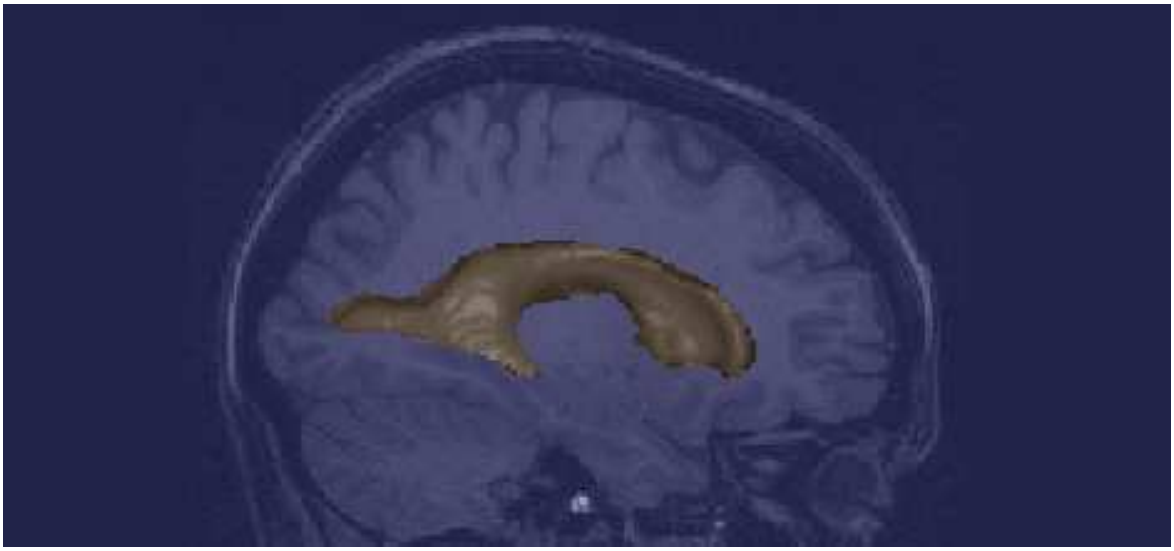
.

IVS

Interactive Volume Segmentation



Quick Guide

Version 1.11, web distribution

October 2003
Felipe P. G. Bergo and **Alexandre X. Falcão**

IVS – Interactive Volume Segmentation

Quick Guide

©2003 Felipe P. G. Bergo and Alexandre X. Falcão

`bergo@seul.org` and `afalcao@ic.unicamp.br`

This guide is available for free download at
`http://www.ic.unicamp.br/~afalcao/ivs`.
Sale of this manual, in either electronic or printed form, is forbidden.

Version History:

2003.10.31	first version of the guide, written for IVS 1.11.
2003.11.04	bibliography corrections, added missing text to sec.3.3.1.

Contents

CHAPTER 1

Installing IVS

A free binary distribution of IVS is available for download from

```
http://www.ic.unicamp.br/~afalcao/ivs
```

This chapter details system requirements and installation procedures.

1.1 System Requirements

The binary distributions of IVS available for download require a computer with at least:

- IA32 CPU with MMX extensions (e.g.: Intel Pentium III, AMD Athlon)
- Linux operating system (with kernel 2.4.0 or more recent)
- X11 windowing system.
- GTK+ 1.2.x GUI library (**not** the 2.x.x series)

IVS itself requires less than 1 MB of disk space, but 3D datasets may need several MegaBytes of storage. The RAM memory required by IVS depends solely on the volume size you'll deal with. A 384-MB RAM computer can handle a 12 Mvoxel dataset without swapping. Using IVS for 3D segmentation with less than 256-MB of RAM is likely to lead to swapping and very poor performance.

While there is no minimum CPU speed requirement, a 700 MHz CPU (or faster) is recommended for reasonable response times.

IVS has been tested only on the 2.4 series of Linux kernels. It may not work on 2.2 and older series.

For proper display, a X11 windowing system configured to a screen resolution of 800x600 (1024x768 or above recommended) is required. A color depth of 16 bpp (64K colors) or more is recommended.

The 1.2 series of the GTK+ library, used by IVS, can safely coexist with the 2.x series, if needed. GTK+ can be found at `http://www.gtk.org`, but most computers with the Gnome desktop environment will probably have it installed already.

1.2 Installing from an RPM package

Users on RPM-based Linux distributions (like Red Hat, Mandrake, SuSE, Conectiva) can download the `ivs-1.11-web.i586.rpm` file and install it with the command

```
rpm -ivh ivs-1.11-web.i586.rpm
```

(which must be executed with superuser privileges). The IVS program will be installed to `/usr/bin/ivs`, and may be started with just `ivs &`.

This RPM package of IVS was generated on Red Hat Linux 7.2, and IVS is linked against glibc 2.2.4. It should be compatible with any RPM-based distribution based on glibc 2.2.4 or more recent.

1.3 Installing from a Slackware package

Slackware Linux users should download the `ivs-1.11-web.tgz` file and install it with the command

```
/sbin/installpkg ivs-1.11-web.tgz
```

(which must be executed with superuser privileges). The IVS program will be installed to `/usr/bin/ivs`, and may be started with just `ivs &`.

This package ws compiled on Slackware Linux 9.0 and IVS is linked against glibc 2.3.1.

1.4 Installing from a simple tarball

Users from other distributions and those who don't have superuser privileges can download the `ivs-1.11-web.tar.gz` and unpack it anywhere with the command

```
tar zxvf ivs-1.11-web.tar.gz
```

This IVS is linked against glibc 2.3.1.

1.5 Sample Dataset

A sample anonymized MRI dataset `head-mri-0001.scn.gz` is available for download. It should be decompressed before being loaded on IVS, with the command

```
gzip -d head-mri-0001.scn.gz
```

CHAPTER 2

User Interface

IVS can be started with the command

```
ivs &
```

When started with no command-line parameters, it will start in the file browser pane, as shown in Figure 2.1a. IVS has 4 operation panes that occupy most of its window space when selected: File browser, preprocessing pane, scene pane and the log pane (Figure 2.1). Panes can be switched by clicking on the tabs on the bottom of the IVS window or by the keyboard shortcuts F5–F8.

IVS can take one command-line argument, which it will try to load as a scene, volume or image file.

2.1 File Browser

The file browser pane allows the user to navigate through the file system and load image/volume data in IVS-compatible formats. The current directory can be changed by clicking on the folder icons, either those on the path representation in the upper portion of the screen (moving up from the current path) or the directories in the current directory.

IVS can load images and volumes in the following formats:

SCN SCN is a volume format used internally at the Institute of Computing (Unicamp) for 3D volume representation. It resembles the PGM/P5

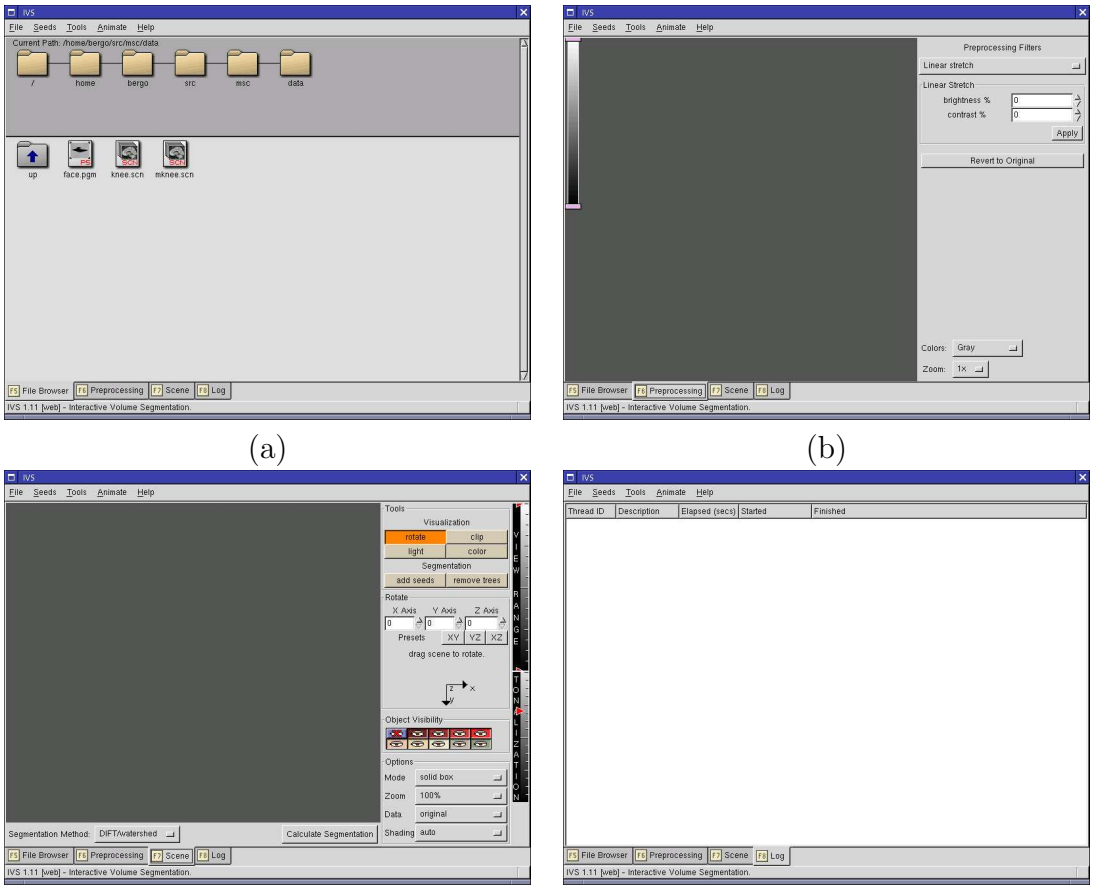


Figure 2.1: IVS Panes

format, except for being 3D, allowing 16-bit data and including information about voxel dimensions in the ASCII header that precedes the raw data section. The example images available for download are in this format.

MVF MVF is IVS's saved session format. It includes the original volume data, the preprocessed volume data, segmentation state, view modes, etc.

PGM IVS can read 2D images in P2 and P5 formats, but cannot assemble a volume from multiple PGM slices.

Analyze 7.5 IVS can read some volumes in the Analyze 7.5 format. Volumes in this format appear as 2 files, one with `.hdr` suffix and another with `.img` suffix. The `.hdr` one should be clicked to load the volume. IVS does not support the SPM variation of the Analyze format.

MINC IVS can read some volumes in MNI's MINC format, but it requires that the `mincextract` and `mincinfo` programs be in the user's PATH, and that Perl 5 or more recent be installed in `/usr/bin/perl`.

To load a image in any of the above formats, click once on the file's icon in the file browser pane, or use the File ▶ Load Scene or Volume... menu command.

2.2 Preprocessing Pane

Once a volume is loaded, IVS keeps two copies of it in main memory: the original volume and the preprocessed one (which, at the first moment is the same as the original). The preprocessing pane provides filters that on most cases take the preprocessed volume as input (also called "current") and overwrite it with the result. The preprocessing pane shows the current volume data in 3 orthogonal cut views (Figure 2.2).

The current position (marked by the green cross-hairs on all 3 views) can be changed by clicking the left mouse button over any of the 3 views. The volume shown by the default is the current (preprocessed) one, but the original image can be viewed by holding the right mouse button anywhere in the canvas area.

Volume intensities are mapped to screen intensities such that the highest value in the volume becomes white, but the viewing range can be modified by dragging the two handles in the view range control in the upper left corner of the canvas area. The Colors control on the bottom right allow the selection of different false-color schemes. The preprocessing pane provides up to 4× enlargement through the Zoom control. When the volume is zoomed, the 3 cut views may not fit the IVS window. The view can be panned by dragging the canvas with the middle mouse button.

The text below the upper-right cut view shwos the voxel intensity at the current cross-hair position. The Revert to Original button copies back the original volume data to the preprocessed volume, undoing the effect of most filters. The controls above the *Revert to Original* button select filters and their parameters. These will be discussed in the next chapter of this guide.

2.3 Scene Pane

The scene pane provides 3D visualization and segmentation tools. This pane provides 6 tool modes, selectable by the orange buttons on the upper right (Figure 2.3):

Rotate This tool rotates the volume around its center. The user can enter

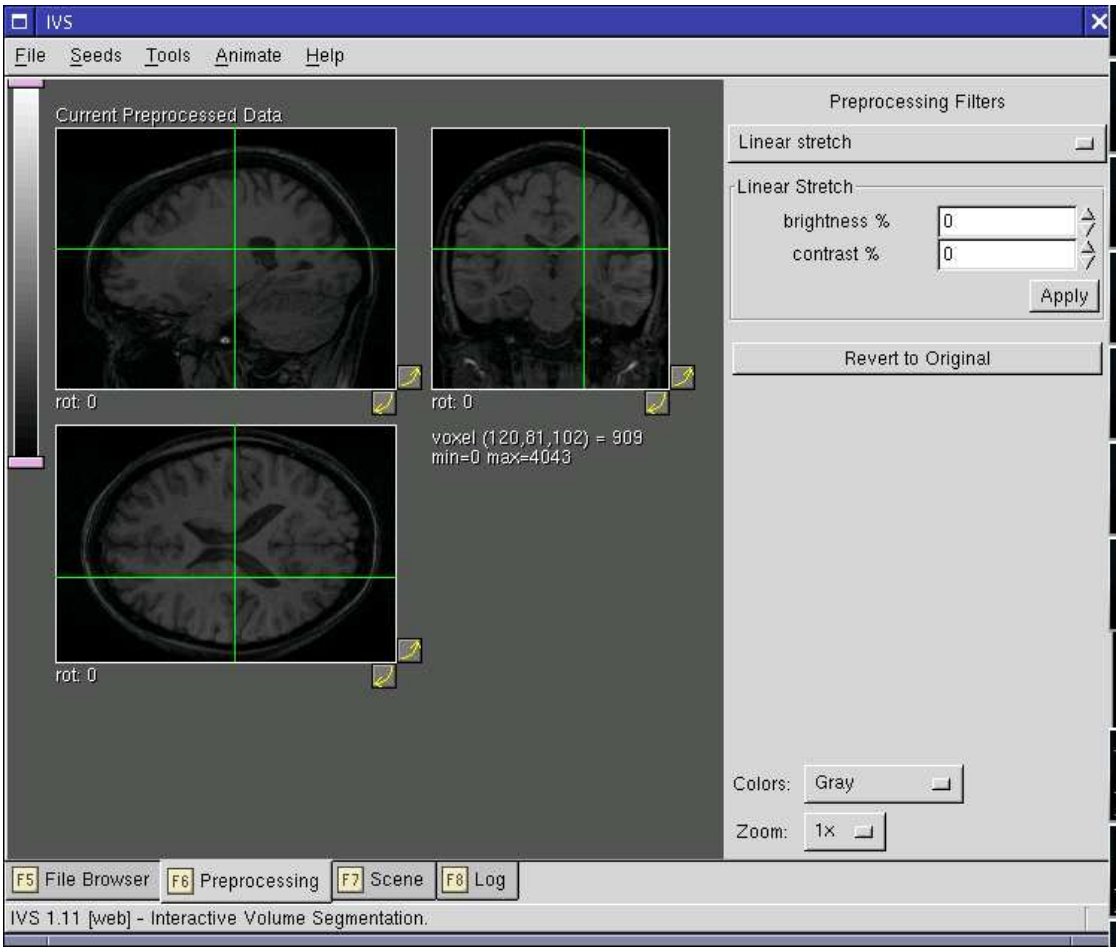


Figure 2.2: Preprocessing pane

angles textually or drag the canvas area holding the left mouse button to change X and Y rotations. While dragging, the scene will switch to a wireframe mode.

Clip This tool clips the viewable region of the volume by an orthogonal rectangular box. Limits may be entered textually, or the clip controls (tracking arrows in the viewing canvas) can be dragged with the left mouse button.

Light This tool controls the shading parameters for the surface rendering mode.

Color This tool provides 4 preset color sets for the canvas background and object labels, and the random button, which creates random color schemes.

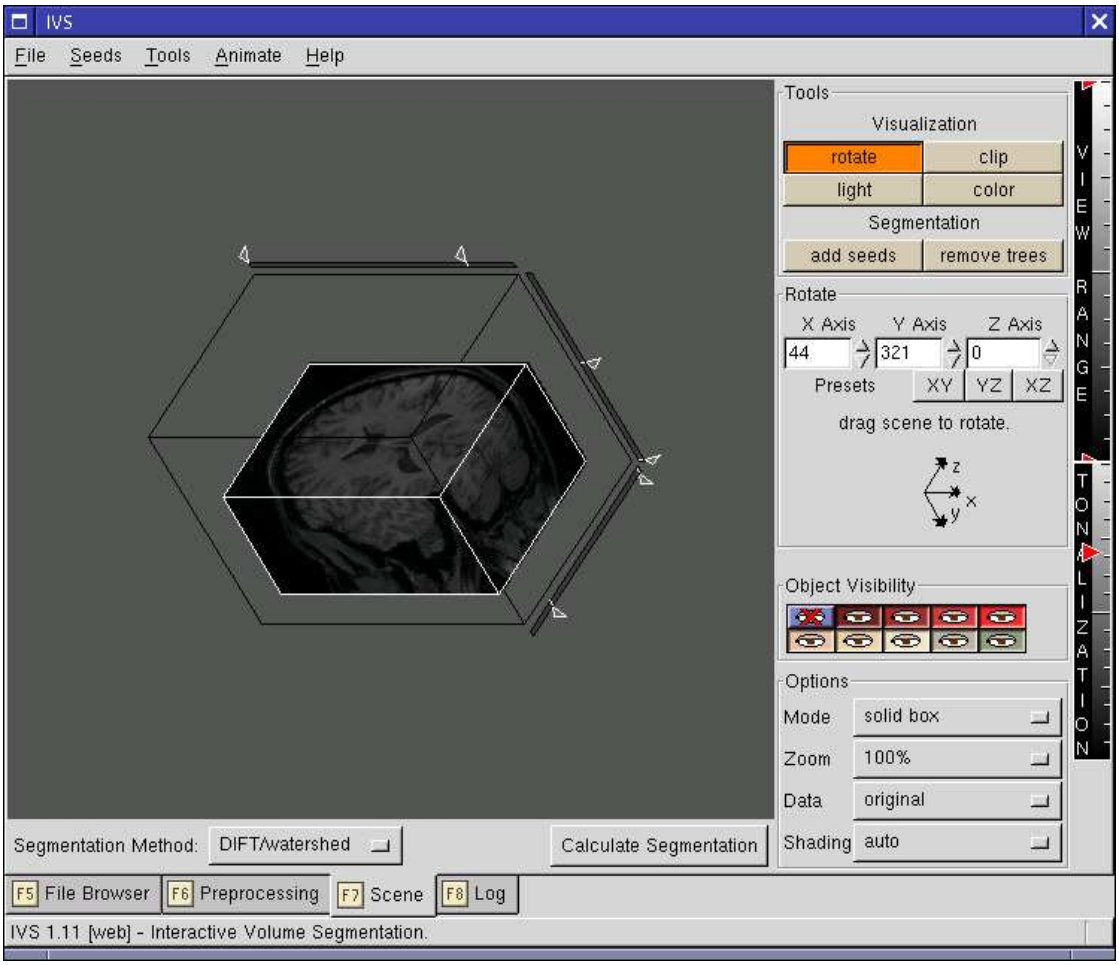


Figure 2.3: Scene pane

Add Seeds This tool allows the selection of seeds for the segmentation algorithms. The user can select seeds by clicking or dragging over the view.

Remove Trees This tool allows the selection of IFT trees for removal with the DIFT algorithm. Trees are selected/unselected by clicking the view with the left mouse button. Selected trees will be shows in inverse colors.

The controls below the tool options allow the selection of visible objects and viewing modes. The most important control is Mode, which defines the rendering mode:

Solid Box No segmentation information is used, the volume data on the faces of the clipping rectangle are displayed. Useful for the first seed selection.

Objects Surface rendering of the segmented objects. Objects marked as not visible in the Object Visibility control are not rendered. A mixture of label color and volume intensity is used on the rendering, the Tonalization control on the right of the screen controls the mixture ratio.

Object Borders Similar to the previous mode, but only frontier voxels between objects are color-coded.

Tree Borders Similar to object borders mode, but tree frontiers in the IFT forest are color-coded.

The Segmentation Method control on the bottom selects the DIFT-based segmentation algorithm, and the Calculate Segmentation button starts the DIFT computation with the seeds and markers currently selected by the *Add Seeds* and *Remove Trees* tools.

2.4 Log Pane

IVS is a multi-threaded application, which allows the user interface to remain responsive while lengthy computations are being carried out. Most computations (rendering, filters, segmentation) generate log events with exact timings. The log pane lists these events. Rendering and visualization are processed by the interface thread, while filtering and segmention are processed by a background thread. The background thread has a task queue. The queue's status is shown on the right end of the status bar at the bottom of the IVS window. Red circles represent tasks queued but not yet started, while the green circle represents the taks currently being processed. Whenever the background thread is active, the green circle will be present, along with a brief textual description of the task and an approximation of the elapsed time. The exact elapsed time should be read from the log pane after the task is completed. The log content can be saved to a text file with the File ▶ Save Task Log... command.

CHAPTER 3

Image Processing Operators

3.1 Preprocessing Filters

3.1.1 Linear Stretching

Appears as *Linear Stretch* on the interface. This is a brightness/contrast adjustment filter. It takes the current volume as input and overwrites it with its output.

3.1.2 Gaussian Stretching

Appears as *Gaussian Stretch* on the interface. This filter changes the voxel intensities from Y to Y' according to the expression below:

$$Y' = K \cdot exp\left(\frac{-(Y-\mu)^2}{2\sigma^2}\right)$$

where μ and σ are the mean and standard deviation values, provided by the user, and $K = 32766$. This filter enhances tissues around the given mean intensity. It takes the current volume as input and overwrites it with its output.

3.1.3 Gaussian blur

This is a $3 \times 3 \times 3$ low-pass convolution filter. It takes the current volume as input and overwrites it with its output.

3.1.4 Median

This is a $3 \times 3 \times 3$ median filter. It takes the current volume as input and overwrites it with its output.

3.1.5 Gradients

This filter group provides 4 different gradient operators. Morphological gradients make the voxel value be the the difference between the maximum (dilation) and minimum (erosion) values within the structuring element. The Morphological $3 \times 3 \times 3$ cross operator uses a cross-shaped strutured element containing the voxel and its 6 closest neighbours, and is the most common gradient. The Morphological $3 \times 3 \times 3$ box operator uses a cubic structuring element containing the voxel and its 26 closest neighbours. It leads to thick borders, but has nasty effects on thin objects. The Directional (maximal) operator takes the directional gradient in all 3 orthogonal directions and picks the maximum value. The Directional (average) operator uses the average of the 3 directional gradients. All gradient operators take the current volume and overwrite it with their results.

3.1.6 Isometric Interpolation

This filter interpolates the volume so that it becomes isotropic (with cubic voxels). It will enlarge the volume so that the side of a voxel in all directions be the lesser dimension on the original volume. A volume with voxel dimension $1.0 \times 1.33 \times 1.50$ will be interpolated to $1.0 \times 1.0 \times 1.0$, for example. This filter overwrites both the original and current volumes with its output, and any segmentation previously done is lost.

3.1.7 Region Clip

This filter crops the volume to a rectangular selection. When this filter is selected, the left and right mouse buttons no longer change the cross-hair position nor show the original volume, but rather select the opposing corners of the rectangle to crop to. This filter overwrites both the original and the current volumes with its output, and any segmentation previously done is lost.

3.2 Saving the Preprocessed Volume

The preprocessed volume can be saved to SCN or raw formats through the File ▶- Export Scene Data... menu command. Data saved by IVS is **always** little-endian (LSB before MSB), regardless of the architecture it is running on.

3.3 Segmentation Operators

IVS's segmentation operators are based on the DIFT algorithm [1], which is itself an extension of the IFT algorithm [2, 3]. The references listed in [1] should provide a reasonable background for those interested.

This version of IVS provides 2 DIFT-based segmentation operators: DIFT/watershed and DIFT/fuzzy. The segmentation operator is selected on the scene pane (Figure 2.3). Once segmentation is started with one operator, it can't be switched to the other, unless the whole segmentation is reset with the File ▶ Reset Scene Annotation command.

3.3.1 DIFT/watershed

The implementation details of this operator are described in [1]. It requires a gradient-like scene, so usually one of the preprocessing filters applied will be a gradient operator. It requires no parameters, so a watershed based segmentation requires only that seeds be marked on the interest objects, and the Calculate Segmentation button will start the segmentation computation right away.

A simple preprocessing recipe for watershed segmentation is:

1. Apply a Gaussian stretch with μ centered on the object(s)'s mean voxel intensity. σ should be enough to keep the object bright.
2. If the object has high frequency regions (like the cerebellum ramifications), apply a smoothing filter like a Gaussian blur or a median filter.
3. Apply a gradient operator to obtain the volume intensities that will be used by the watershed operator.

3.3.2 DIFT/fuzzy

This operator is a simplification of the fuzzy-connectedness model [4, 5, 6, 7, 8]. It requires the selection of some (μ, σ) pairs that represent the objects of interest before the first segmentation iteration starts.

Once the user has selected the first seed set and clicks the Calculate Segmentation button, the dialog box of Figure 3.1 will be shown.

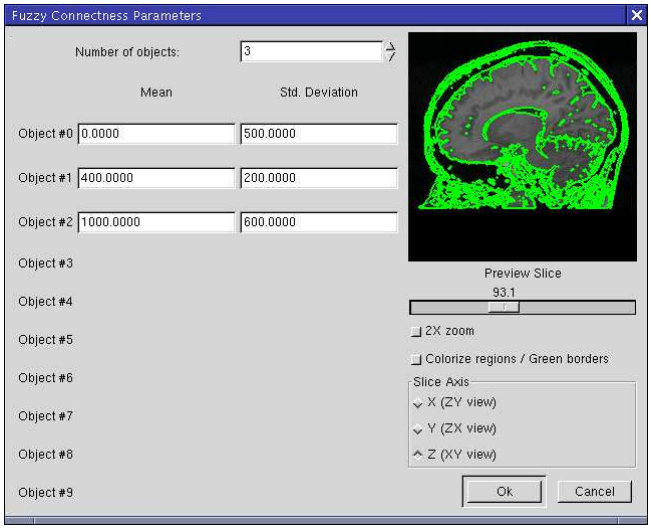


Figure 3.1: Fuzzy Parameters Dialog Box

Usually, the user will select $n + 2$ (μ, σ) pairs, n being the number of objects being segmented. Each object will lead to a (μ, σ) pair, plus one pair representing tissues darker than all objects and another pair representing tissues brighter than all objects. The example in Figure 3.1 shows a reasonable parameter selection to segment the lateral ventricles.

The green borders on the thumbnail image show the borders of the influence regions of each (μ, σ) pair. This image is over-segmented, and serves only as a guide. The user can obtain this preview for all slices, in any one of the 3 orthogonal directions. The user can also toggle the Colorize regions / Green borders control to change between the preview modes shown in Figure 3.1 and Figure 3.2.

These parameters must be chosen only before the first iteration. This dialog box can be brought up again anytime by the Tools ▶ Fuzzy Parameters Dialog... command, but changing the parameters in the middle of a segmentation session will lead to non-optimal forests and unpredictable results. If you really want to use the current seed/root set with different parameters, the proper way to proceed is to save the root set to a file with the Seeds ▶ Save Roots... command, reset the segmentation

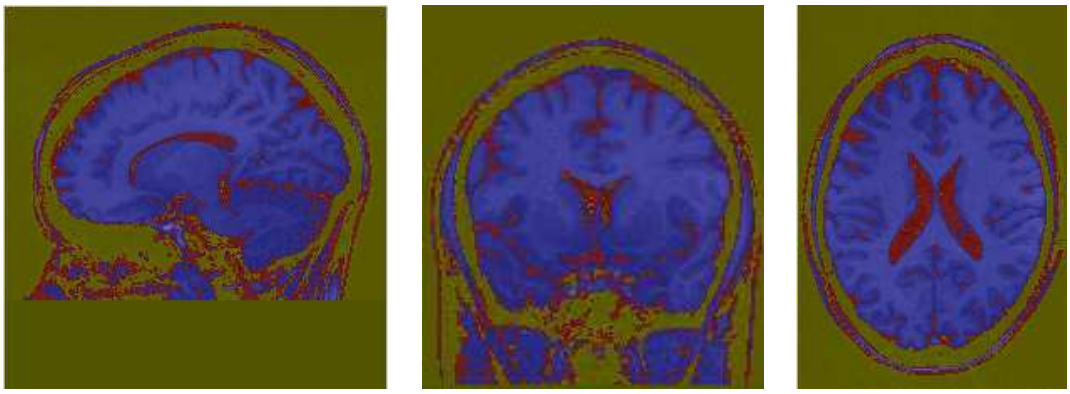


Figure 3.2: Previews in different directions with region coloring.

with the File ▶ Reset Scene Annotation command, load the seeds with the Seeds ▶ Load Seeds... command and then click the fboxCalculate Segmentation button to start a new segmentation (the Fuzzy parameters dialog will be brought up again, since this is the first iteration of the new segmentation).

Fuzzy-connected segmentation does not require gradient-like scenes. Its success depends on the voxel intensity homogeneity within the objects being segmented. Sometimes it can be applied over the volume with no preprocessing at all, but smoothing filters like median and Gaussian blur may help.

3.4 Saving Segmentation Results

The segmentation session can be saved to a MVF file with the File ▶ Save Scene... command. The MVF format saves most of the IVS desktop state: segmentation state (with the correct maps that represent the IFT forest), view state, current and original volumes, etc. The major lacking at the current version (due to the recent addition of DIFT/fuzzy) is that the MVF format does not save the DIFT/fuzzy parameters ((μ, σ) pairs), thus if a fuzzy-segmented scene is saved and reopened later by another IVS instance, the fuzzy parameters must be set again in the Fuzzy Parameters Dialog before new segmentation iterations are computed and, as has been said before, picking different parameters will lead to incorrect and unpredictable results.

The current rendition of the Scene pane can be saved to a PPM (P6) image with the File ▶ Export Scene View... command. The generated image does not have the clipping guides. If those guides are desired, use a screen capture tool like the Gimp [1].

[1] `http://www.gimp.org`

CHAPTER 4

Interactive Segmentation Examples

4.1 DIFT/Watershed Segmentation Example

In this example we'll segment the brain and the lateral ventricles with DIFT/watershed.

Run IVS and load the sample volume `head-mri-0001.scn`. You can do it with the command line

```
ivs head-mri-0001.scn &
```

and then switch to the preprocessing pane (Figure 4.1a).

Inspection of the white matter values shows intensities around 1500. First, apply a Gaussian stretch filter with mean 1500 and standard deviation 350 (Figure 4.1b). Next, apply a morphological $3 \times 3 \times 3$ cross gradient (Figure 4.1c).

Switch to the Scene pane, and use the rotate and clip tools to find a reasonable axial cut where both brain and lateral ventricles are visible (Figure 4.1d) – the exact values for the view in this figure are rotation angles 82, 339, 0; and clip bounds 0,209; 71,144; 0,155.

Change the Zoom control to enlarge the view (Figure 4.1e), and select the add seeds tool. Add seeds for the background (make sure to cross the skull and skin with it), brain (WM+GM) and lateral ventricles (which we will denote LV from here on), with 3 different labels. We have used 0 (BG), 6 and 9 in this example (Figure 4.1f).

Make sure the selected segmentation method is *DIFT/watershed* and start the DIFT computation by clicking the Calculate Segmentation button. Once the computation is

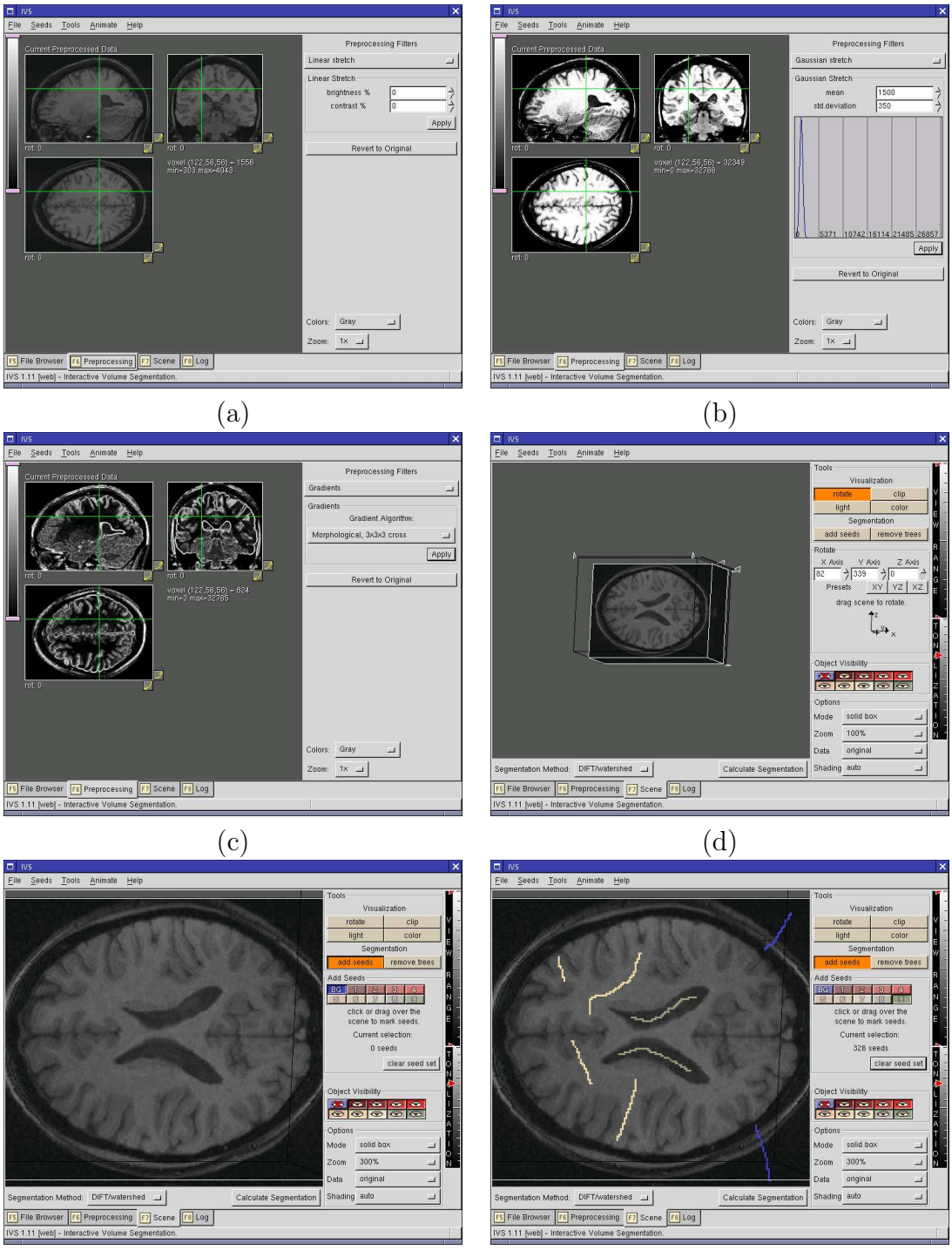


Figure 4.1: DIFT/Watershed Segmentation Example

completed (after about 14 seconds on a 1100-MHz Athlon), IVS switches automatically to *Objects* mode and you will see the view of Figure 4.2a.

Reducing the Tonalization control on the right edge of the screen and/or making the background object visible (by clicking the only button with a red cross on the *Object Visibility* control) will allow better verification of the segmentation correctness in this mode (Figure 4.2b). It can be seen that some brain tissue was lost at the lower left part of this view. Let's correct it with a differential IFT: select the add seeds tool and mark seeds as shown in Figure 4.2c. Clicking the Calculate Segmentation button computes the correction (under 1.2 seconds in a 1100 MHz Athlon) and leads to the state shown in Figure 4.2d.

We are not going to show a full segmentation session (which can take up to 100 corrections), but rather give examples of the views used for verification: switching to *Object Borders* mode we have the view shown in Figure 4.2e.

Slice-by-slice verification can be carried out by selecting the clip tool and moving the proper clipping limit with the associated spin button control (Figure 4.2f, spin button control marked with red circle).

The *Objects* mode can be used to display lone objects. Let's use it to display only the LV object: switch to *Objects* mode, change the zoom one level down, and click on the visibility controls for the background and brain labels, making them invisible. Then select the clip tool and click the Full Volume button to reset the clip to the whole volume. The result is shown in Figure 4.3a.

As an example of tree removal, select the remove trees tools and click over the LV's tail. The corresponding IFT tree will be highlighted, as in Figure 4.3b. Clicking the *Calculate Segmentation* button would compute the optimal path forest for the current root set, excluding the root of the highlighted tree. This is useful for removing influence zones that leaked too much or were marked in the wrong place by misjudgement or due to bad mousemanship.

As the view in Figure 4.3a shows, a minor region of the LV wasn't segmented (small failure on the upper left). This is a typical artifact from partial volume effect. To correct it, let's get a coronal cut view of the problematic region using the rotate and clip tools. Figure 4.3c shows the resulting view. Figures 4.3d–f show the seed marking and the segmentation results.

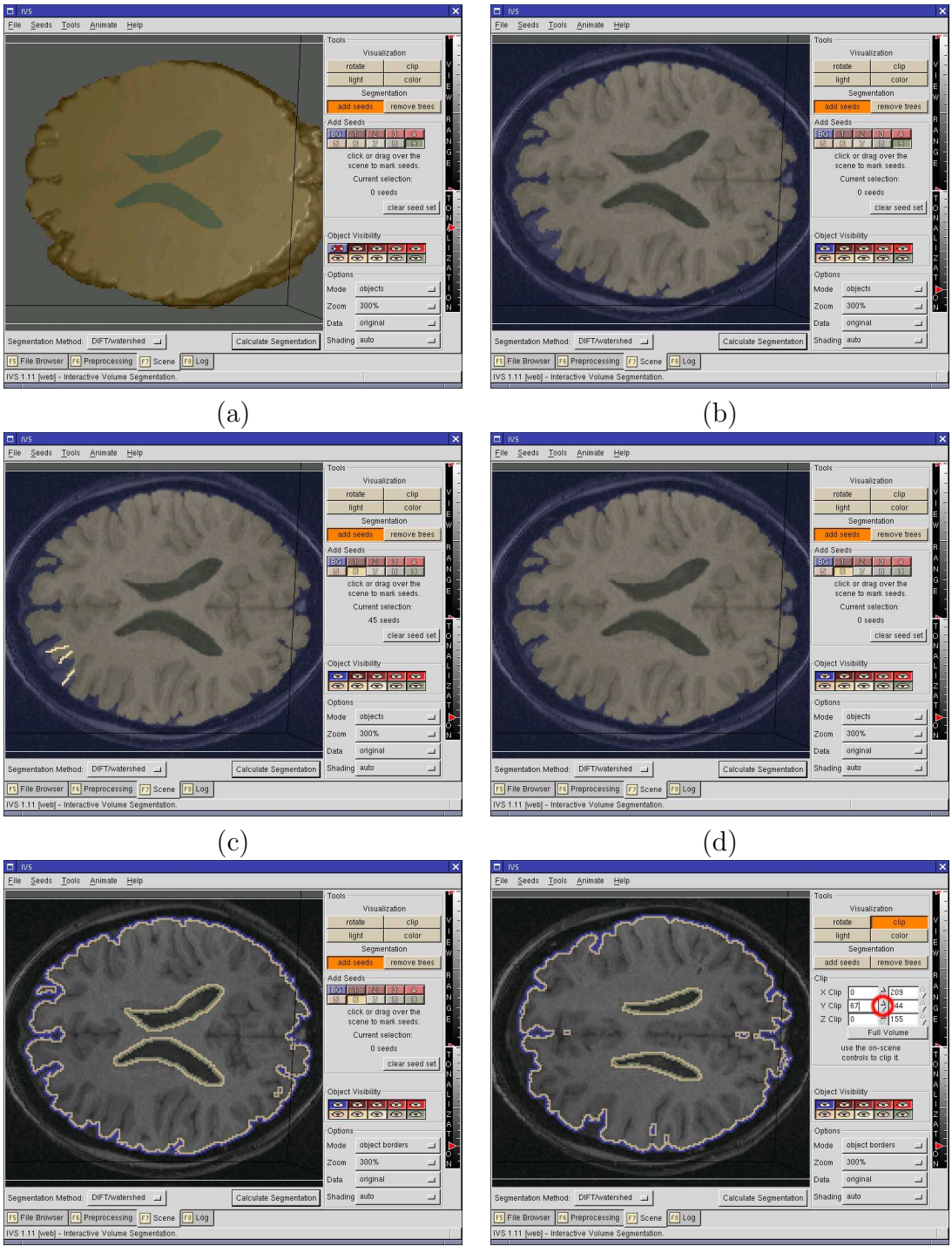


Figure 4.2: DIFT/Watershed Segmentation Example (continued)

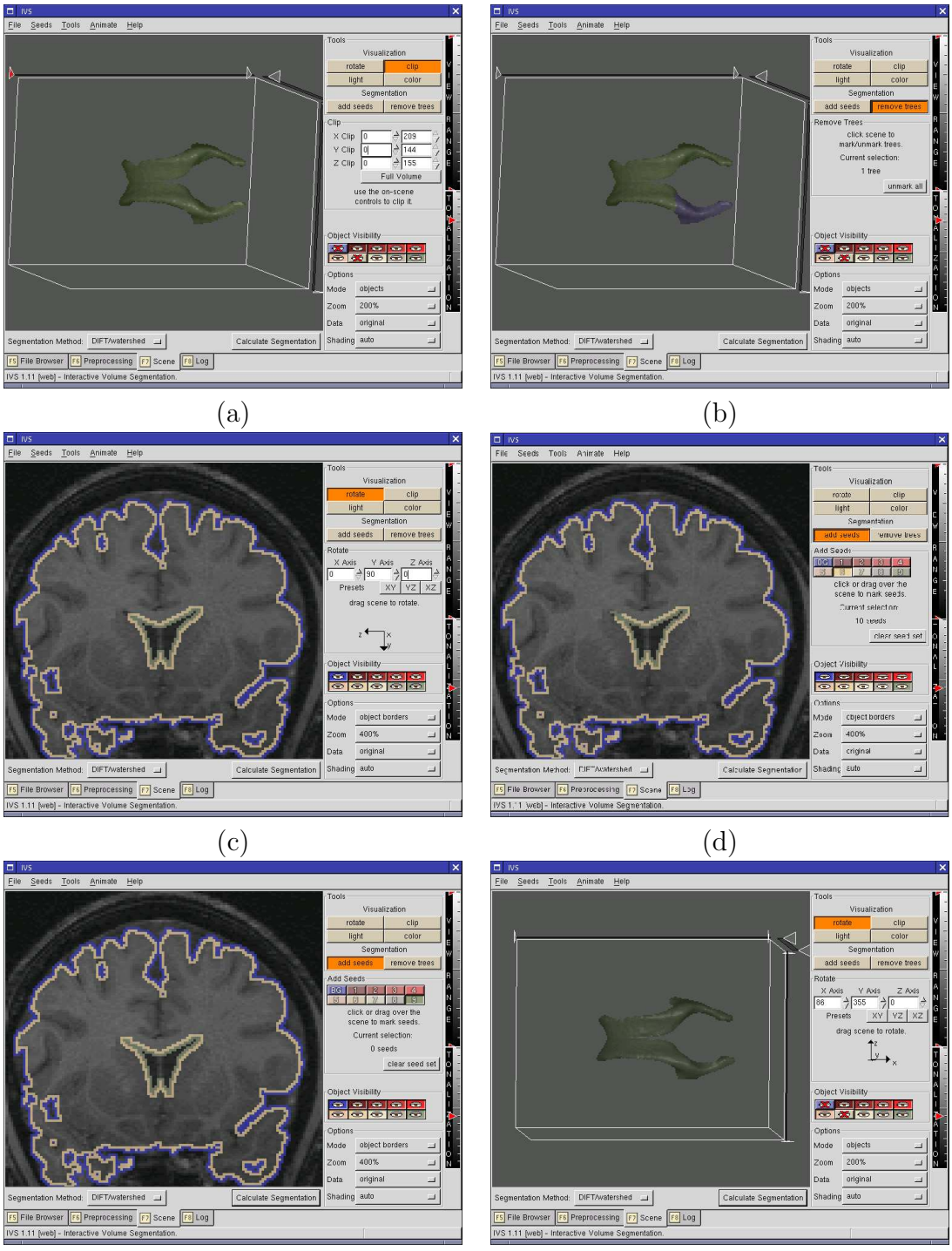


Figure 4.3: DIFT/Watershed Segmentation Example (continued)

4.2 DIFT/Fuzzy Segmentation Example

The correction and verification procedure shown in the previous section is independent of the segmentation operator. In this section we will show the steps of a DIFT/fuzzy segmentation of the lateral ventricles up to the first DIFT iteration only.

To start the segmentation, run IVS with the command line

```
ivs head-mri-0001.scn &
```

We will use no preprocessing in this segmentation, and thus we can go straight to the Scene pane and use the rotate and clip tools to obtain an axial cut view of the volume (Figure 4.4a).

Voxel intensity inspection (which can be performed on the scene pane by moving the mouse over the view holding the right button down) shows that the ventricle tissue varies in the 600–900 range, and GM tissue is in the 1200 range. Darker tissues (CSF, background) have intensities 400 and below. Now we select the add seeds tool and mark seeds in the LV with one label and outside it with another (Figure 4.4b). We make sure that the *Segmentation Method* control has *DIFT/fuzzy* selected and click the Calculate Segmentation button. The Fuzzy-Connectedness parameters dialog is brought up (Figure 4.4c). We raise number of objects to 3 and enter the object means found earlier by inspection: 400, 750 and 1200. The 1200 value showed itself too low on the preview, and we raised it to 1400, resulting in a good preview frontier between the LV and the WM/GM tissue (Figure 4.4d). After clicking Ok the DIFT starts being computed. Once finished (took 28 seconds on a 1100-MHz Athlon), this first iteration led to the segmentation shown in Figures 4.4e–f. From here on, interactive corrections happen in the exact same way as in the DIFT/watershed approach.

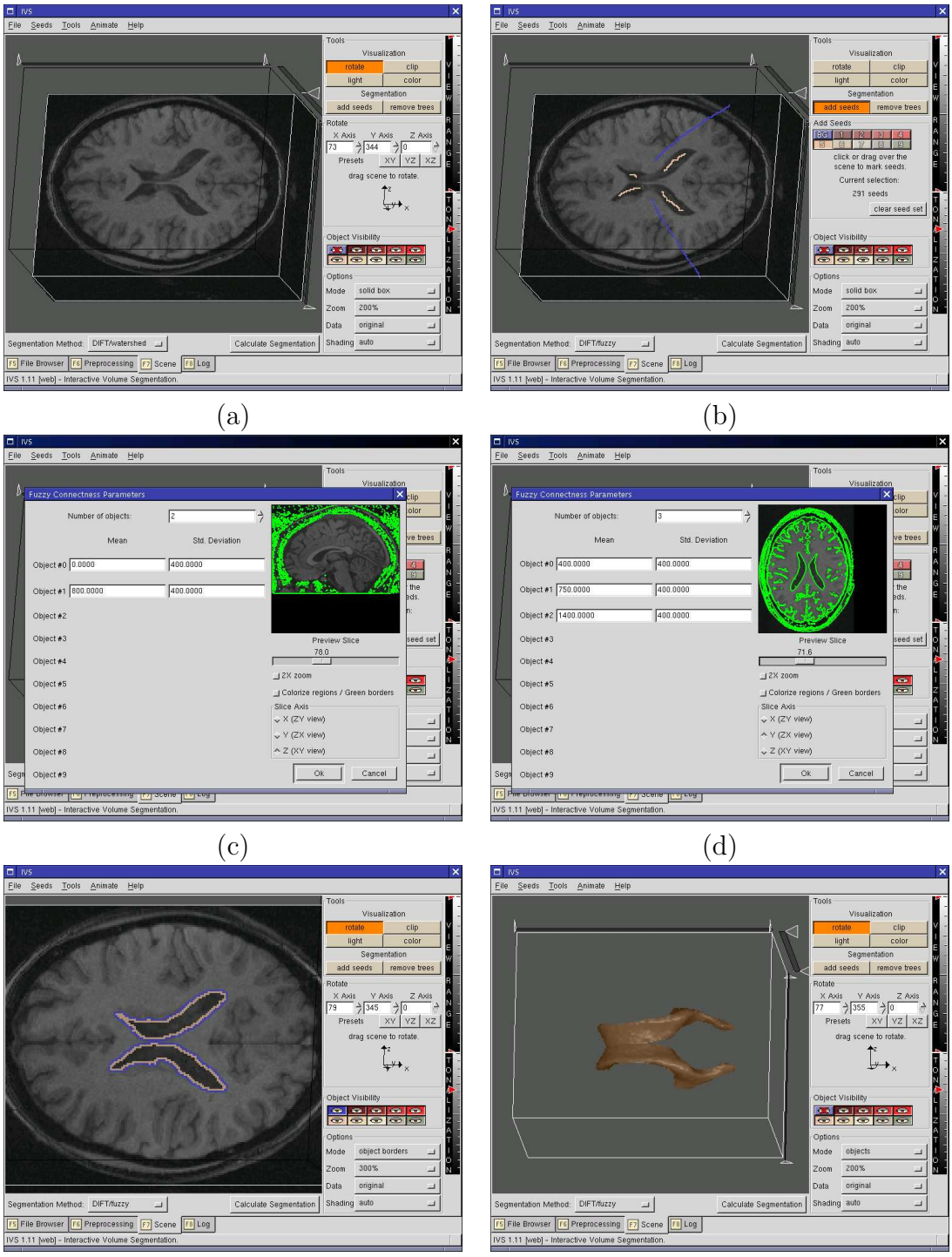


Figure 4.4: DIFT/Fuzzy Segmentation Example

Bibliography

[1] F.P.G. Bergo and A.X. Falcão. Interactive 3D segmentation of brain MRI with differential watersheds. Technical Report IC-03-16, Institute of Computing, University of Campinas, July 2003. Available from http://www.ic.unicamp.br/ic-tr.

[2] A. Falcão, J. Stolfi, and R. Lotufo. The IFT project. Available: `http://www.ic.unicamp.br/~afalcao/ift.html`, 2003.

[3] A.X. Falcão, J. Stolfi, and R.A. Lotufo. The image foresting transform: Theory, algorithms and applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(1):19–29, January 2004.

[4] J.K. Udupa and S. Samarasekera. Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. *Graphical Models and Image Processing*, 58:246–261, 1996.

[5] J.K. Udupa, P.K. Saha, and R.A. Lotufo. Fuzzy connected object definition in images with respect to co-objects. In *Proc. of SPIE on Medical Imaging*, volume 3661, pages 236–245, Feb. 1999.

[6] P. Saha and J.K. Udupa. Scale-based fuzzy connected image segmentation: Theory, algorithms and validation. *Computer Vision and Image Understanding*, 77(2):145–174, 2000.

[7] P.K. Saha and J.K. Udupa. Relative fuzzy connectedness among multiple objects: theory, algorithms, and applications in image segmentation. *Computer Vision and Image Understanding*, 82:42–56, 2001.

[8] L.G. Nyúl, A.X. Falcão, and J.K. Udupa. Fuzzy-connected 3D image segmentation at interactive speeds. *Graphical Models*, 64(5):259–281, 2002.