” asks for an additional outline file, which it will create, and which consists of the original image file with regional contours added.

These programs require an input “*.stat” file in the following format:

```
/pic/pics/ct/kearney/kearney.ct  
2  
/pic/pics/ct/kearney/experts/kearney.wa  
/pic/pics/ct/kearney/experts/kearney.rwd  
4  
111  
112  
131  
132
```

To run the statistics program type:

```
match3 kearney.stat kearney.dat
```

This will write to the file “kearney.dat”, the statistics shown in the following pages.

The first line defines the location and filename of the CT scan from which data were extracted. The “2” on the second line defines the number of data sets to be compared. Lines three and four define the individual ROI files to be compared. The “4” on line five indicates that four structures are to be analyzed. Subsequent lines list the structures by their anatomical code.

The following is a sample output file from a comparison of two readers and four structures:

REGION NUMBER 111

OUT OF INTERSECTION OF ALL DATA:

centroid at (119.9,110.5,80.0) (mm)
volume (mm³) : 523.230
mean intensity : 105.100
median intensity : 106
standard deviation: 8.849

IN INTERSECTION OF ALL DATA:

centroid at (119.1,104.0,80.0) (mm)
volume (mm³) : 569.496
mean intensity : 111.106
median intensity : 111
standard deviation: 7.534

STATISTICS FOR FILE /pic/pics/ct/kearney/experts/kearney.wa ONLY

centroid at (119.5,107.1,80.0) (mm)
volume (mm³) : 1091.043
mean intensity : 108.231
median intensity : 108
standard deviation: 8.194

STATISTICS FOR FILE /pic/pics/ct/kearney/experts/kearney.rwd ONLY

centroid at (119.1,104.0,80.0) (mm)
volume (mm³) : 571.178
mean intensity : 111.097
median intensity : 111
standard deviation: 7.524

The following is a sample input file for the comparison of two readers plus the computer, analyzing all subcortical structures:

```
/pic/pics/ct/wetzel/wetzel.ct
3
/pic/pics/ct/wetzel/experts/wetzel.wa
/pic/pics/ct/wetzel/experts/wetzel.jc
/pic/pics/ct/wetzel/computer/wetzelxxx.r40
10
111
112
131
132
135
136
117
118
201
301
```

This produces approximately 10 pages of output, the first of which will look like the following page. Note that the data for in and out of intersection now refer to the three readings, each of whose individual values are given.

OUT OF INTERSECTION OF ALL DATA:

centroid at (110.8,97.0,70.0) (mm)
volume (mm³) : 623.333
mean intensity : 103.897
median intensity : 107
standard deviation: 13.015

IN INTERSECTION OF ALL DATA:

centroid at (110.3,92.1,70.0) (mm)
volume (mm³) : 517.341
mean intensity : 109.537
median intensity : 111
standard deviation: 8.827

STATISTICS FOR FILE /pic/pics/ct/wetzel/experts/wetzel.wa ONLY

centroid at (111.2,94.8,70.0) (mm)
volume (mm³) : 990.940
mean intensity : 109.386
median intensity : 111
standard deviation: 7.839

STATISTICS FOR FILE /pic/pics/ct/wetzel/experts/wetzel.jc ONLY

centroid at (109.4,92.7,70.0) (mm)
volume (mm³) : 667.076
mean intensity : 104.491
median intensity : 108
standard deviation: 13.302

STATISTICS FOR FILE /pic/pics/ct/wetzel/computer/wetzelxxxx.r40 ONLY

centroid at (110.6,94.8,70.0) (mm)
volume (mm³) : 1140.674
mean intensity : 106.455
median intensity : 110
standard deviation: 11.280

4.2.2 The Superstat Program

To compare several region files to one truth file, use superstat. Superstat has an input very similar to the above programs and a sample is listed below. The output will be in chart form with region structures as well as region names. The program requires that “.replace” is in the current directory. The syntax is as follows:

```
superstat <inputfile> <outputfile>
```

The input file must be in the following format:

```
CT-image
n = number of vlist files to be read in, including the truth
truth vlist
remaining vlists [1..n-1 lines]
m = number of region numbers
region numbers [1-m lines]
```

The following is an example input file:

```
/pic3/elldata/wetzel.ct - the CT-image
6 - the number of vlists
/pic/pics/ct/wetzel/experts/wetzel.wa - vlist considered to be truth
/pic/pics/ct/wetzel/experts/wetzel.rd2 - vlists to be compared with truth
/pic/pics/ct/wetzel/experts/wetzel.jc      :
/pic/pics/ct/wetzel/experts/wetzel.rwd    :
/pic/pics/ct/wetzel/experts/wetzel.rd3    :
/pic/pics/ct/wetzel/computer/wetzel.vlist :
4 - number of regions of interests
135 - Rindex
136      :
117      :
118      :
```

A sample output is:

Superstat Version 2

```
/pic3/elmdata/wetzel.ct
```

1. /pic/pics/ct/wetzel/computer/wetzel.vlist
2. /pic/pics/ct/wetzel/experts/wetzel.rd2
3. /pic/pics/ct/wetzel/experts/wetzel.jc
4. /pic/pics/ct/wetzel/experts/wetzel.rwd
5. /pic/pics/ct/wetzel/experts/wetzel.rd3

```
STR 1 2 3 4 5 Name
```

```
135 0.000 0.728 0.717 0.664 0.770 L._LENTICULAR
136 0.000 0.662 0.648 0.673 0.741 R._LENTICULAR

117 0.000 0.825 0.826 0.741 0.806 LEFT_THALAMUS
118 0.000 0.771 0.761 0.774 0.847 RIGHT_THALAMUS
```

4.2.3 The Stat2d2 Program

Stat2d2 allows the user to see the results of statistical comparison slice by slice. Stat2d2 has the same input file as the above programs and a sample is listed below. The output will be in chart form listing the results of each region structure indexed by slice. The syntax is as follows:

```
stat2d2 <inputfile> <outputfile>
```

The following is an example input file:

```
/pic3/elmdata/wetzel.ct
2
/pic3/vlist/wetzel/experts/Wetzel.wa
/pic3/vlist/wetzel/rwdat/wetzel.2.50.vlist
6
111
112
135
136
117
118
```

A sample output of the program is as follows:

```
doing slice 9
Region out_union intersection results
111 217.000000 0.000000 0.000
112 218.000000 0.000000 0.000
135 454.000000 0.000000 0.000
136 562.000000 0.000000 0.000
117 0.000000 0.000000 -----
118 0.000000 0.000000 -----
```

```
doing slice 10
Region out_union intersection results
111 426.000000 0.000000 0.000
112 437.000000 0.000000 0.000
135 1321.000000 1386.000000 0.512
136 1229.000000 1561.000000 0.559
117 210.000000 0.000000 0.000
118 0.000000 0.000000 -----
```

```
doing slice 11
Region out_union intersection results
111 483.000000 0.000000 0.000
112 532.000000 0.000000 0.000
135 1220.000000 1816.000000 0.598
136 830.000000 1818.000000 0.687
117 756.000000 1780.000000 0.702
118 1352.000000 1556.000000 0.535
```

```
doing slice 12
Region out_union intersection results
111 643.000000 0.000000 0.000
112 741.000000 0.000000 0.000
135 838.000000 1204.000000 0.590
136 685.000000 1010.000000 0.596
117 1056.000000 2065.000000 0.662
118 737.000000 2172.000000 0.747
```

```
doing slice 13
Region out_union intersection results
111 660.000000 0.000000 0.000
112 893.000000 0.000000 0.000
135 623.000000 0.000000 0.000
136 587.000000 0.000000 0.000
117 868.000000 1942.000000 0.691
118 930.000000 2011.000000 0.684
```

```
doing slice 14
Region out_union intersection results
111 804.000000 392.000000 0.328
112 818.000000 404.000000 0.331
135 0.000000 0.000000 -----
136 0.000000 0.000000 -----
117 1529.000000 0.000000 0.000
118 2001.000000 0.000000 0.000
```