
Seg3D

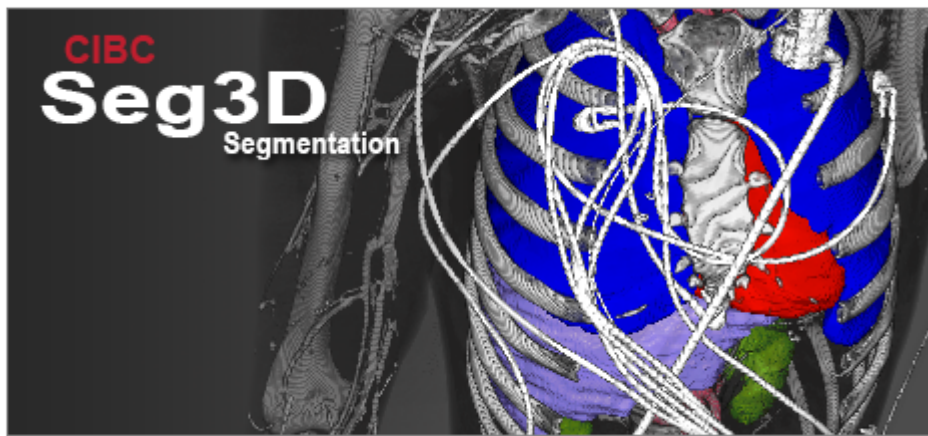
Release 0.01

To be Added

Apr 07, 2023

USER DOCUMENTATION:

1	Installation	3
2	User Documentation	5
3	Indices and tables	119
	Bibliography	121



Seg3D (www.seg3d.org) is a free volume segmentation and processing tool developed by the NIH Center for Integrative Biomedical Computing at the University of Utah Scientific Computing and Imaging (SCI) Institute.

INSTALLATION

Check the *Platform Specifications* for system requirements.

Installers are provided for Windows and Mac OS X. Linux users need to build Seg3D from *source*

USER DOCUMENTATION

A *basic functionality manual*, *tool and filter reference*, and *tutorial* are available.

2.1 Basic Functionality

This document is meant as a basic reference and walkthrough of the Seg3D layout and functionality. Specifically, every feature of the software will be addressed in a brief manner. Tool and filter documentation are found in the *Seg3D Tool Documentation*. Note: If you are looking for a refresher on the available keyboard and mouse shortcuts, there is a list under the **Help** menu of the Seg3D.

2.1.1 About Seg3D

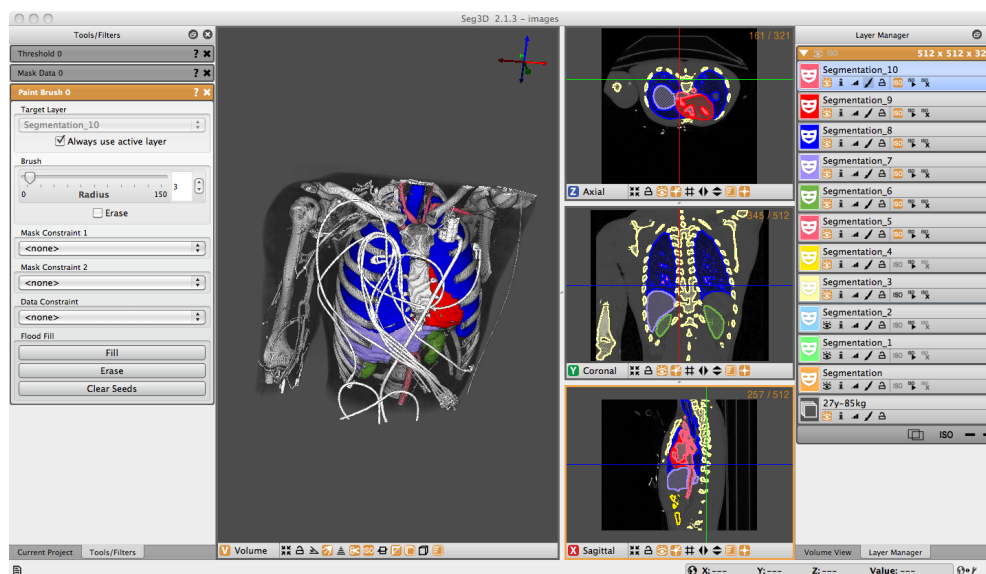


Fig. 2.1: Seg3D in use.

Seg3D is a lightweight software tool developed for use in visualizing and segmenting image data. The core intended use of Seg3D involves loading 3D scalar data, such as MRI or CT scans, and generating labels mask to identify various regions of interest in the original image data. Seg3D facilitates the process using interactive tools such as image processing filters and manual masking techniques.

In addition to manual segmenting, the current version of Seg3D includes some features that facilitate automated segmenting. Provenance is a recently added feature which tracks the steps used to create label masks. This feature allows

the user to then repeat the same sets with different parameters using the Python and Controller windows. Future extensions of these capabilities will allow for scripting in the terminal and porting to other programs such as SCIRun and VisTrails.

Seg3D is an crucial element of our software suite also involving BioMesh3D and SCIRun that make possible the development of image based computational models. Label masks generated with Seg3D are used in other software for such applications as 3D visualization or computational modeling. BioMesh3D requires a segmentation like those from Seg3D to generate high quality, multi-material computational meshes. SCIRun is a modular problem solving environment that generates and runs visualization and simulation tasks on segmentations from Seg3D and meshes from BioMesh3D. This interaction makes Seg3D a key step in image based modeling.

2.1.2 Software Requirements

Seg3D is distributed as a binary download for Linux, Windows, and OS X. Please visit the [SCI software portal](#) to download the latest [Seg3D binary](#).

2.1.3 Functions:

Basic Program Functions

This section gives a brief overview of the options found under the File, Edit, and Help menus. These menus contain options that allow you to save and open new projects, and set preferences.

Edit

The Edit menu contains several features which are convenient for working on a project.

Undo

This function allows the user to undo a changes made to the unsaved project. Changes made to the project are stored in a buffer, the size of which can be changed under Seg3D preferences. This function can be called using the shortcut CTRL/CMD+Z.

Redo

Redo an operation on a project that was previously undone. This function can be called using the shortcut Shift+CTRL/CMD+Z.

Copy Mask Slice

Copy a single slice from a mask layer which can then be pasted onto other layers. This function can be called using the shortcut CTRL/CMD+C.

Paste Mask Slice

Paste a previously copied label from one slice of a mask label to all the slices of another mask label. This function can be called using the shortcut CTRL/CMD+V.

Punch Through Volume

After creating new label data on a single slice of a mask layer, that label can be copied or “punched” to every other slice by selecting this option. This function can be called using the shortcut CTRL/CMD+P.

File

The File tab contains all the features necessary to manage Seg3D projects and files

New Project

The new project option will prompt you to save your current project such that it can be closed. A dialog box will then open asking for a Project name and a Project Path (Fig. 2.2). The Project name will be used to create a folder for all the project data. The Project path defines the location of this folder, which can be set by typing the path or through a separate dialogue box by pressing 'Choose Alternate Location'. The default location for this path can be set in the preferences section discussed below. A new project can also be created using the shortcut CTRL/CMD+N.

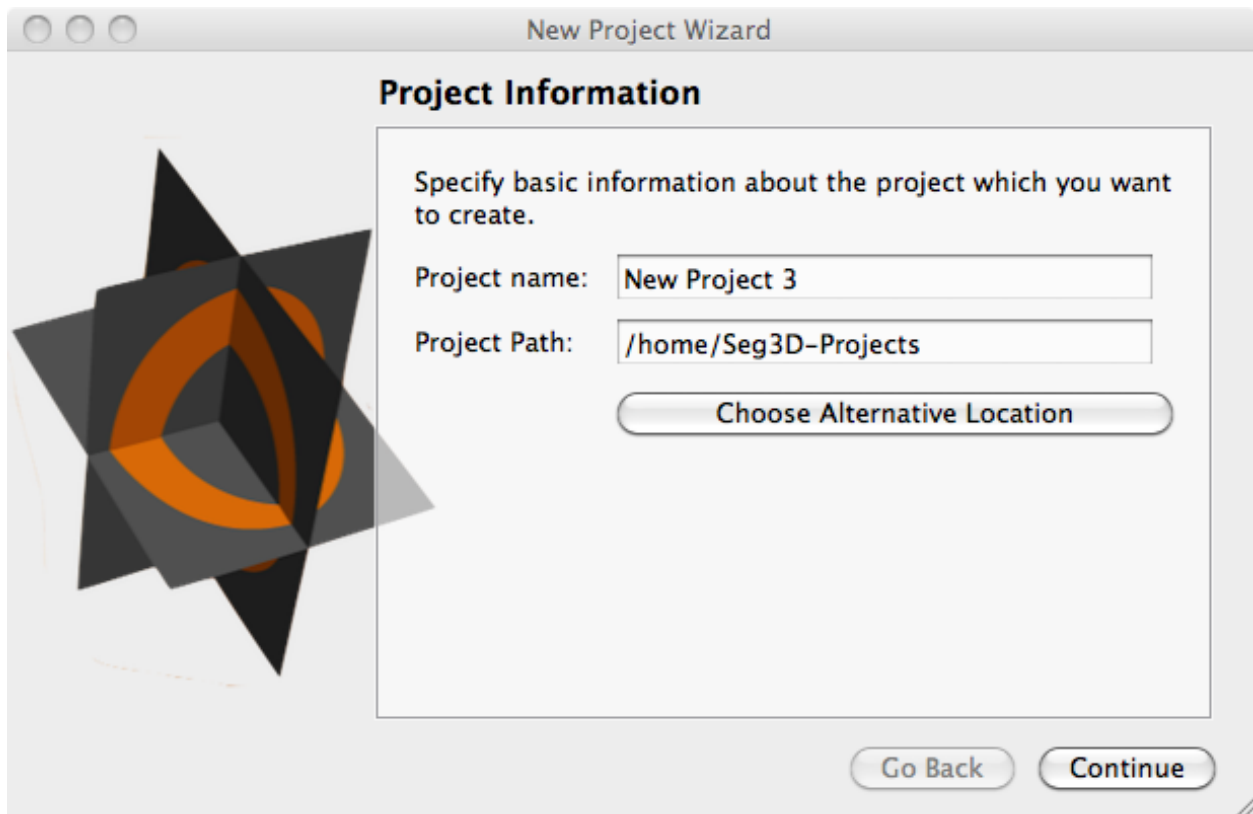


Fig. 2.2: Project Information window is shown when a new project option is chosen. The project name and location are set in this window.

Open Project

The open project selection will prompt you to save you current project and open a new project based by selecting it from a file menu. This function can be called using the shortcut **CTRL/CMD+O**.

Show Project Folder

This option will open a file menu showing the projects contained in your default project location. This function can be called using the shortcut **Shift+CTRL/CMD+O**. The default project location can be changed in Seg3D preferences.

Save Project

If the project as already been given a name and location, this will save the project. If the project has not yet been named, it will prompt the user for a project name and location. This function can be called using the shortcut **CTRL/CMD+S**.

Save Project As

This will create a copy of the current project under a new name and project location. This function can be called using the shortcut **Shift+CTRL/CMD+S**.

Launch Another Copy of Seg3D

This opens a new session of Seg3D. It does not close the current session, but simply opens another copy.

Import Layer From Single File

Data and label maps can be stored in a variety of file formats. Use this option when the data or label map is stored in one single file. This function can be called using the shortcut **Shift+CTRL/CMD+O**.

Import Widget

After the user selects the file/files to be read, a menu will appear asking the user to define the type of data being read as seen below (Fig. 2.3).

There are two types of files that can be read by Seg3D: Image data and label maps. The first selection indicates that the file you wish to read in contains image data. The next three selections all apply to label maps. The second selection assumes that all non-zero values are part of a single label map. The third selection will define each unique value in the file as part of a separate label map. However, the numerical value assigned to represent that label will be incrementally generated by Seg3D rather than assigned from the input file. The fourth selection is similar to the third in that it creates labels based on unique values in the input file. However, in this case the value used to represent the label is the same as the value being read from the file.

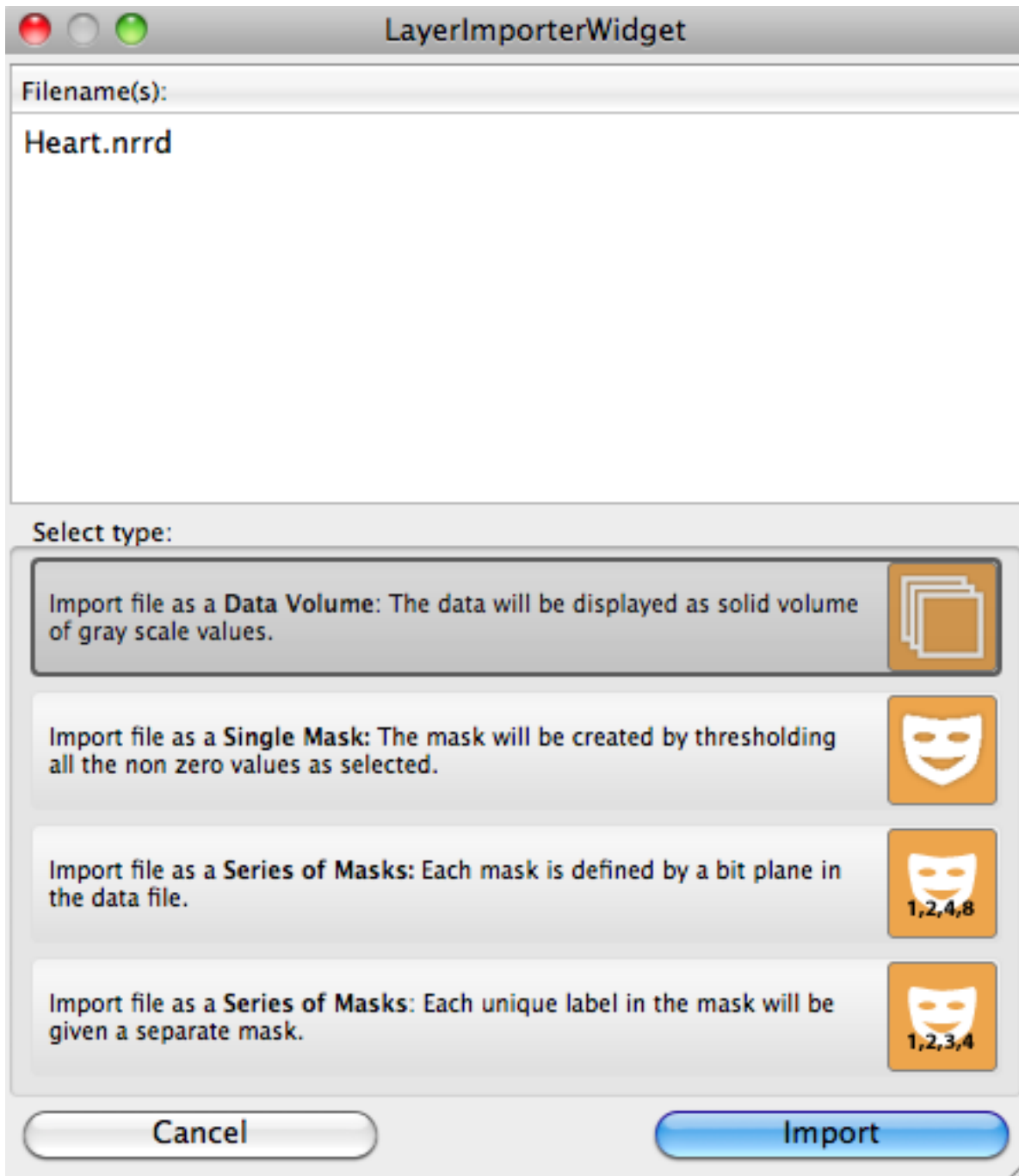


Fig. 2.3: When importing data or label masks, the user is prompted to define the type of data being read.

Import Layer From Image Series

Data and label maps can be stored in a variety of file formats. Use this option when the data or label map is stored in multiple files. See Import Layer From Single File for more information about the Import Widget. This function can be called using the shortcut Shift+CTRL/CMD+I.

Export Segmentation

Label maps can be exported in a variety of file formats. If a layer containing a label map is selected, then the user may select the Export Segmentation option which opens the segmentation dialogue (Fig. 2.4).

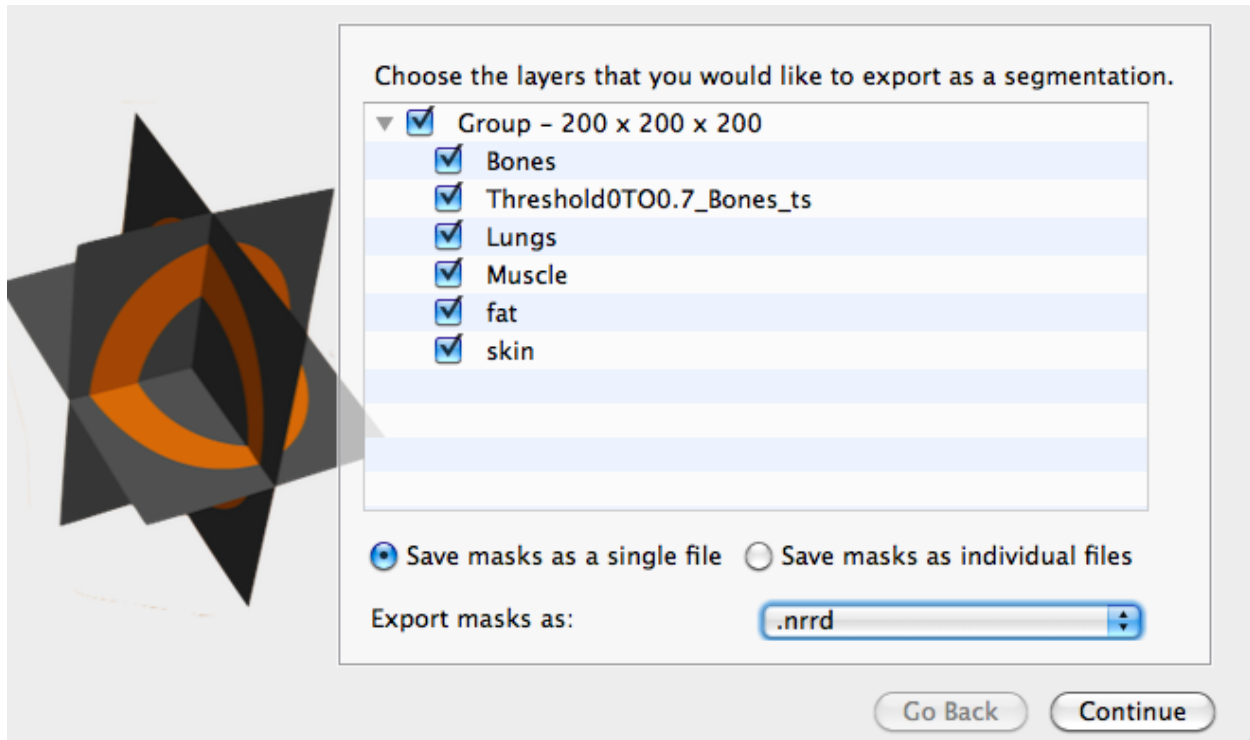


Fig. 2.4: This window appears when the user selects export segmentation. It allows the user to select the layers they wish to export as label masks and the format they wish to export.

This window allows the user to select the layers to export as label masks and the format they wish to export. The available formats include: .nrrd, .mat (matlab), .tiff, .bmp, .png, .dcm. This function can be called using the shortcut CTRL/CMD+E. The segmentation may be saved as a single file or as multiple files.

After choosing a file name (single file only) and location for the segmentation, you will be shown the Export to Segmentation Summary. This window shows the layers chosen to save as a segmentation. In the case of saving as a single file, there is an additional layer (the background, which is the remainder of the volume) and an option to choose the value to represent each of the layers (Fig. 2.5). It should be noted that the segmentation will only represent one value per voxel, so if any of the selected layers overlap the higher value will overwrite the region of overlap. If saving as multiple files, this window will show the names of the layers only.

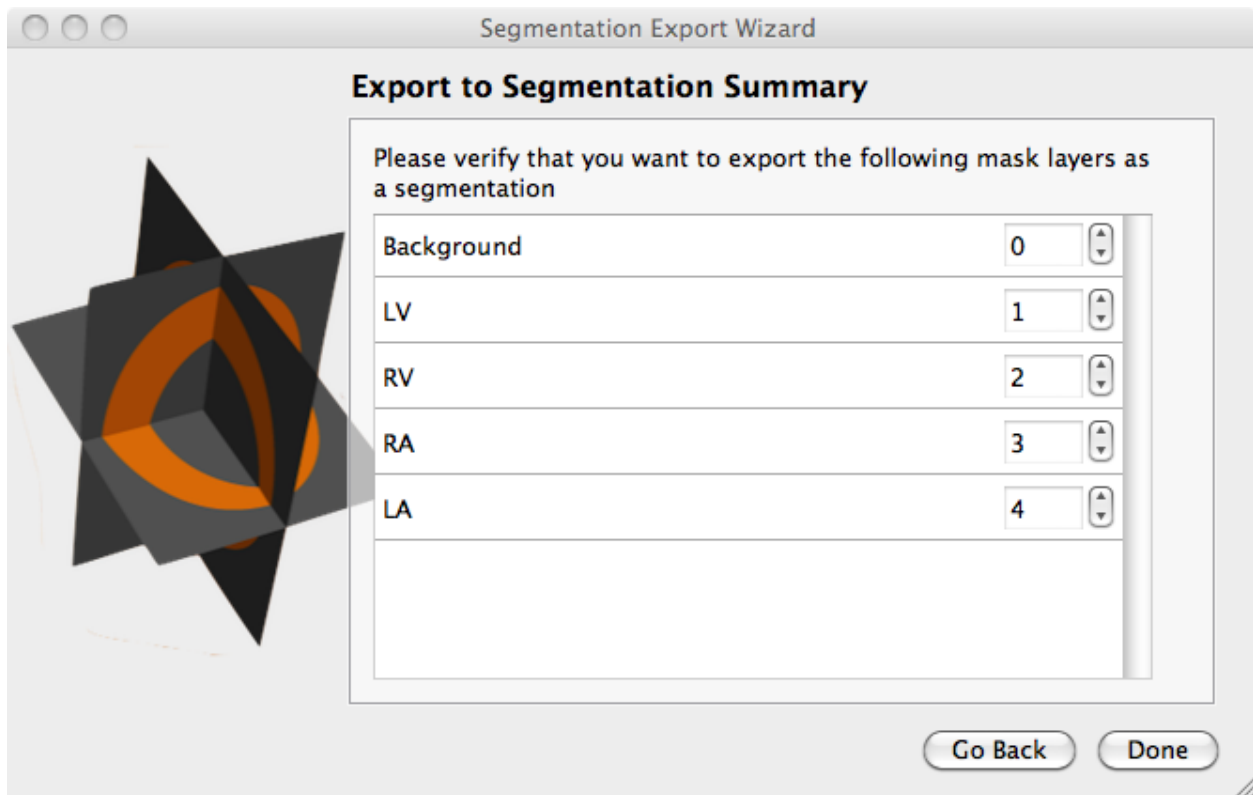


Fig. 2.5: This window appears when the user selects export segmentation and a file is chosen. This allows the users to choose the labels to use for each material.

Export Active Data Layer

If image data layer is selected, then the user may choose to Export Active Data Layer. This selection opens a file menu window where the user can select the location they wish to save the data file as well as specify the file format. The available formats include: .nrrd, .dcm, .tiff, .png, .mrc, and .mat (matlab). This function can be called using the shortcut Shift+CTRL/CMD+E.

Recent Projects

This menu shows recently opened projects. These projects can be opened quickly by clicking on the name of the project.

Help

This menu provides additional information on other features and modules. It has two basic functions:

Search

This feature allows you to search all the menus by typing in a search box.

Keyboard Shortcuts

This function opens a list of all the keyboard and mouse shortcuts available in Seg3D.

Preferences

The preference dialogue box can be opened by selecting **Seg3D Preferences** on Mac OS X. This dialog box contains four tabs which we will discuss here.

General

The in Seg3D preferences, the general tab allows you to change many details of the “autosave” options. The user may turn on the autosave feature and set a frequency for which the project is save. The “Don’t interrupt me” option can be turned on so that the auto save does not happen while another operation is being performed. Along with the frequency the amount of compression used to save each project can be controlled in this tab.

The **Import/Export** options allows you to turn on an off a feature that will export a dicom header with all dicoms that are exported.

The **Project** options allow you to change the default project location, change the buffer size, and change how the project is saved e.g. change if the input files are embedded into the project.

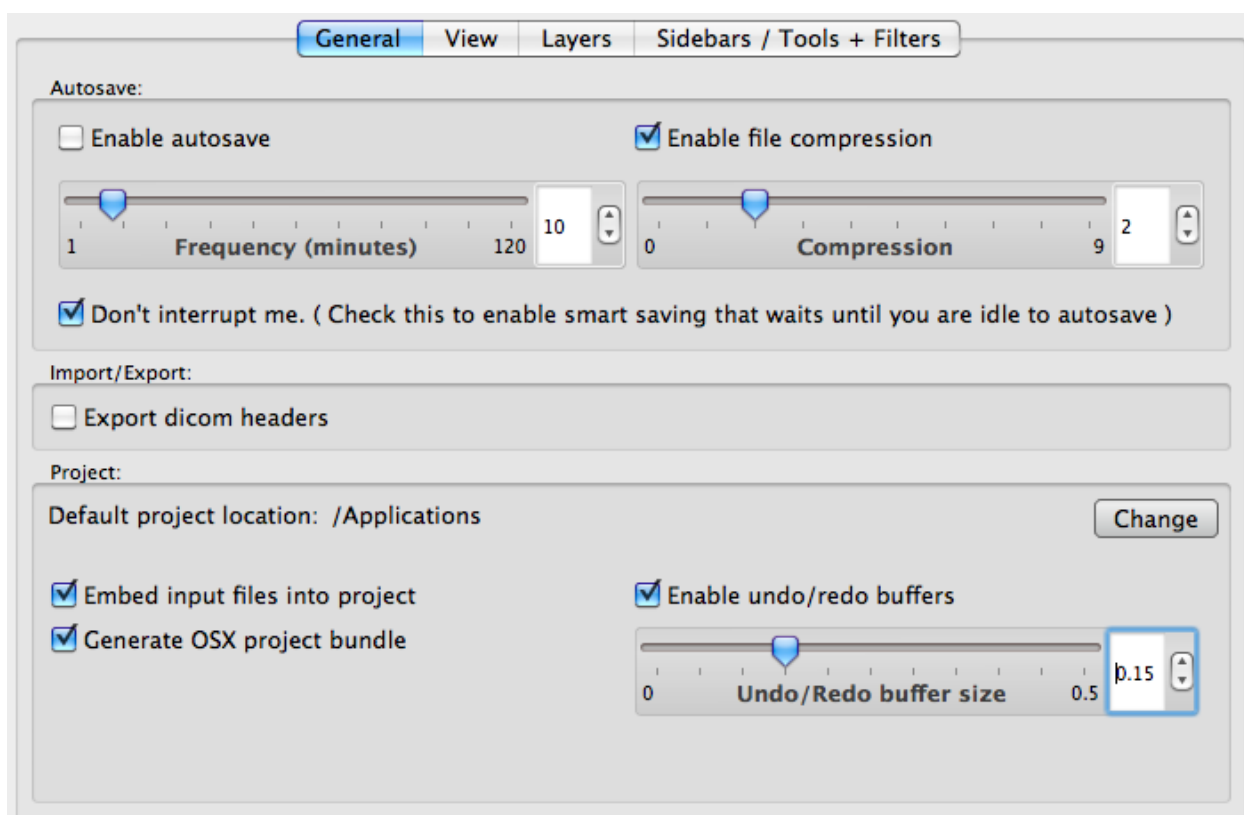


Fig. 2.6: The Seg3d preferences under the general tab.

View

Under this tab the user is able to change many of the default view options such as the window layout (1 big window with 3 smaller windows), background color, and Grid size. In addition there are many options to customize what is seen in the visualization window. At the bottom of this window the user can change the axis labels such that the X, Y, and Z directions are not fixed to represent Sagittal, Coronal, and Axial, but the user can define which axis label is assigned to each direction.

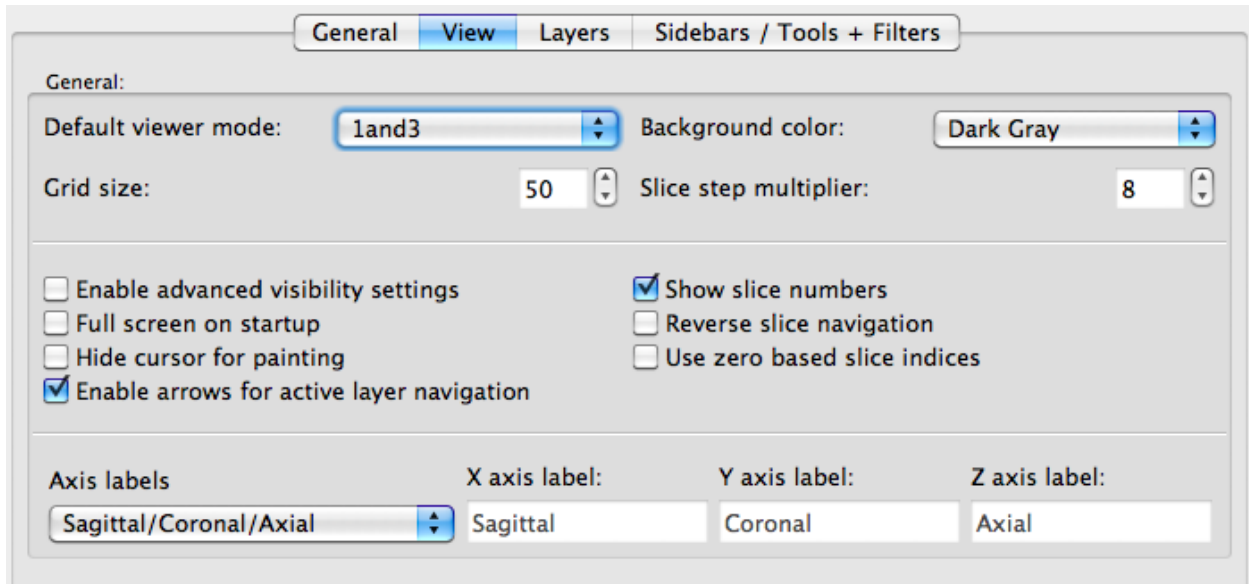


Fig. 2.7: The Seg3d preferences under the view tab.

Layers

Under this tab the user can change the appearance of the label masks. These options include: layer opacity, available colors, mask fill option (striped or solid), and border thickness.

Sidebars/Tools + Filters

Under this tab the user can select which menus appear on start up and which tools/filters will appear in the sidebars. This intended for the user to customize their user interface to improve the ease of use.

Seg3D Viewer

The Seg3D viewer is the main interface between the users and the software. This viewer is meant to be highly interactive and intuitive. Using a combination of mouse functions, keyboard shortcuts, and visual buttons the users can quickly visualize and segment data. This chapter will describe the functionality of several aspects of the Seg3D viewer including the overall layout of the interface, 2D slice viewer, 3D volume viewer, and the various viewing options available.

The default layout when Seg3D is opened and a new project is created is shown in the figure above. As seen from the image the interface for Seg3D consists of a number of viewers (both 2D and 3D), windows to control the functionality of Seg3D, and a tool bar at the bottom of the screen.

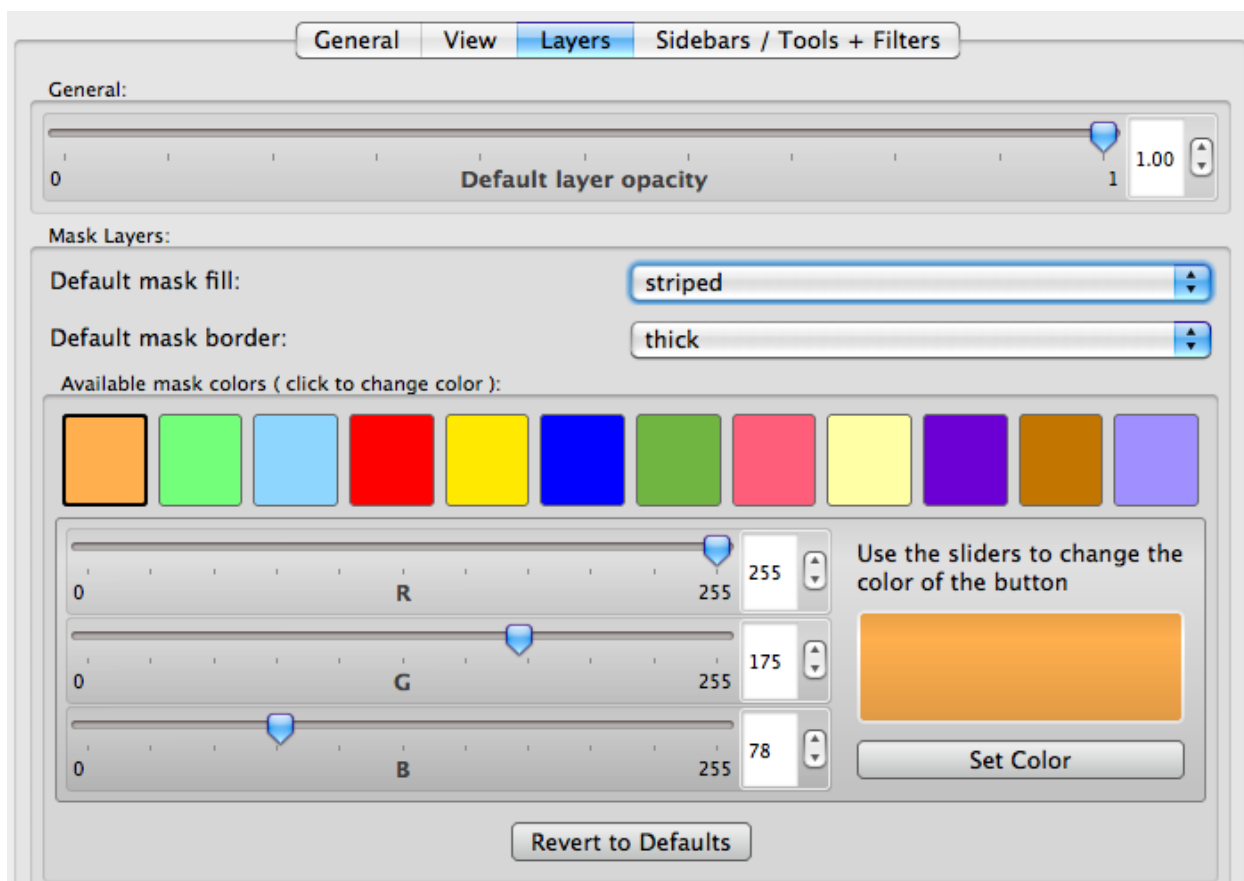


Fig. 2.8: The Seg3d preferences under the layers tab

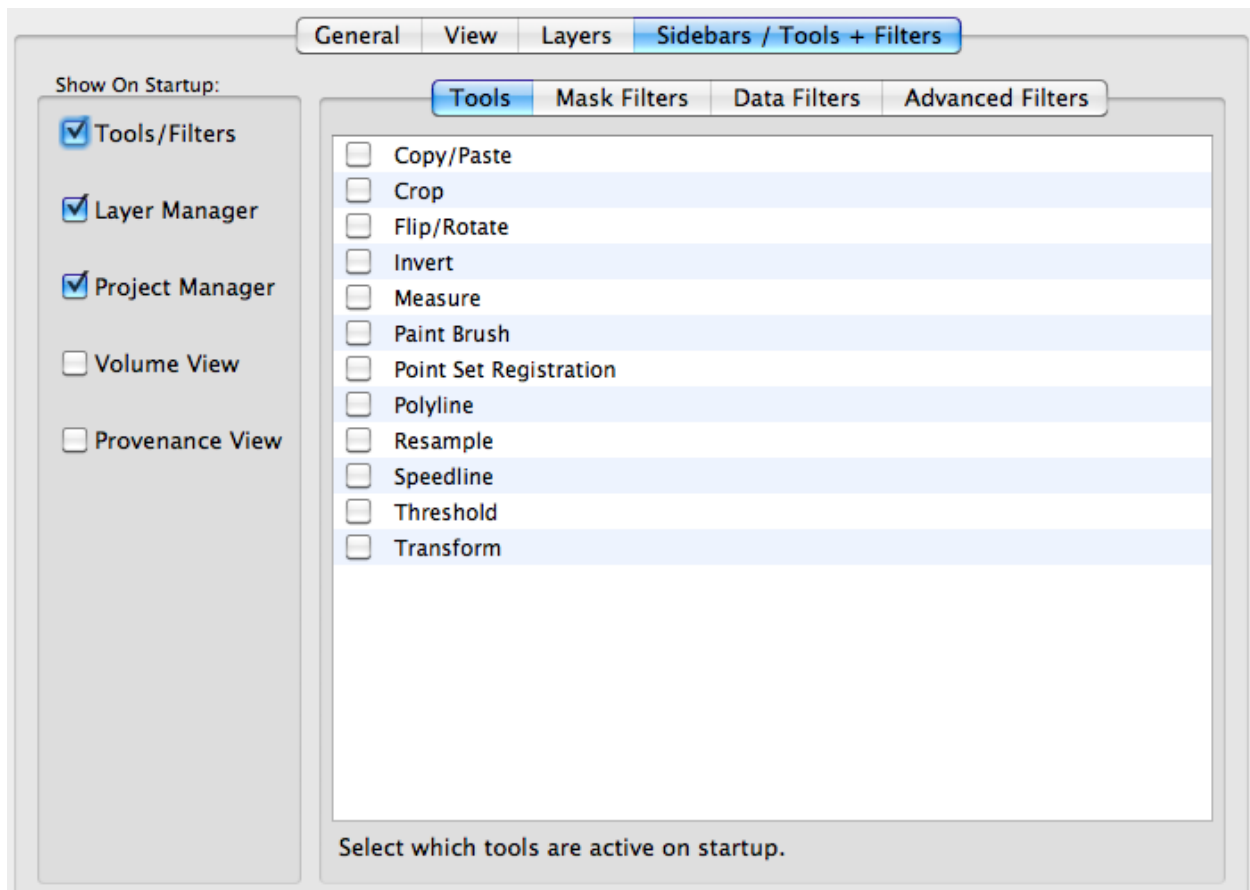


Fig. 2.9: The Seg3d preferences under the sidebars/tools + filters tab.

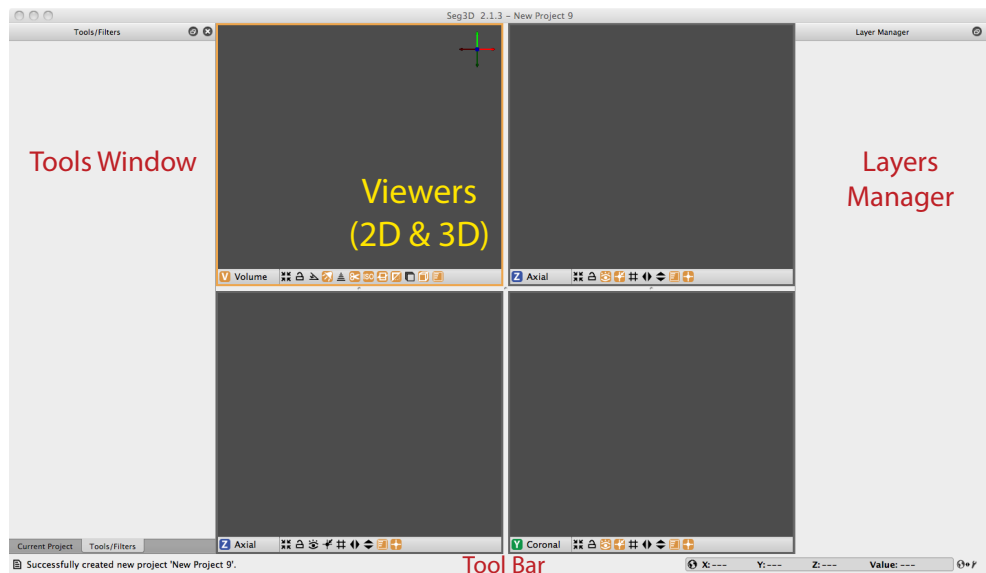


Fig. 2.10: Seg3D interface.

Viewer Panels

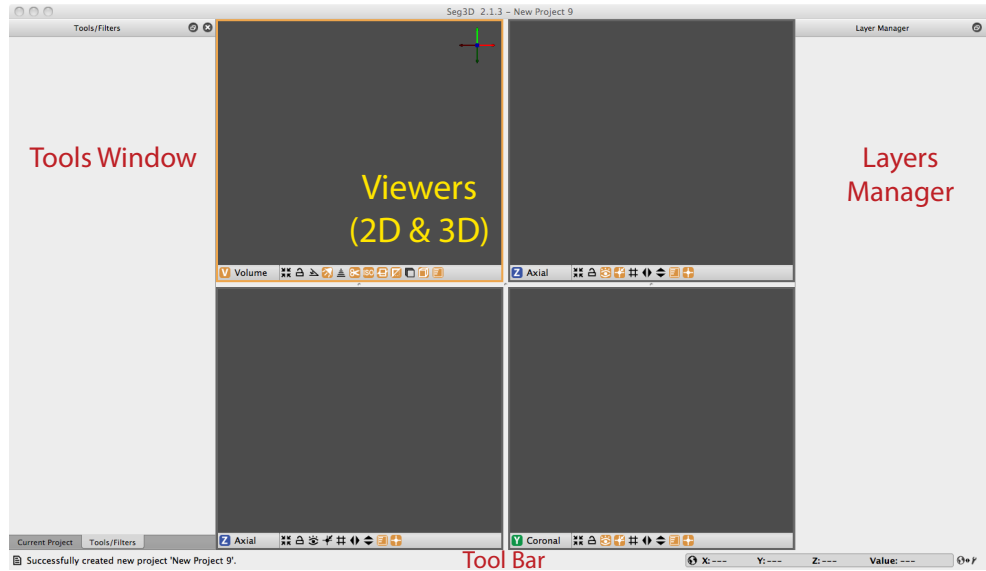


Fig. 2.11: Seg3D interface.

As seen in the screenshot above, there are a number of viewer panels in Seg3D, though the exact number can be set (see [Viewing Options](#)). Each of these viewer windows can be changed in size and in type. The type options are: volume, axial, sagittal, and coronal. The names of the 2D planes can be changed by the user in [Preferences](#). You can change the view of each pane by clicking on the view name and choosing the view to change it too. This is a drop down menu, so you can scroll over the name and the view will also change. There are also shortcuts to change views (V–Volume, X–Sagittal, Y–Coronal, and Z–Axial). There are several icons in each view panel, as well as mouse and keyboard shortcuts. The controls for each of the 2D viewers will be described below, as will the controls for the 3D volume viewer.

2D Slice Viewer

Tables 2.1 and 2.2 below show the mouse and keyboard actions which can be used to control the visualization and manipulation of image and segmentation data. Though these functions are general, there are tools which used these functions for specify purposes or may otherwise block a couple of these functions, the most prominent example is the scroll wheel in the paint brush tool is used to change brush size. In this case, you can still scroll through slices by holding Shift. In all cases, an alternative is given in the software. Also presented in this section is a list and description of the icons presented in the 2D slice viewer ([Table 2.3](#)).










Table 2.1: Mouse Functions in the 2D Viewers

Mouse Command	Function
left button drag	Modify brightness and contrast. Vertical is contrast, horizontal is brightness.
scroll up/down	Move up/down a slice. Note: using Shift maybe needed while using some tools (like paint brush)
CMD/CTRL+right button	Move slices in other planes to intersect at cursor. Viewers must have the picking icon enabled.
Shift+left button drag	Pan view
Shift+right button drag	Zoom view in/out

Table 2.2: Keyboard Actions in the 2D Viewers

Keyboard Action	Function
up arrow, >	Move up one slice
down arrow, <	Move down one slice
Shift+up arrow, Shift+>	Jump up n slices (set n in preferences)
Shift+down arrow, Shift+<	Jump down n slices (set n in preferences)
left/right arrow	Change active layer to previous/next layer
SPACE	Toggle layer visibility on/off (active layer)
G	Toggle grid visibility
P	Toggle picking state
T	Toggle overlay visibility
L	Toggle lock viewer (to other locked views of the same type)

Table 2.3: Icons and Actions in the 2D Viewers

Icon	Function
	Autoview Icon: This icon forces the panel to fit the objects in viewer with maximum size.
	Lock View Icon: This icon toggles the viewer lock on the panel (shortcut: L). Any changes to any locked viewers will change all the views. Viewers must be the same type and each must be locked to use this function.
	Visibility Icon: This icon toggles the visibility of the plane in the 3D volume viewer.
	Picking Icon: This icon toggles the ability of other planes to pick the slice to view in the panel. Only one viewer can have this option enabled. If there is only one type of plane, this cannot be disabled.
	Grid Icon: This icon toggles the visibility of the grid in the viewer
	Flip Horizontal Icon: This icon will horizontally flip the visualization of the slices in the viewer.
	Flip Vertical Icon: This icon will vertically flip the visualization of the slices in the viewer.
	Overlay Icon: This icon will toggle the visibility of the overlay on the viewer. This will allow unobstructed viewing of the slices.
	Picking Lines Icon: This icon toggles the visibility of the picking lines (shows other slices) in the viewing panel.

3D Volume Viewer

Though there is no segmentation that can be performed in the 3D volume viewer, it is a very useful function in Seg3D. It allows the user to see the 3D representation of the original and segmented data. There are many objects that can be viewed in the 3D viewer such as the 2D slices in 3D, isosurfaces of the segmented data, volume rendering of the image data, depth cues, and clipping planes. The purpose of this viewer is to allow you to view your data in as many ways as possible to facilitate segmentation.












Table 2.4: Mouse Functions in the 3D Volume Viewer

Mouse Command	Function
left button drag	Pan scene
middle button drag	Rotate scene
right button drag	Zoom in/out on scene



Table 2.5: Keyboard Actions in the 3D volume Viewer

Keyboard Action	Function
H	Toggle volume lighting
I	Toggle isosurface visibility
T	Toggle overlay visibility
L	Toggle lock viewer (to other locked views of the same type)

Table 2.6: Icons and Actions in the 3D Viewer

Icon	Function
	Autoview Icon: This icon forces the panel to fit the objects in viewer with maximum size.
	Lock View Icon: This icon toggles the viewer lock on the panel (shortcut: L). Any changes to any locked viewers will change all the views. Viewers must be the same type and each must be locked to use this function.
	Snap to Axis Icon: This icon will move the scene so that the viewing angle is aligned with the nearest axis. This will effectively “straighten” the scene.
	Lighting Icon: This icon toggles the lighting in the 3D viewer. With the lighting disable, the images will be shaded as if they were flat, i.e., no shading.
	Depth Cue Icon: This icon toggles the depth cue for the 3d viewer. This depth cue acts as fog, blending objects further from the camera into the background. Fog parameters can be changed in the volume view window (see Fog Panel).
	Clipping Icon: This icon toggles the viewing of the clipping planes in the 3D viewer. Clipping planes must be created before they can be seen in the volume viewer (see Clipping Planes Panel).
ISO	Isosurface Visibility Icon: This icon toggles visibility of the isosurfaces in the 3D viewer. This functions effects all the isosurfaces are declared visible in the layer manager.
	Slice Visibility Icon: This icon toggles visibility of the slices in the 3D viewer.
	Invisible Slice Visibility Icon: This icon toggles visibility of the invisible slices in the 3D viewer. Invisible slices are left when 2D viewers are destroyed.
	Volume Rendering Visibility Icon: This icon toggles visibility of the Volume Rendering in the 3D viewer. Volume Rendering must first be created in the Volume View window (see Volume Rendering Panel).
	Volume visibility Icon: This icon toggles visibility of the borders of the active layer in the 3D viewer.
	Overlay Icon: This icon will toggle the visibility of the overlay on the viewer. This will allow unobstructed viewing of the scene.




Controlling Windows

The windows on docked the side of the screen can be any of these five windows: *Project*, *Tools*, *Layer Manager*, *Volume Viewer*, and *Provenance* windows. Each of these windows can be closed by pressing on the close button  and opened by choosing the window from the **Window** menu or using hot key. When opened, the window will appear in the last location used, or the default if never opened. The windows can also be detached from the side (by pressing the detach button  and reattached to the side by moving the window close to the side of the Seg3D window, or by double clicking on the window header. Any detached window can be moved by clicking and dragging on the window header and resized by clicking and dragging on the corner of the window. Attached windows can also be resized by clicking and dragging on border between it and the viewing panels, but are more limited in the changes allowed.

Tool Bar

The tool bar located at the bottom of the Seg3D window contains some useful functions. The first is the message history icon on the left side of the tool bar (Table 2.7). For more information on this window, see *MessageHistoryWindow*. The other use function is on the right side of the tool bar, the information tool bar. This tool bar can switch between displaying information about the active layer at the mouse location and a quick menu for the active layers by clicking on the switch tool button (Table 2.7).

Table 2.7: Icons and Actions in the Tool Bar at the Bottom of Seg3D

Icon	Function
	Message History Icon: Opens Message History window.
	Switch Tool Icon: Switches between displaying location of mouse in the volume and the quick menu to switch between the active tools and layers.
	World Icon: Switches between coordinate system displayed in the information tool bar. The options are relative (indexed) and absolute (world).

The information toolbar will show information about the volume at the location indicated by the mouse. As shown in Fig. 2.12, the information given is the x, y, and z coordinates and the value of the layer at the mouse position in the selected volume. By clicking on the world icon (Table 2.7), you may toggle between indexed values (relative) and the world values (absolute, considers spacing). The data shown is changed when the mouse is moved, the active slice is changed, or the active layer is changed.

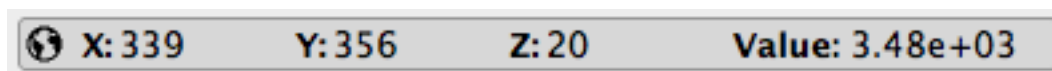


Fig. 2.12: Geometric Information shown in the information tool bar.

By clicking on the switch tool icon (Table 2.7), the tool bar will display a quick menu for the available tools and layers instead of geometric information. As seen in Table 2.7, this menu will display the active tool and layer. If the either is clicked, a dropped down menu will appear allowing the user to switch to an open tool or layer. This can be especially useful in full screen mode.

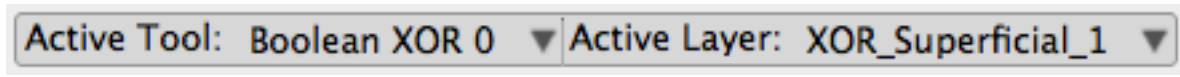


Fig. 2.13: Quick menu shown in the information tool bar.

Viewing Options

There are several options to change the layout and data viewing in Seg3D. As mentioned before, several windows can be moved and undocked to customize the layout. The viewing panels can also be altered so that there are virtually any number (up to six) and any size. Viewing panel sizes can be adjusted by clicking and dragging on the borders. This may change the sizes of several panels at once, but there are several different configurations available in the **View** menu at the top of the screen. The various options are explained below.

Full Screen

This is a unique and useful viewing option will cause Seg3D to use all of the real estate of whatever screen you are using. Full screen mode will cover everything else that may be open on your desktop, including the common menus on the top and bottom of the screen. The top tool bar will reappear if the mouse is held at the top of the screen momentarily, allowing the user to choose any tools or filters desired. Full screen mode is enable and disabled by clicking on the option in the menu, or by using the shortcut CTRL/CMD+F.

One Viewer

This and the following viewing options should be self evident. This option will show only one viewer. This is useful if you only care about one view at a time because it maximizes screen real estate. This one viewer can still be switch between the three slice orientations and the volume viewer.

One and One

This will display two viewers side by side.

One and Two

This will create three viewers, one bigger one on the left and two smaller ones stacked on the right.

One and Three

This is the default configuration, though it can be changed in the preferences. This generates four viewers, one large one on the left and three smaller ones stacked on the right. One and Three is a useful configuration because most people do more work in a single view, giving the user one large space to work, but use the other views as references only, so they do not take as much space.

Two and Two

This will generate four viewers of equal size, two on the left and right, which is useful when segmenting in multiple planes simultaneously.

Two and Three

This generates five viewers, two on the left and three on the right, for when four just doesn't cut it.

Three and Three

This generates six viewers, the maximum number, three on the left and right. This is generally used when multiple planes in the same direction are needed. These planes can be locked together as the user scrolls through them.

Table of Shortcuts

Keyboard Action	Function
CTRL/CMD+	Toggle Full Screen Mode
Alt+0	One viewer only
Alt+1	One and One
Alt+2	One and Two
Alt+3	One and Three
Alt+4	Two and Two
Alt+5	Two and Three
Alt+6	Three and Three

Seg3D Windows

In addition to the viewer windows discussed in the above chapter, there are several other windows involved in streamlining the user interface of the Seg3D software. Each of these windows can be accessed through the 'Window' drop-down menu. By default these windows appear in certain positions defined below, but each can be undocked from the Seg3D interface and either left as stand alone windows or repositioned elsewhere on the Seg3D application (see [Controlling Windows](#)).

Controller Window

The Controller Window is not opened by default when Seg3D is opened. To open the window, click the 'Window' option in the task bar and select the Controller option. The window may also be toggled by using the key command: Shift+CTRL/CMD+C. Notice the Controller Window option in the window drop-down menu is located below a separator. This indicates that the window will open a new pop-up window outside of the general Seg3D user interface.

The Controller window supplies the user with all of the data, variables, history, and logs of the current Seg3D session. There are four tabs that can be selected in this window:

Actions Tab

The Actions tab displays all of the actions that have been taken during the Seg3D session. Console entries can also be made with this feature.

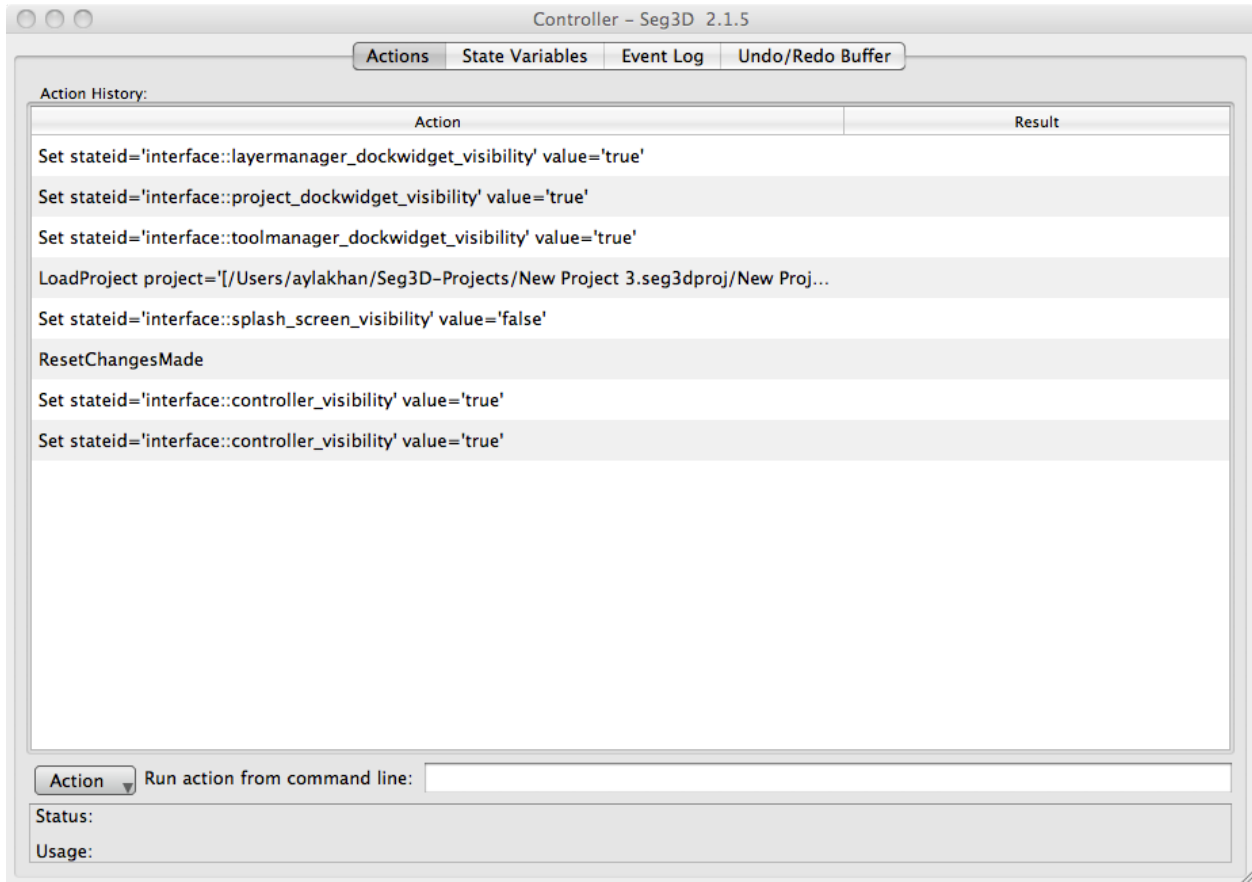


Fig. 2.14: Controller Window on initial open.

A series of actions area provided in a drop-down menu at the bottom of the window (Fig. 2.15). Once selected, the action will be placed into a command line to the right of the drop-down menu.

The user can then define the parameters for the action, or simply press enter to see what parameters are required to make the action valid. If additional parameters are needed to run the action, they will be displayed in a message in the box below the command line (Fig. 2.16).

Event Log Tab

The event log shows actions that have been taken on the session since the latest load of the session. If Seg3D is closed, and then reopened, the event log will only record the history for the newly opened session. This differs from the action tab in this way, which collects actions for the entire history of the session. The event log also differs from the action log in that it records all events involved with running Seg3D, not just the actions. This is also a non-editable tab.

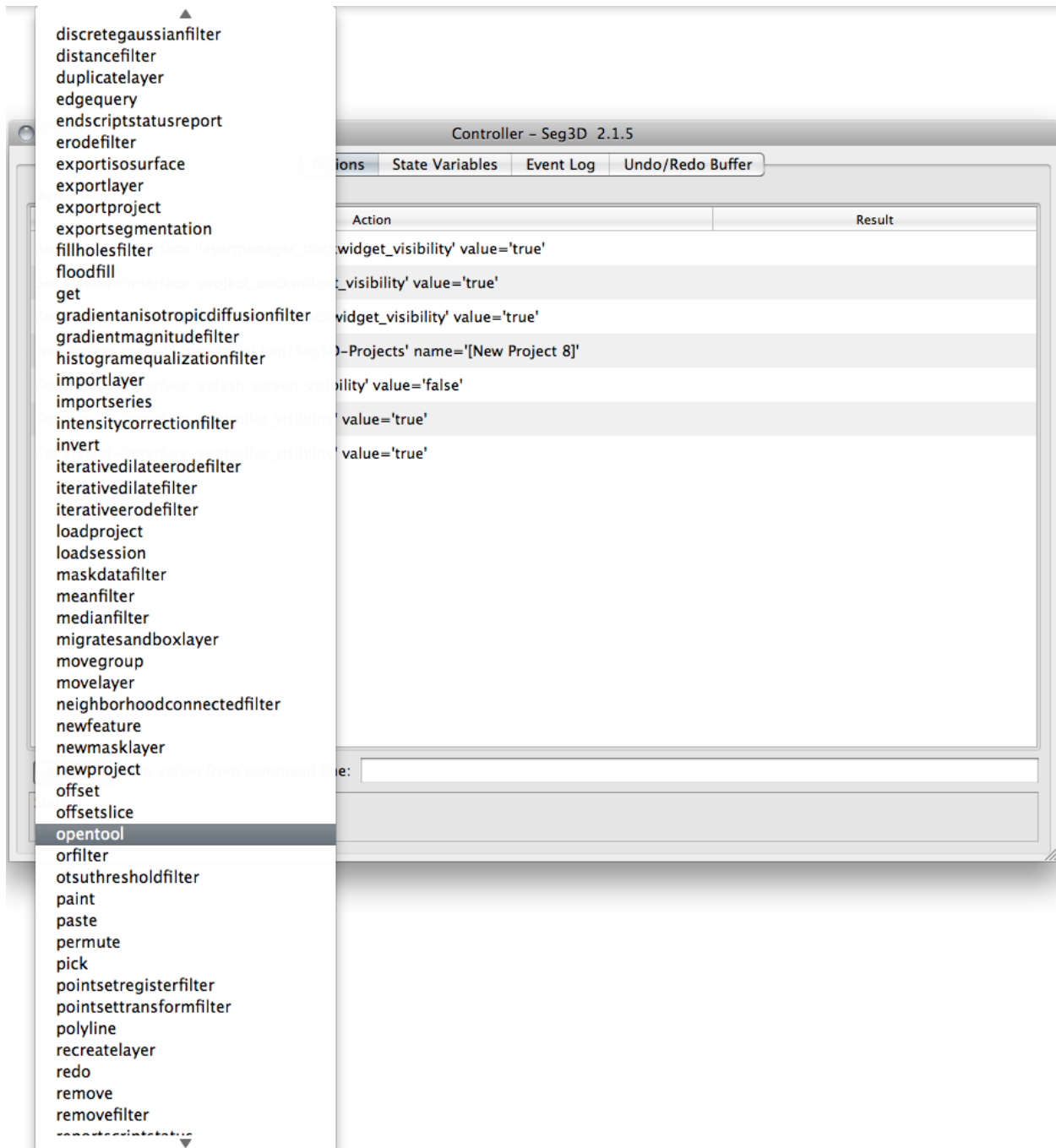


Fig. 2.15: Controller Window action list.

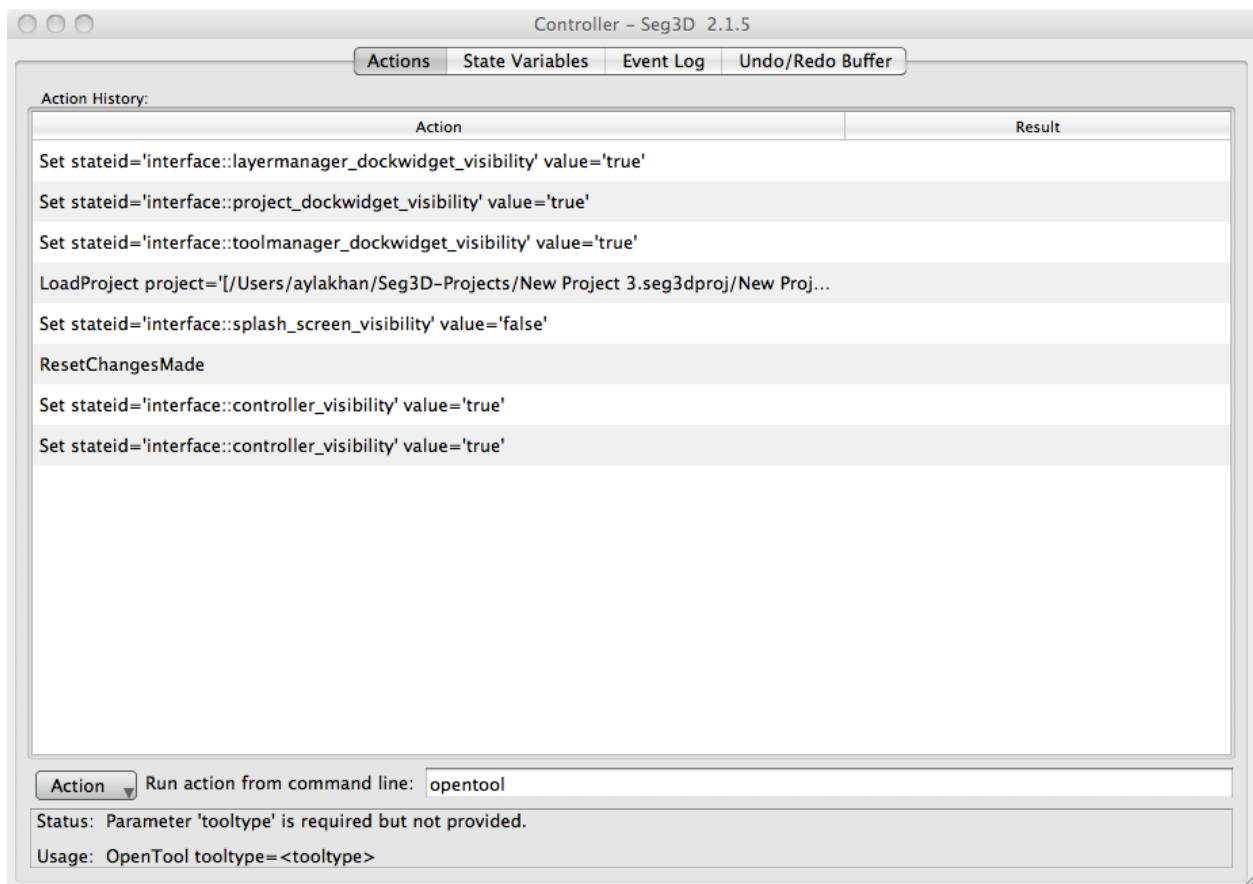
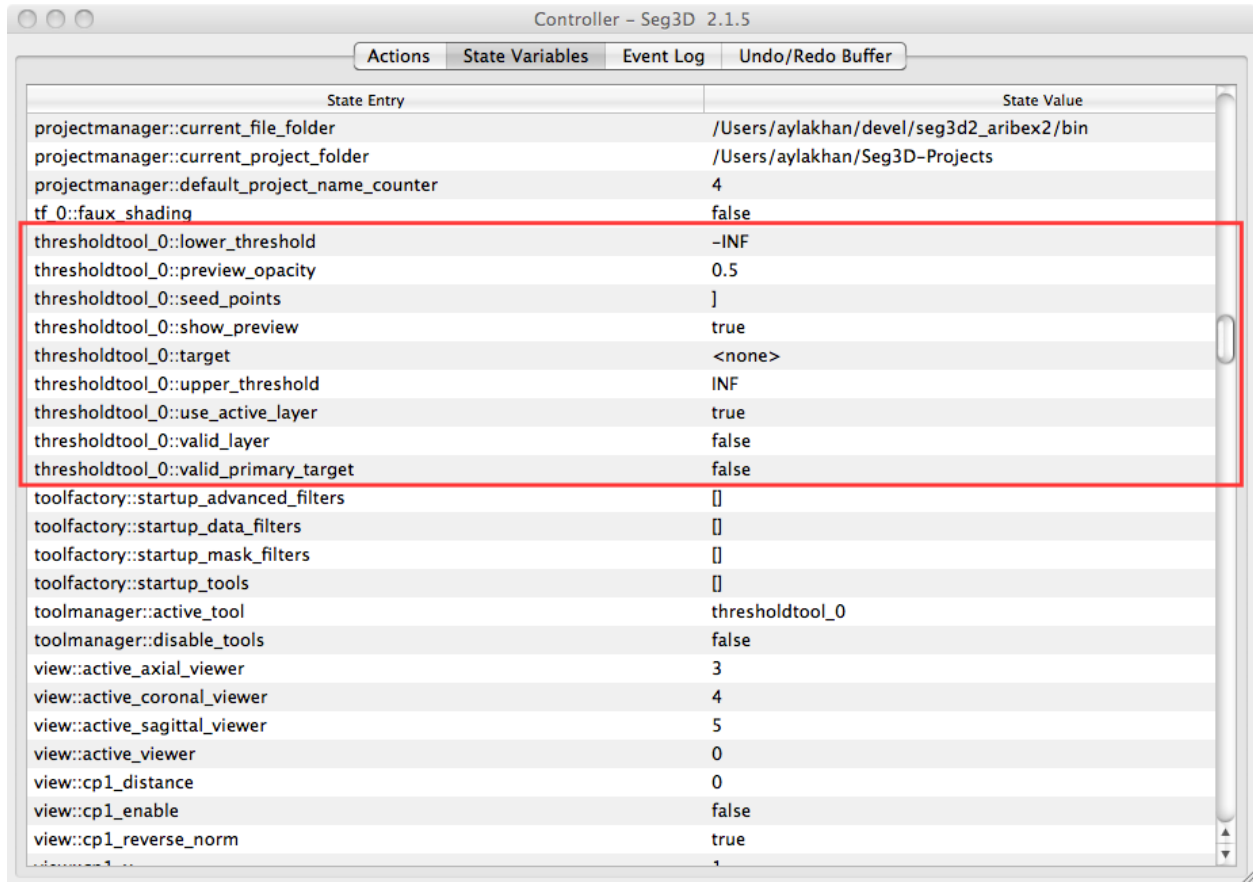


Fig. 2.16: Action usability message.

State Variables Tab

The state variables tab shows a list of all the state settings for the session. Settings cannot be edited in this window, instead use the built-in get and set functions. The state variables include Seg3D viewer settings, tool and filter arguments and settings. For example, Fig. 2.17 shows Threshold tool state variables.



State Entry	State Value
projectmanager::current_file_folder	/Users/aylakhn/devel/seg3d2_aribex2/bin
projectmanager::current_project_folder	/Users/aylakhn/Seg3D-Projects
projectmanager::default_project_name_counter	4
tf_0::faux_shading	false
thresholdtool_0::lower_threshold	-INF
thresholdtool_0::preview_opacity	0.5
thresholdtool_0::seed_points]
thresholdtool_0::show_preview	true
thresholdtool_0::target	<none>
thresholdtool_0::upper_threshold	INF
thresholdtool_0::use_active_layer	true
thresholdtool_0::valid_layer	false
thresholdtool_0::valid_primary_target	false
toolfactory::startup_advanced_filters	[]
toolfactory::startup_data_filters	[]
toolfactory::startup_mask_filters	[]
toolfactory::startup_tools	[]
toolmanager::active_tool	thresholdtool_0
toolmanager::disable_tools	false
view::active_axial_viewer	3
view::active_coronal_viewer	4
view::active_sagittal_viewer	5
view::active_viewer	0
view::cp1_distance	0
view::cp1_enable	false
view::cp1_reverse_norm	true
view::cp1_viewer	1

Fig. 2.17: Threshold tool state variables.

Undo/Redo Buffer

The Undo/Redo Buffer stores past actions for the current open project. Closing the session will erase the buffers. In these buffers a user can find the actions, in order, than can be either undone or redone by using Seg3D's Undo and Redo options.

Volume View Window

The Volume View Window is not opened by default when Seg3D is opened. To open the window, click the 'Window' option in the task bar and select the Volume View option. The window may also be toggled by using the key command: Shift+CTRL/CMD+V. Once activated, the window will appear on the right and side of the Seg3D panel covering the layer mask window. To toggle back to the layer mask window, click the Layer Manager tab at the bottom of the page. The Volume View Window has three separate options for volume displaying.

Clipping Planes Panel

The Clipping Planes Panel allows users to define multiple clipping planes with which to clip the volume. + and - signs at the top of the panel represent individual clipping planes. The + sign is an enabled clipping plan, while the - sign is disabled. The clipping panel option in the viewer window is activated by default.

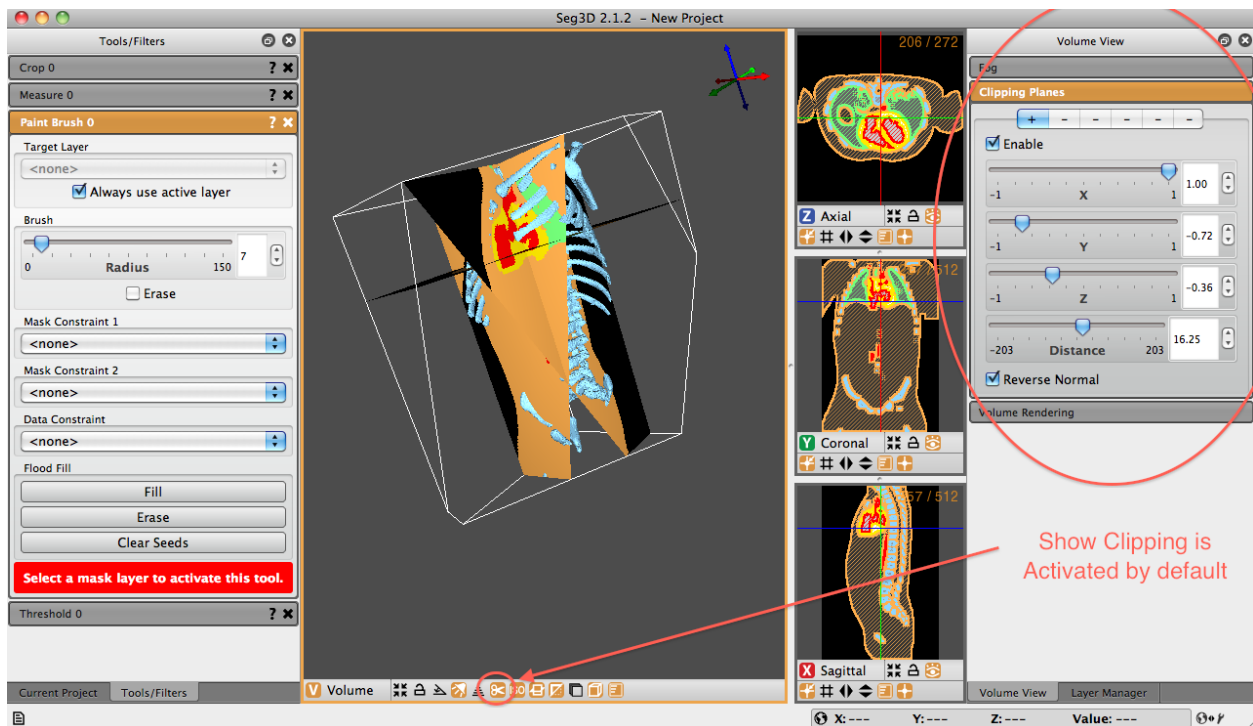


Fig. 2.18: Volume View Window - Clipping Panel Displayed. Show clipping option at the bottom of the viewer window is activated by default.

Once an clipping plane has been selected, it must be enabled. Click the 'Enable' checkbox. After the plane is enabled, the user is able to define the influence of each cardinal direction on the plane. The number 1 in a box indicates that the clipping plane will slice a 45-degree angle within the positive side of that cardinal direction on the 3d viewing window. A -1 will clip a -45-degree angle from the negative side.

In order to activate a clip from any cardinal direction, the user must click the slider associated with the desired plane. If the slider is not clicked, no clipping will take place...even with the clipping plane enabled. If, say, the z slider is not clicked, no clipping will occur in that plane. Combining these directional clips allows the user to define a plane that is tipped and tilted in any 3-dimensional direction. A fourth slider option, the Distance slider, allows the user to define how the position of the clipping plane with respect to the volume. If the distance slider is set all the way to the left, no image will be displayed (it will be completely clipped). If it is set to the right, the entire image will be displayed (no clipping is applied). Anything in between will show some clipping of the image.

The final option in the Clipping Planes panel is the option to 'Reverse Normal.' This option allows the user to reflect the clipping plane. That is, if the image is clipped within the positive XY plane, the 'Reverse Normal' option will display the clip within the negative XY plane Fig. 2.18 shows the result of a clipping plane applied in all three cardinal directions.

Fog Panel

The Fog Panel allows the user to control the fog density. In order to observe the effects of fog, the user must activate the Show fog icon on the bottom of the 3D viewer window. Once active, a fog will be observed on the image. The depth of the fog is determined by the distance that the 3-dimensional component is away from the viewer. In Fig. 2.19, the pelvis is tilted away from the viewer. It is therefore farther from the viewer and has more fog effect applied to it.

Fog density can be tuned by using the slider in the the Volume View Window. By default the density value is 1.00. If the fog density increases, the fogging effect increases.

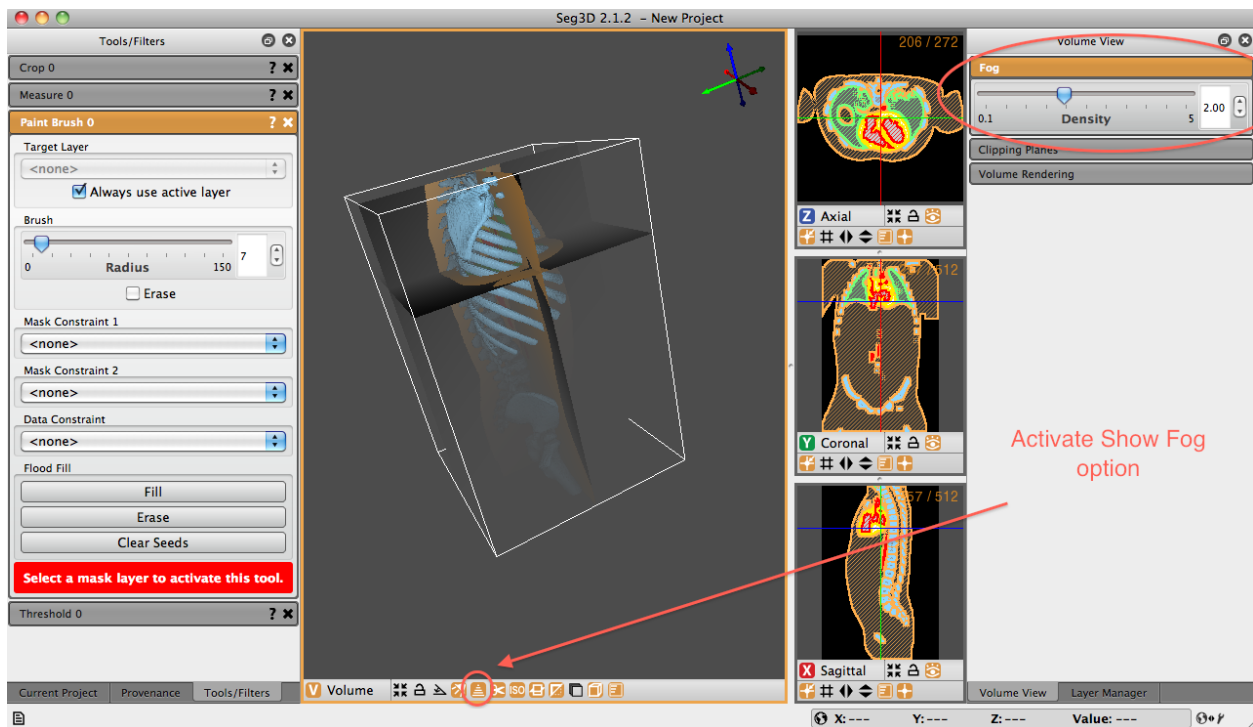


Fig. 2.19: Volume View Window - Fog Panel Displayed. The user must activate the Show fog option at the bottom of the viewer window.

Rendering Options

With these two options selected, the user now has three rendering options. These options are Simple, Faux Shading, and Ambient Occlusion.

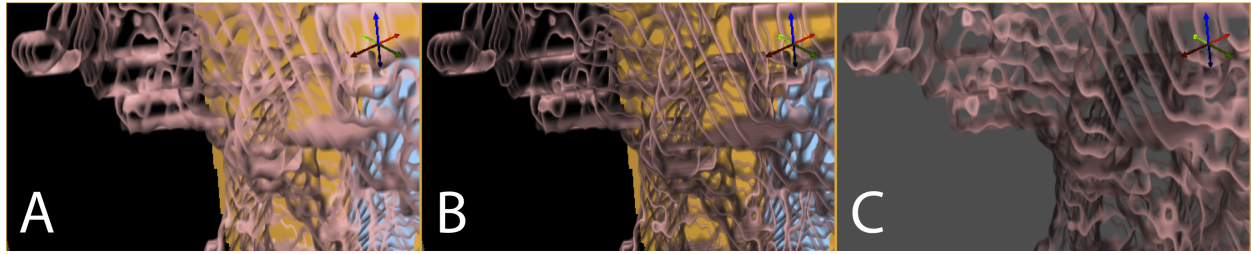


Fig. 2.20: Volume Rendering Options - A) Simple B) Faux Shading C) Ambient Occlusion

Simple

The simple rendering strategy gives the user the option to select the Sampling Rate. The higher the rate, the more accurate the representation of the selected region. An option in the 'Transfer Function' section of the pane that allows the user to choose 'Solid' transfer function representation changes the appearance by coloring each selected slice to a solid color. More on this feature will be addressed below. Note that the images in Fig. 2.20 do not have the solid option selected.

Faux Shading

Faux Shading is the same as the simple rendering option with the exception that a simple shading is added to the visible volume. Where the simple rendering fades regions of the slice to clear (where the 'Solid' transfer function option is NOT selected), the Faux Shading options fade slices to a grayer hue, dependent on where it sits from other 3-dimensionally viewed volumes. If the 'Solid' feature is selected, the Simple and Faux Shading rendering are the same.

Ambient Occlusion

Ambient Occlusion has several additional options. In addition to the sampling rate, an occlusion angle (range: 0 to 80 degrees) and a sample resolution (range: 1 to 10) are now available.

Transfer Function

Within the transfer function pane, a histogram exists that defines the volumetric image. The transfer function can be displayed on a Linear or Logarithmic scale. In the transfer function window, the user can choose to add or delete a feature. A feature can be dragged into positions defining regions of the histogram that the user wants to render. The default Feature is a line. By clicking and dragging the line, the feature can be moved. By clicking a point on the line, the individual point can be moved. Additional points can be added to a feature by clicking anywhere in the histogram panel that is not already defined by a line, that is, in the gray or black regions.

Each of the features observed can also be enabled/disabled and viewed as solid or graded. The solid feature (as addressed above) allows the feature to be represented as a graded slice or as a solid one. Solid slices simply apply the same color saturation to each section of the volume slice. De-selecting the solid option applies a gradient to the slices, with the outer edge being the brightest color.

Each point and feature can be repositioned, reshaped, and recolored in order to distinguish between defined regions. To change the color of a feature, select the desired feature and adjust the slider regions below the histogram in the Red/Green/Blue features. Colors can be used to easily distinguish different features in the volume. Additionally, each features Ambient and Specular lighting can be adjusted as well as volume shininess is adjustable

Volume Rendering Panel

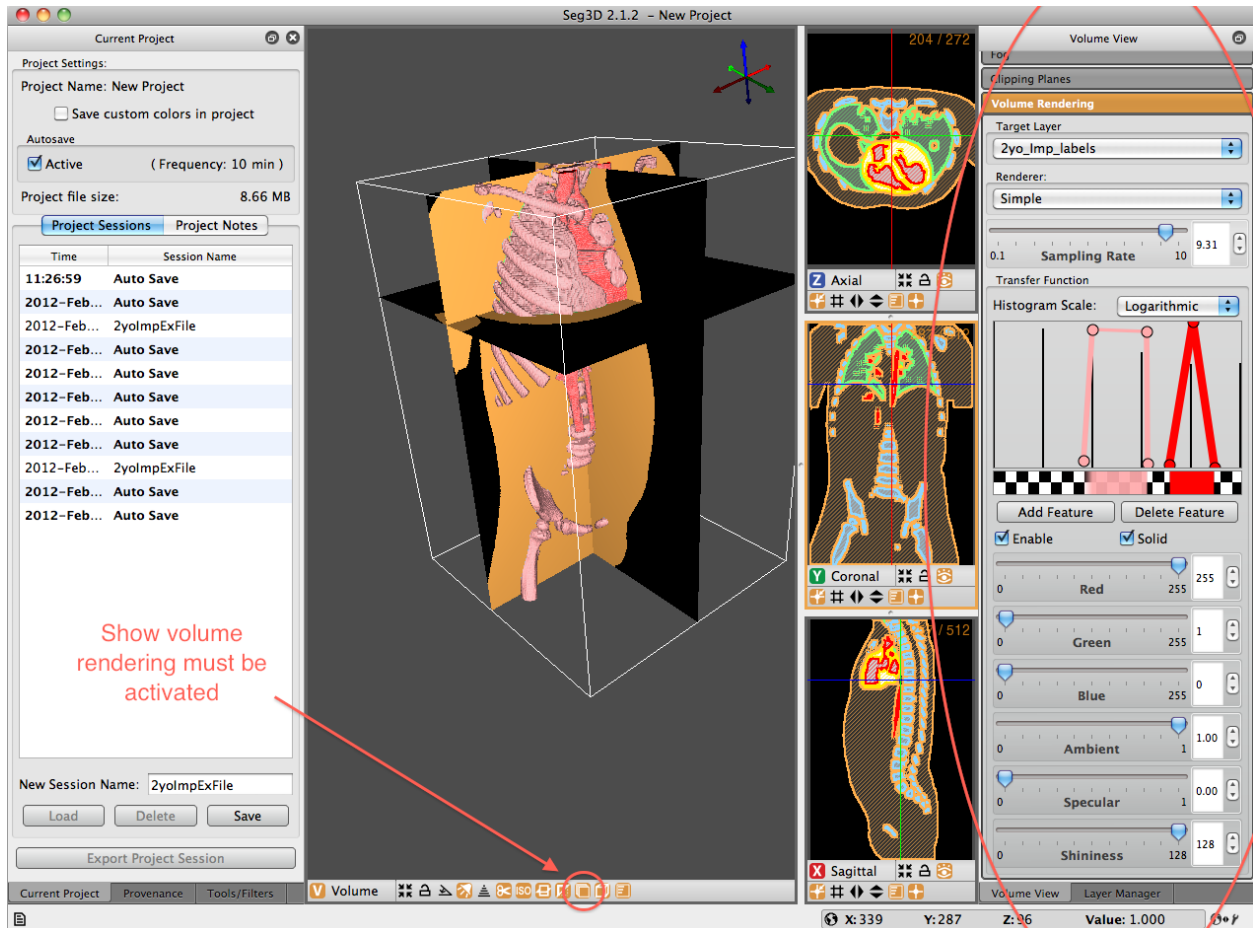


Fig. 2.21: Volume View Window - Volume Rendering Panel Displayed. Show volume rendering icon at the bottom of the viewer window must be activated.

The volume rendering panel allows users to generate a volume directly from the volume file, based on the application of transfer functions. These volume representations are created by extracting values from the transfer functions available and displaying them on a slice-by-slice basis. If the user zooms the image in closely, each individual slice can be seen (Fig. 2.20). The user must define the volume over which they would like to perform the volume rendering. The user must also make sure that the 'show volume rendering' option is selected at the bottom of the 3D viewer window.

Layer Manager Window

The layer manager window is the last of the three windows that open by default upon launching Seg3D. This window is positioned to the right side of the Seg3D window pane and contains all of the mask and volume files involved with the session. If a file is not selected when Seg3D launches, this window will be blank, otherwise it will contain the volume and mask surface files associated with the opened file.

The layer manager window is the last of the three windows that open by default upon launching Seg3D. This window is positioned to the right side of the Seg3D window pane and contains all of the mask and volume files involved with the session. If a file is not selected when Seg3D launches, this window will be blank, otherwise it will contain the volume and mask surface files associated with the opened file.

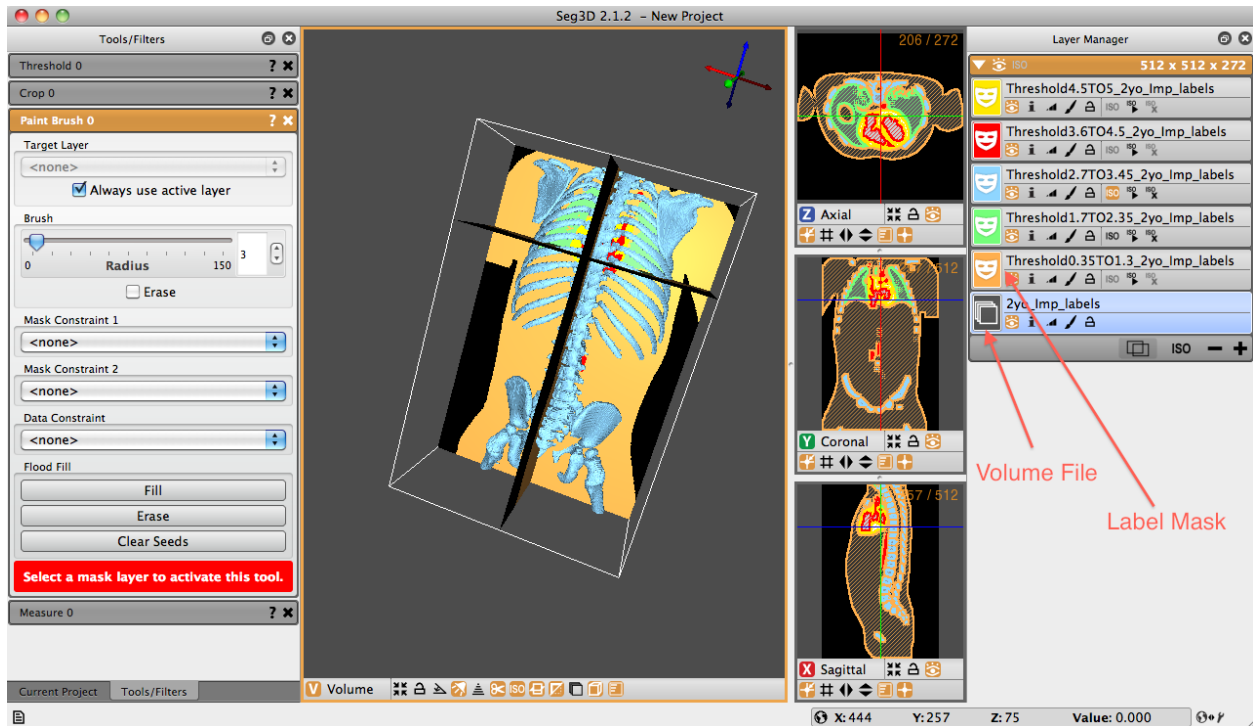









Fig. 2.22: Layer Manager Window



A volume file is represented by a gray image with multiple stacked planes. A label mask is represented by a colored icon with a white mask in the middle. The colors correspond to the label masks seen in the viewer windows. Names of label masks are, by default, a conglomeration of the tools applied to the original volume. For example, in Fig. 2.22, five label masks have been created from the volume file `2yo_imp_labels`. The uppermost mask (in yellow) has the name `Threshold4.5T05_2yo_imp_labels`. This name was generated because the threshold tool, with values between 4.5 and 5, was applied to the original volume file. If another tool were to be applied on the yellow mask, the new mask would state the name of the tool, followed by the complete name of the yellow label mask. Names of masks can be manually changed by clicking the current name and typing in the desired text.

Table 2.8: Icons and Actions Available for Each Layer

Icon	Function
	Visible Icon: The eye icon displays the current mask. When the eye is highlighted the mask is visible in the view plane, when it is not, the mask is 'turned off.'
	Info Icon: The 'i' icon displays information about the layer.
	Opacity Icon: The histogram-looking icon allows the user to change opacity levels of the layer.
	Appearance Icon: The paintbrush-like icon allows the user to change the layer's appearance.
	Lock Icon: The lock icon allows users to lock the layer to further editing.
ISO	Isosurface Visible Icon: This icon toggles visibility of the layers isosurface. Isosurfaces must be computed first.
	Isosurface Compute Icon: This icon generates the isosurface for the mask. It must be clicked before the IsosurfaceMenu or IsosurfaceDelete icons become active.
	IsosurfaceDelete Icon: This icon deletes active isosurfaces (if the mask has one).
	Shift+Left Button Drag Layer: This moves a layer from one group to another with resampling.




Each volume or mask label has standard, associated icons below their names. [Table 2.8](#) displays and describes each of these icons. These icons represent tools that are available for each individual layer.

Table 2.9: Icons and Actions Available at the Top of Each Layer Group

Icon	Function
	Expanded Layer Group Icon: This icon indicates that the layer group is visible, showing all the layers in it. The layer group can be collapsed, and expanded again by clicking on this icon. This turns into a right arrow when collapsed.
	Visible Icon: This icon will toggle the visibility of all the layers in the group. If some of the layers are not visible, this icon will make those layers visible so that all the layers are visible. If all layers are visible, this will turn off visibility.
ISO	Isosurface Visible Icon: This icon toggles visibility of all computed isosurfaces in the layer group. If there are computed isosurfaces that are not visible, this icon will make those visible. If all computed Isosurfaces are visible, this will hide them all.

Layers in Seg3D are arranged into layer groups. Layer groups are formed with layers that have the same geometric information, that is the same origin, spacing, and size. Groups are separated by panels with an orange header. Generally speaking, most tools and filters requiring more than one input can only operate on layers in the same group (and therefore the same grid geometry).

Table 2.10: Icons and Actions Available at the Bottom of Each Layer Group

Icon	Function
	Duplicate Layer Icon: This icon allows the user to duplicate one or more layers. Once this icon is clicked, the user will be prompted to choose the layers to duplicate by checking the boxes next to the layers. The duplicated layers will have the same name with “copy” appended to the front.
ISO	Isosurface Menu Icon: This icon will open a menu that allows the user to change some of the isosurface rendering parameters for all the isosurfaces in the layer group. These parameters are the quality and capping of the isosurface.
	Delete Layer Icon: This icon allows the user to delete one or more layer from the layer group. When the icon is clicked, the user will be prompted to choose the layers to delete by clicking the box next to the layer. There will be a confirmation window after the selection is made.
	New Layer Icon: This icon will create a new mask layer that is the same size as those in the layer group but will be empty.

There are some functions that are operated as a group. These are indicated by the icons on the top of the pane, in the orange bar (Shown in [Table 2.9](#)). Additionally, there are some other icons at the bottom of the group that control some of the group functions ([Table 2.10](#)). Hovering the cursor over the icon will display the the use of each additional icon.

Message History Window

The Message History Window is not opened by default when Seg3D is opened. This window is opened by clicking on the message history icon in the bottom left of the screen in the *tool bar* ([Table 2.7](#)). This window will show the messages that Seg3D displays when many functions are performed. This can be used to track your steps at a high level.

Project Window

The project window is one of three windows that is opened by default when Seg3D launches. It is located on the left side of the panel and is secondary to the default 'Tools Window' which is also located on the left but appears over top of the project window. In order to access the window, simply click the tab at the bottom of the left side named 'Current Project' (see [Fig. 2.23](#)) or go to the 'Window' drop down menu, de-select 'Project Window' and reselect it.

Now you come to the 'Current Project' window. The window is divided into two panes. At the top of the window is the Project Settings pane. The name of the current project is listed. Below this is the option to 'Save custom colors in project.' This option allows the user to keep the defined colors assigned to specific label masks within the project. Below this is an 'Autosave' option. Seg3D autosaves sessions every 10 minutes by default. The default time between autosaves can be changed in the general Seg3D preferences.

The second pane holds the 'Project Sessions' and 'Project Notes' panels. The project session contains information on the saving of various sessions throughout the work-time in Seg3D - including autosaves. These session saves can be selected and loaded at any time. Loading a session save does not delete saves that came later. To delete, the user must explicitly select the session and push the delete button at the bottom of the pane. Additional sessions may also be saved with user specified names. The second panel of this pane (the 'Project Notes' panel) allows the user to write notes to along the way.

At any time the user can export the session to a new location - with or without a different name than the original session by clicking the 'Export the Project Session button on the bottom of the 'Current Project' window.

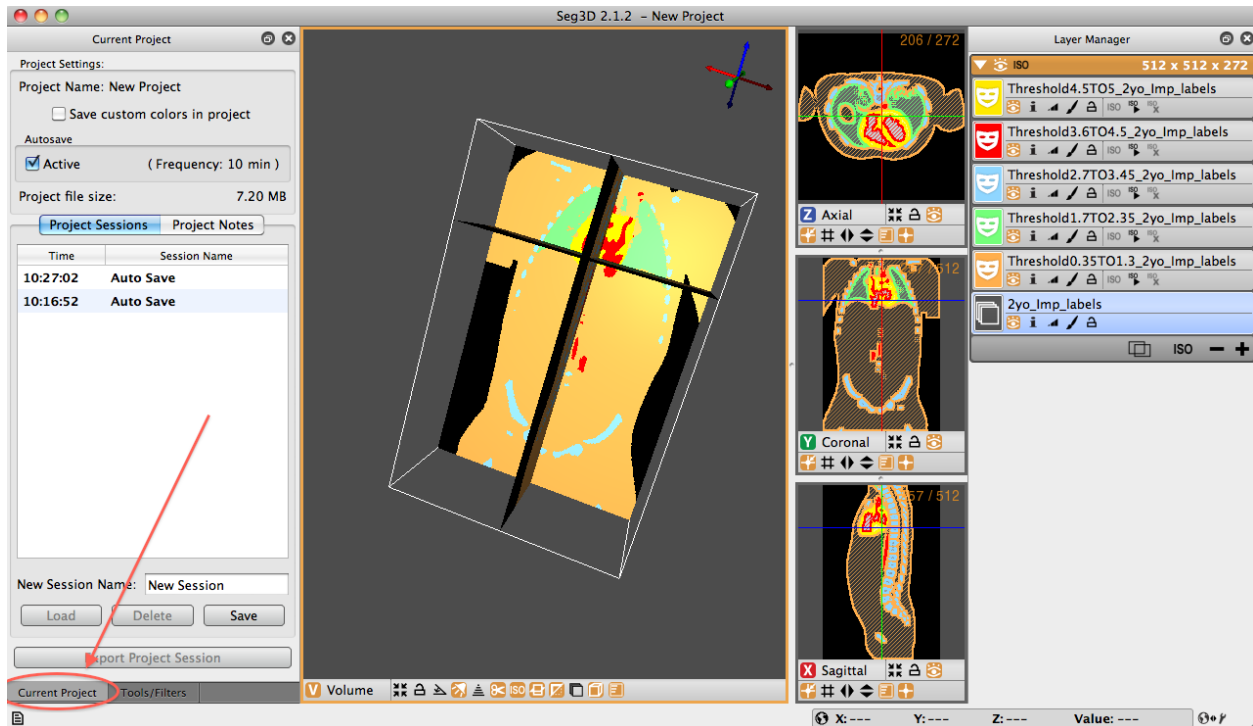


Fig. 2.23: Project Window

Provenance Window

The Provenance Window is not opened by default when Seg3D is opened. To open the window, click the 'Window' option in the task bar and select the Provenance option. The window may also be toggled by using the key command: Shift+CTRL/CMD+H. When the window opens, it will appear on the left side of the Seg3D user interface over top of the 'Current Project' panel and the 'Tools' panel.

The Provenance Window shows the history of the active layer. In order to see which actions lead to the current layer, highlight the layer in the 'Layer Window' on the left of the Seg3D display. Initially, nothing will appear in the Provenance window. Click the Refresh button at the bottom of the panel and the complete history of that mask will appear.

For example, Fig. 2.24 shows an example of a 'FillHoles' Layer. This layer was generated by applying a FillHoles tool to the thresholded mask of the lungs. As can be seen in the Provenance panel, the first step in creating this mask was to Import the volume layer. Next, a threshold was taken. Last, the FillHolesFilter was applied to the Threshold mask. By selecting one of the steps in the Provenance window, the parameters defining that portion of the mask history will be shown. For example, by selecting the Threshold option, the thresholding range will be displayed.

Python Console

The Python Console window is not opened by default when Seg3D is opened. To open the window, click the 'Window' option in the task bar and select the Python option. The window may also be toggled by using the key command: Shift+CTRL/CMD+Y. Notice the Python Window option in the window drop-down menu is located below a separator. This indicates that the window will open a new pop-up window outside of the general Seg3D user interface.

The Python Window opens a Python scripting console that can be used by advanced users to control Seg3D (Fig. 2.25). Python code wrappers are generated allowing tools, filters, the viewer and data to be controlled in Python scripts. Basic Seg3D Python code samples are available under [Python Code Snippets](#). An extensive treatment of python coding will not be attempted in this document; for more information on python coding, consult other resources.

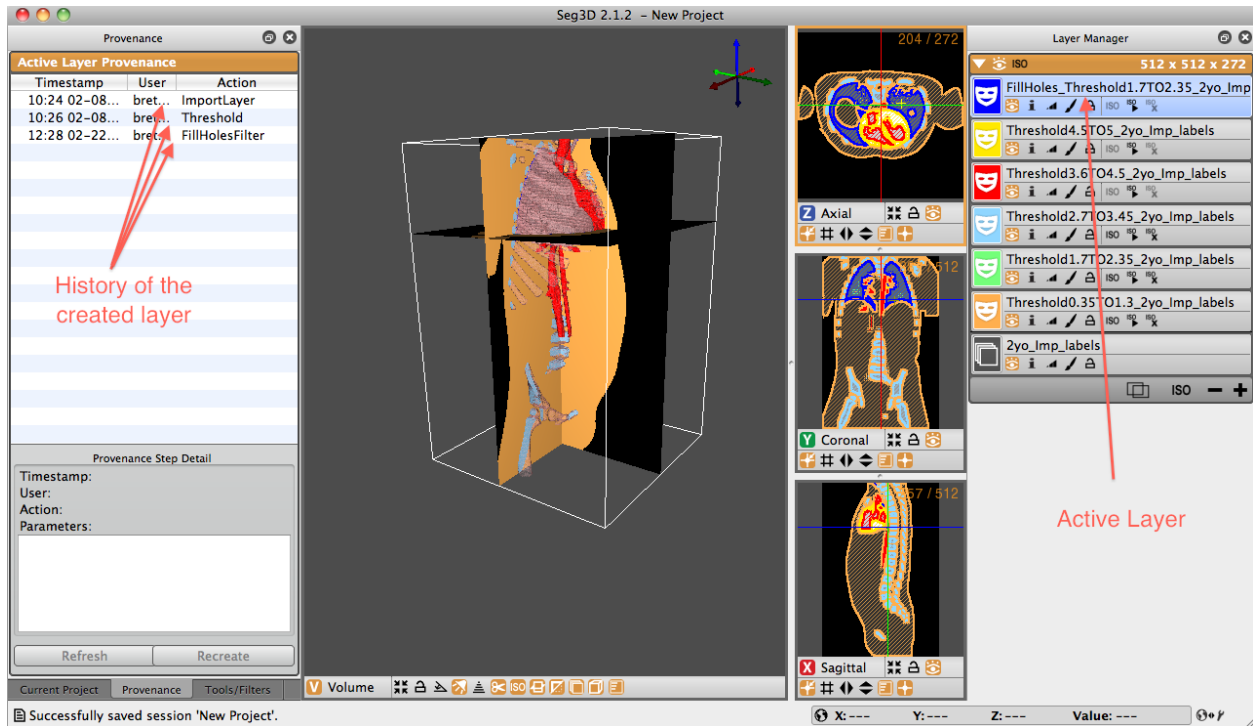


Fig. 2.24: Volume View Window - Clipping Panel Displayed. Show clipping option at the bottom of the viewer window is activated by default.

Tools Window

The tools window is another of the three windows that is opened upon Seg3D launch. This window is also on the left of the Seg3D panel and is, unlike the project window, is displayed when Seg3D is opened. Users can toggle between project and tool windows by selecting the specific tabs at the bottom of the left most window pane.

The tool window houses the current tools that the user has selected from the 'Tools' drop down menu. The active tool will be highlighted in orange and will be displayed. Inactive tools, that have been opened during the session, will be grayed out and minimized. To access one of the other tools, click one of the grayed out items or select it from the 'Tools' drop down menu. In Fig. 2.26, the 'Paint Brush' tool is active, while the 'Threshold,' 'Crop,' and 'Measure' tools are inactive.

Welcome Screen

Anytime you start Seg3D, users are first presented with the Seg3D welcome screen. As seen below, the welcome screen consists of the Seg3D splash screen and a menu displaying the available options for starting and continuing a segmentation in Seg3D. The included options are to load a recent project, open existing project, start a new project, quickly open a file for viewing, and to quit Seg3D. These options are the same as some of the options in the **File** menu which are explained in this chapter as well as the [File](#) page.

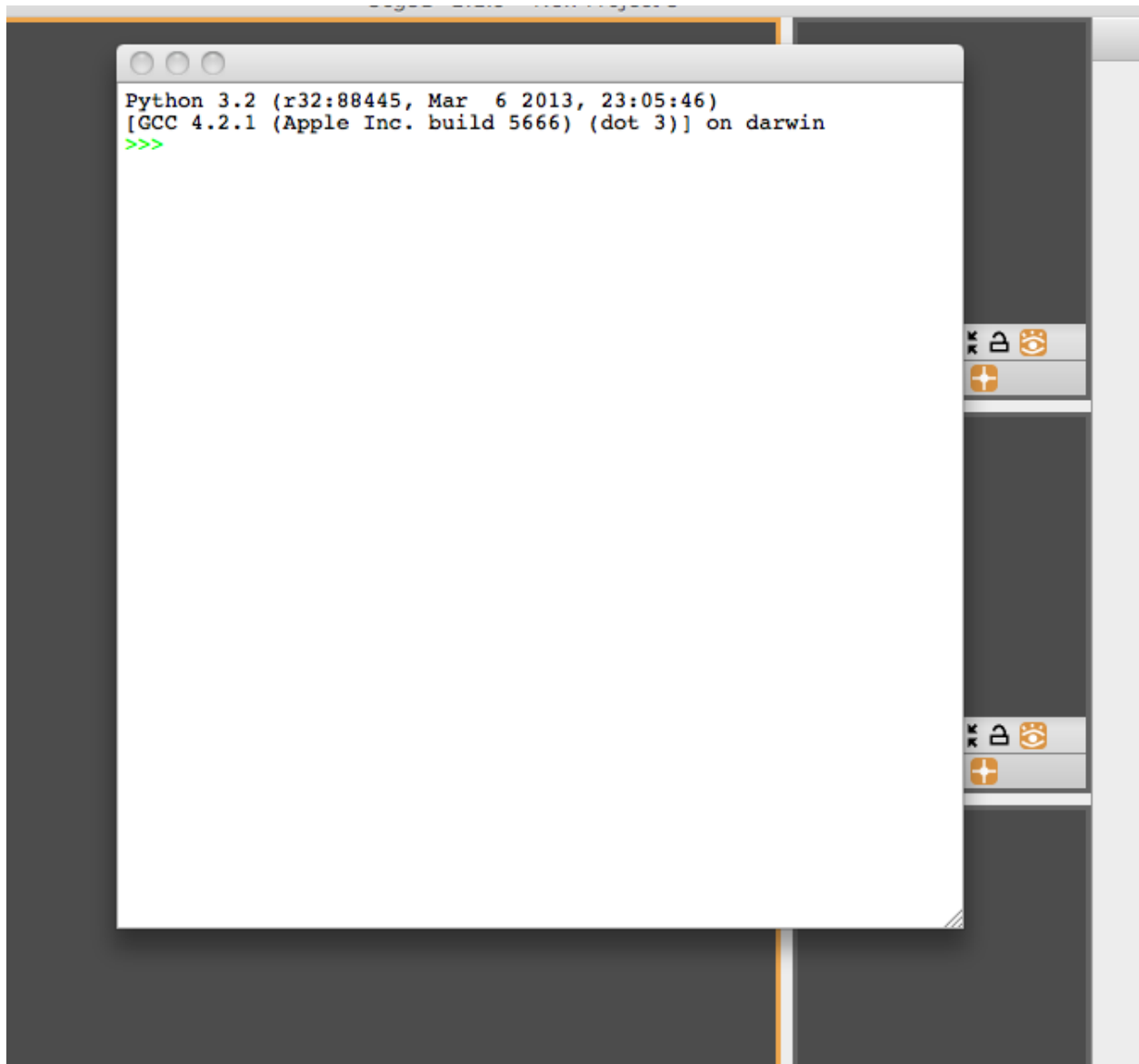


Fig. 2.25: Python Console Window

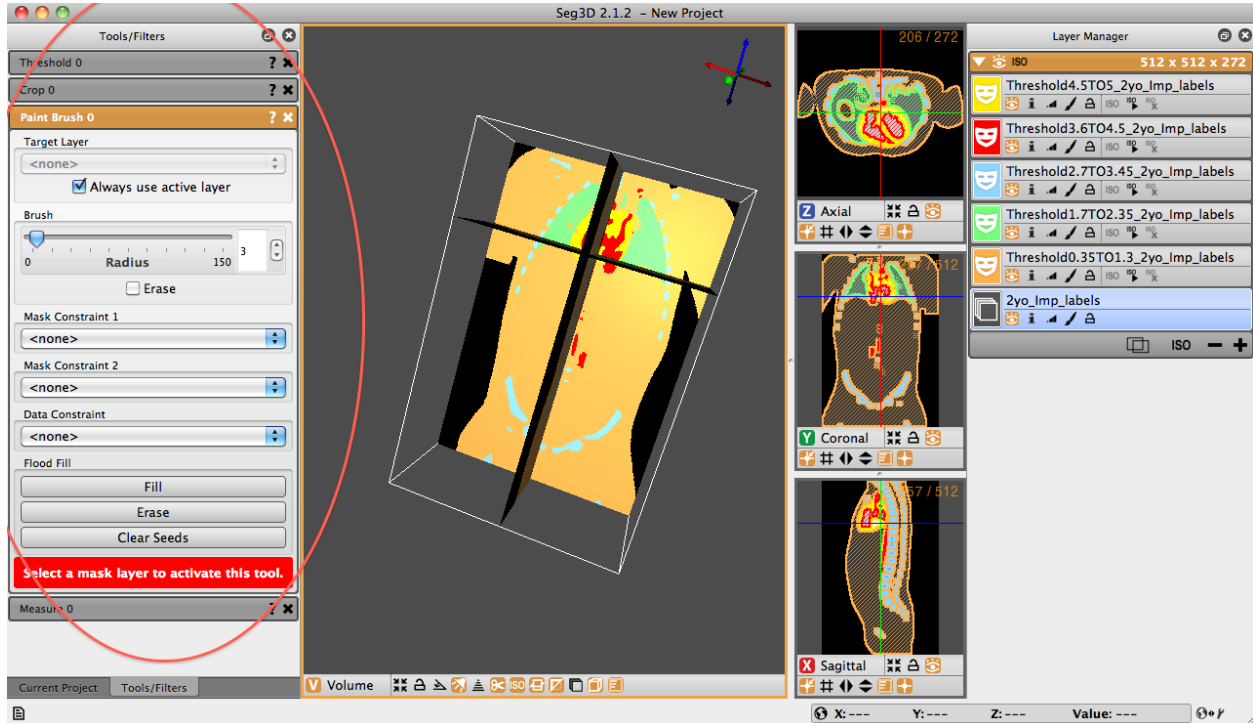


Fig. 2.26: Tools Window



Fig. 2.27: Seg3D welcome screen.

Load Recent Project

This option allows the users to quickly open a project recently opened in Seg3D. The list of recent projects is shown in the text box above the **Load Recent Project** button. This option will be unavailable unless one of the recent projects in the list is selected. Updating or reinstalling Seg3D may erase the saved list of recent projects even though the projects still exist.

Open Existing Project

This option is the same as the **File** *Open Project* menu item. It will open a dialogue window to allow the user to choose an existing *.s3d or *.seg3dproj file to load into Seg3D.

Quick Open File

This option is the same as the **File** *Import Layer From Single File...* menu item. You may choose a file to load into Seg3D for viewing. All of the filters and tools will be available, but the project can not be saved automatically until a project name is created by using the **File** *Save Project As* menu item. This option does not allow the opening of image stacks.

Quit

This option will quit Seg3D. This exists for the rare occasion that you accidentally open Seg3D.

Start New Project

This option is the same as the **File** *New Project* menu item. It will display the New Project Wizard (Fig. 2.2), allowing the user to choose a project name and location so that the project can be saved easily and automatically.

2.2 Seg3D Tools

There are many different tools and filters implemented in Seg3D. These tools and filters are there to attempt several tasks, from reformatting original data, smoothing and de-noising image data, to automatic and manual segmenting. Every tool and filter will be describe briefly in an effort to provide the user with a list of methods that may be used to segment the data that they have. These methods have been separated into four categories, general tools, mask filters, data filters, and advanced filters.

Most of the tools and filters have a 'replace' option. The default for most of these functions is to create a new layer, but 'replace' will cause the new layer to instead replace the old one. This is very useful in avoiding the generation of multiple layer that are only intermediate steps. Also, most tools and filters will by default insert the active layer as the primary input. You may disable this and choose manually which layer to modify.

Many of the tools and filters must use a certain type of data as the input, either mask or image data. If the wrong type of layer is chosen as the input, the tool/filter interface will display a warning message in a red box near the execution button informing you of this. The tool will also be disabled until the correct type of layer is chosen. This second feature applies to empty inputs also.

A final note is that when tools have multiple inputs, the inputs must all be from the same layer group (geometric grid), with exception of the registration tool. This ensures that all the voxels in the multiple inputs have corresponding voxels to every other input volume.

2.2.1 Advanced Filters

Canny Edge

This filter will output the pixels that are edges in a mask layer.

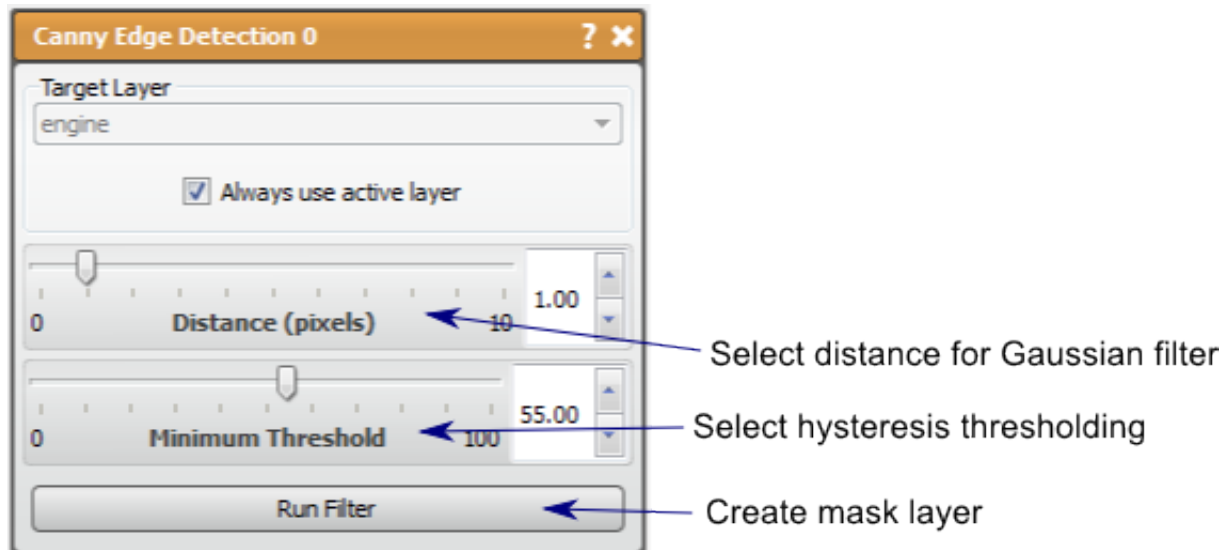
Detailed Description

The Canny Edge filter computes the edges in a data layer using a Canny Edge Detection Filter [Can86]. This filter will output the pixels that are edges in a mask layer.

This filter is composed of four steps:

1. Smooth the input image with Gaussian filter.
2. calculate the second directional derivatives of the smoothed image.
3. Non-Maximum Suppression: the zero-crossings of the 2nd derivative are found, and the sign of the third derivative is used to find the correct extrema.
4. The hysteresis thresholding is applied to the Gradient Magnitude (multiplied with zero-crossings) of the smoothed image to find the link edges.

There are two parameters for this filter, *distance* and *minimum threshold*. The *distance* parameter is the controls the Gaussian blurring in the first step and is the same as in the Gaussian blur filter. Higher distance will result in fewer edges detected and the edges that are detected will likely be thicker. This is because these edges are being blurred out. The *minimum threshold* parameter controls the hysteresis thresholding in the fourth step. This parameter will set the level of that will count as an edge or not. This means that a higher minimum threshold will have fewer edges in the result and the remaining edges while more likely be true edges.



Confidence Connected

This filter will produce a mask of pixels statistically similar to pixels surrounding defined seed points.

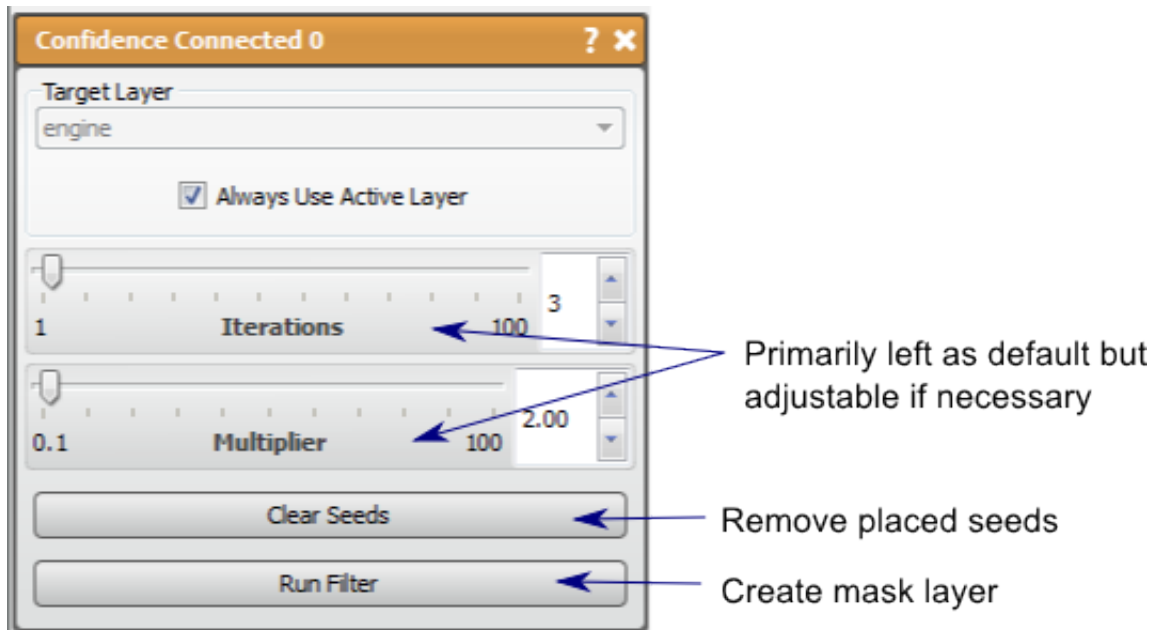
Detailed Description

The Confidence Connected filter is similar to the Neighborhood Connected filter in that it will find connected pixels in a data volume that are statistically similar to pixels surrounding the user defined seed points and output them as a mask layer. The seed points set in the 2D viewer (**left mouse** to place, **right mouse** to remove).

The pixels neighboring the provided seeds are used to calculate the initial statistics, the variance and mean, then the connected pixels that fit the threshold region defined by the *Multiplier* parameter (number of standard deviations from the mean) are selected. This process is repeated by recalculating the statistics of the selected region and recomputing the connected region based on these new values. Therefore, the filter can be repeated many times.

The two parameters in this filter are *Iterations* and *Multiplier*. The *Iterations* parameter describe the number of times the statistics and connected region are calculated. Small increases in the number of iterations may not significantly alter the results. The *Multiplier* parameter describes the threshold of the pixel intensities as described in the previous paragraph. The higher the multiplier the more pixels will be selected by the filter. A small increase in this parameter can very significantly alter the results of the filter.

Note: The default values (Iterations=3 and Multiplier=2) should be the best values for most sources of image data.



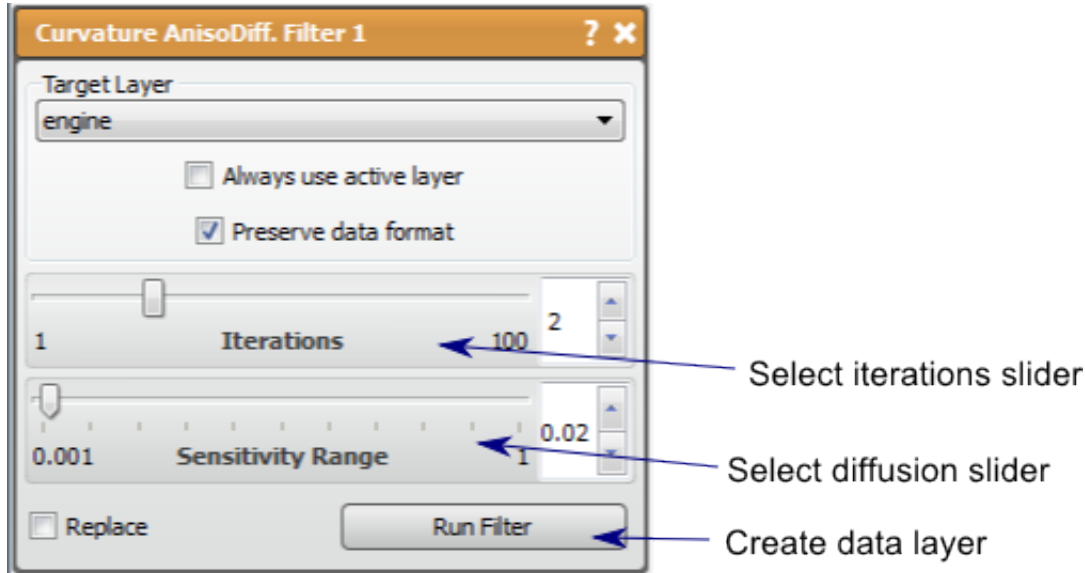
Curvature Anisotropic Diffusion

This filter performs anisotropic diffusion using the modified curvature diffusion equation (MCDE).

Detailed Description

The Curvature Anisotropic Diffusion performs anisotropic diffusion on a data layer using the modified curvature diffusion equation (MCDE). Anisotropic diffusion filtering is used to reduce noise (or unwanted detail) in images while preserving specific image features, such as edges and is based on the work of Pietro Perona and Jalhendra Malik, [PM90]. The output of this filter is a smoothed data layer.

The parameters for this filter are *Iterations* and *Sensitivity Range*. The *iterations* parameter will control how many iterations of the filter will be executed. The more iterations, the smoother the image. The *sensitivity range* parameter controls the ‘diffusion’ of the objects in the image, therefore the higher the sensitivity, the more ‘diffusion’ and thus more smoothing.

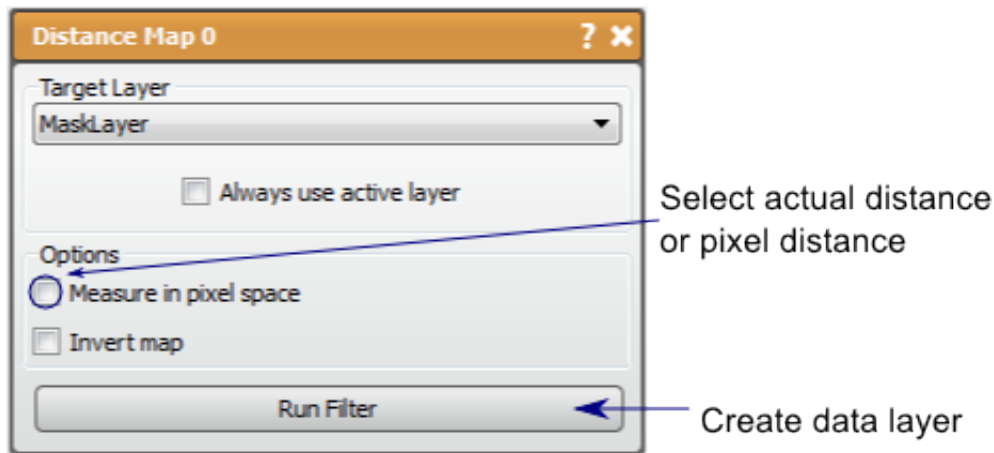


Distance Map

This filter will calculate the distance to the surface of the mask from within the volume.

Detailed Description

The Distance Map filter will calculate the distance to the surface of a mask for every pixel in the volume. This filter will output a distance map as a data layer with the values considered inside the mask marked as negative. This filter is useful in many applications, such as providing an indicator function for the surface of the mask. This distance map can also be used to generate a smoother isosurface than may be possible from the original segmentation. The user may choose between pixel distance or actual distance.



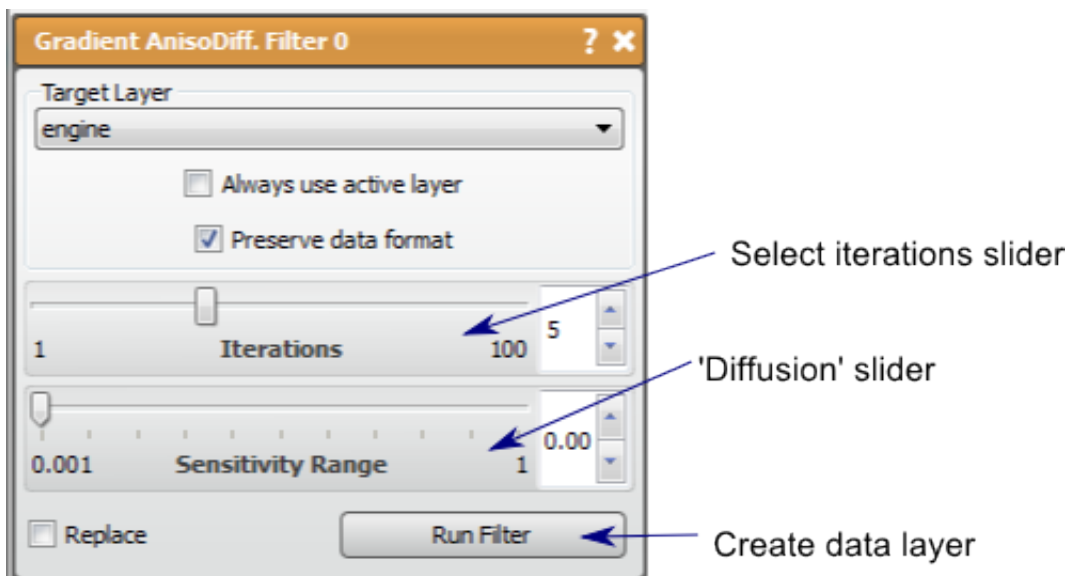
Gradient Anisotropic Diffusion

This filter performs anisotropic diffusion on a data layer and an additional edge enhancing step.

Detailed Description

The Gradient Anisotropic Diffusion performs anisotropic diffusion on a data layer and uses an additional edge enhancing step to preserve useful features. Anisotropic diffusion filtering is used to reduce noise (or unwanted detail) in images while preserving specific image features, such as edges and is based on the work of Pietro Perona and Jalhandra Malik, [PM90]. The output of this filter is a smoothed data layer.

The parameters for this filter are *Iterations* and *Sensitivity Range*. The *Iterations* parameter will control how many iterations of the filter will be executed. The more iterations, the smoother the image. The *sensitivity range* parameter controls the ‘diffusion’ of the objects in the image, therefore the higher the sensitivity, the more ‘diffusion’ and thus more smoothing.



Gradient Magnitude

This filter calculates the Gradient magnitude of a data layer and output in a data layer.

Detailed Description

The Gradient Magnitude filter will calculate the Gradient Magnitude of a data layer and output a new or replaced data layer containing the Gradient Magnitude. The Gradient Magnitude is useful as it can highlight regions with edges in the data image because those regions have high gradients.

This filter is very simple to use because there is only one input and no parameters to adjust, so choose a data layer and go.

Histogram Equalization

This filter attempts to transform histogram data to be flat as possible.

Detailed Description

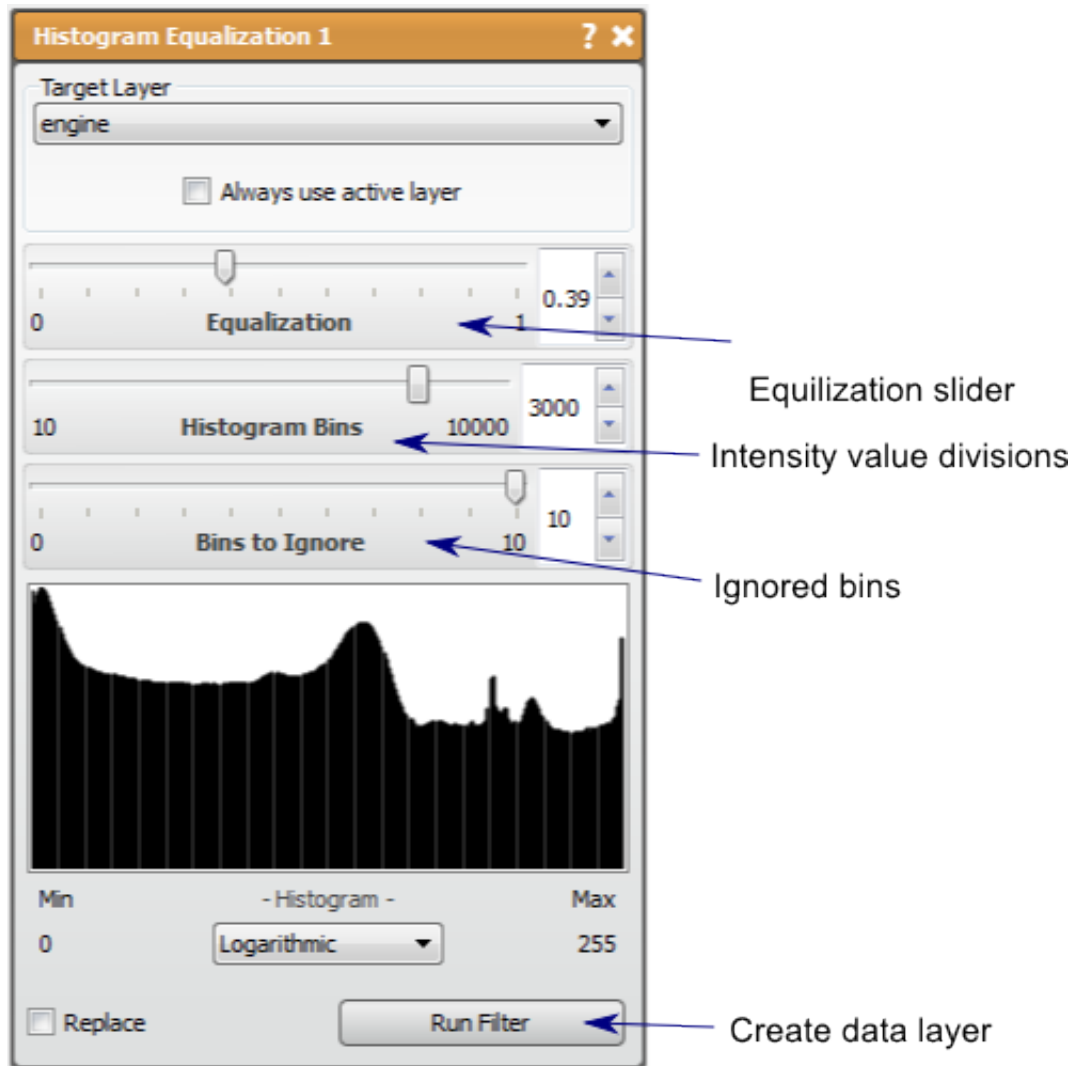
The Histogram Equalization filter will compute the histogram of the data image layer and attempt to transform the image so that the histogram is as flat as possible. The result of this transformation is most often an increase in contrast that can be considered a visual enhancement of the image.

There are three parameters to this filter: *Equalization*, *Histogram Bins*, and *Bins to Ignore*.

The *Equalization* parameter provides the most obvious difference to the image. This filter will combine the original image and the Histogram Equalized image in the result and this parameter controls the contribution of each. When the *Equalization* is 1, the result will be only the equalized image, and when the *Equalization* is 0, the original image will be returned.

The *Histogram Bins* parameter controls the number of bins used in the original histogram used to calculate the transformation, thus this parameter essentially controls the resolution of the intensity transform. This means that the lower number of bins, the fewer intensity values used in the final image, which can reduce smooth transitions in the image. Generally, having a higher bin number is better to maintain a smooth image, but the bin number need not be higher than the number of origin possible intensity values, which is usually dependent on data type. If the data type is unsigned 8 bit integers, for example, there are 256 intensity values, so more than 256 bins in the filter will not affect the result.

The *Bins to Ignore* parameter controls the number of ignored bins in the intensity transformation. When bins are ignored, the pixels that would have been in the bins are moved to the next. The result of this is that the resulting value range is altered, which could provide additional contrast in some areas.



Intensity Correction

This filter attempts to make a homogeneous intensity over the data layer.

Detailed Description

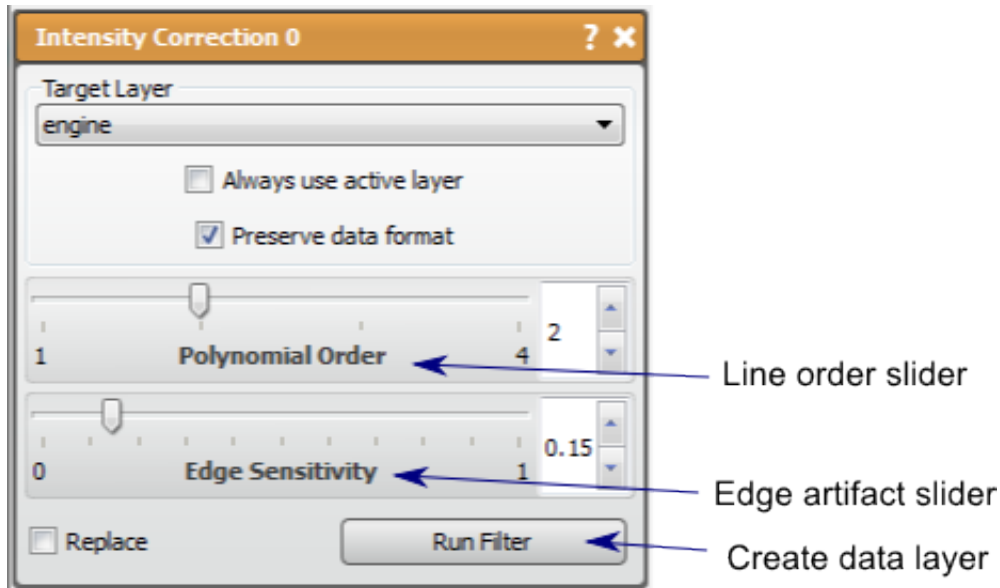
The Intensity Correction filter attempts to make the intensity over the data layer homogenous. This is very useful when dealing with MR images, as the intensity values and contrast can vary over space due to magnetic field variations. This filter will attempt to fit a spline to average intensity of the image, then to use that spline to equalize the intensity values over the space.

There are two parameters in this filter, *polynomial order* and *edge sensitivity*.

The *polynomial order* parameter controls the kind of line used to correct the image. A *polynomial order* of 1 assumes that any changes in intensity is linear through the image. A higher *polynomial order* may be needed if there are more complex patterns of intensity variation, such as seemingly isolated regions of high or low intensities.

The *edge sensitivity* parameter controls the correction algorithm that prevents artifacts do to edges in the image. The

higher the sensitivity, the less the algorithm will correct for edge artifacts. As *edge sensitivity* approaches 0, the algorithm attempts to leave edges unchanged and may suppress the Intensity Correction altogether.



Segmentation Level Set

This filter iteratively seeks regions similar to a given seed volume.

Detailed Description

The Segmentation Level Set filter is an iterative segmentation filter that uses a seed volume to find the regions in a data volume that are similar to the original seed. From mask layer provided as the seed, the statistics of the seeded region are calculated. The filter will then expand the segmented region to surrounding pixels that match the statistics of the original seeded area. The propagation may also retract the segmented area in some cases if the seeded areas do not meet the criteria from the threshold range or edge weight. The filter will continue to expand (or contract) until the iterations are terminated.

To run this filter, choose a data layer and a mask layer to use as the seed. The more complete the seed layer is, the fewer the iterations needed to obtain the desired segmentation. With the proper layers selected, alter the parameters as desired and press *Run Filter*. A new layer will appear and will update the results of the filter with each iteration. The user may wait until the filter performs the set number of iterations, or may terminate the process if the desired volume is achieved by pressing Finish in the new layer created in the layer window.

The parameters in this filter to modify are: *Iterations*, *Threshold Range*, *Curvature Weight*, *Propagation Weight*, and *Edge Weight*.

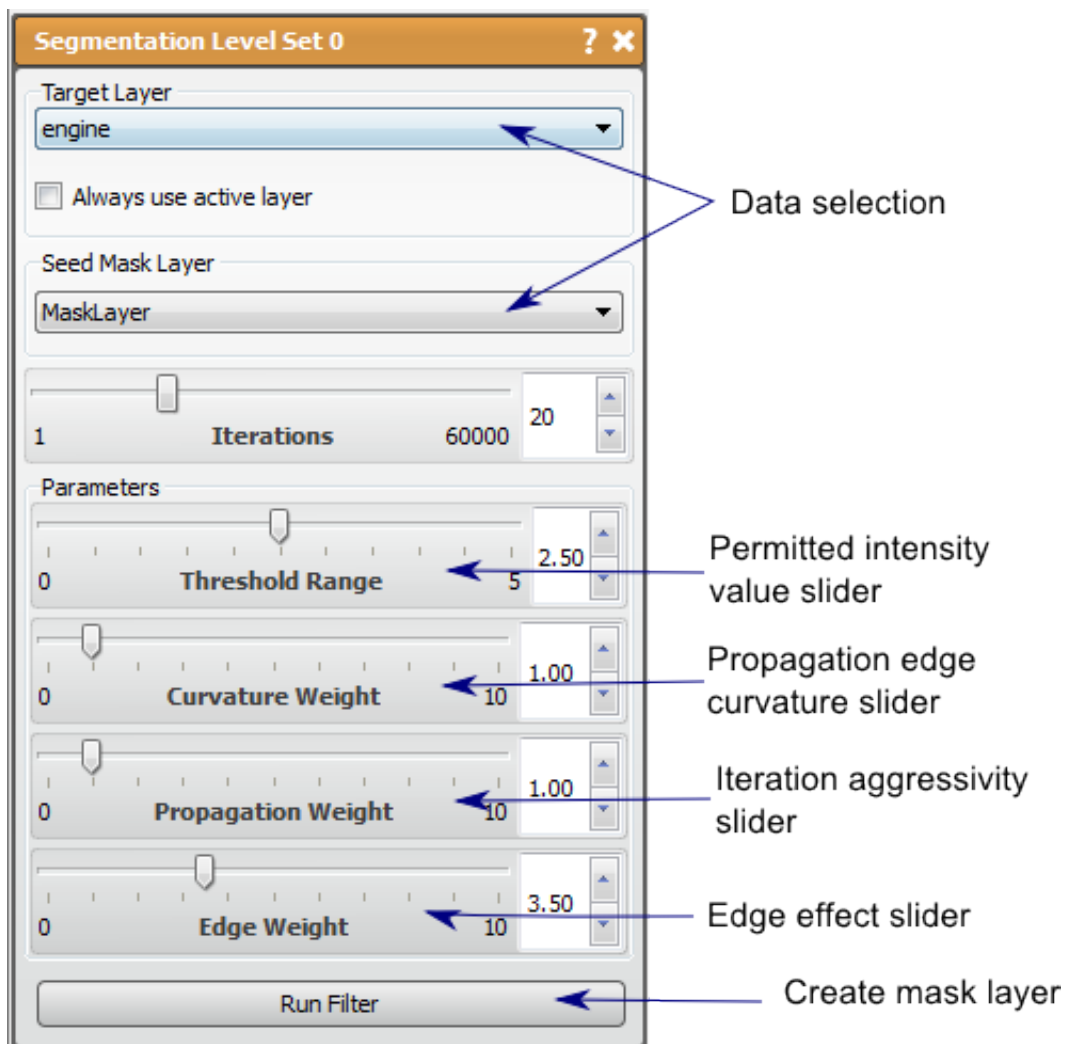
Intensity values permitted to be included in the propagation of the segmented region. This range is related to the standard deviation of the original seed volume. If this parameter is set too low, so of the seed region will be removed as the segmentation propagates. If this parameter is set to 0, there will be no output data. This is the most strict parameter, in that all other parameters are met after the threshold range is established.

The *CurvatureWeight* parameter will control the propagation edge in terms of curvature. The higher the *CurvatureWeight*, the smoother the boundary will be. It should be noted that curvature can only be weighted in the range set by both the *Threshold Range* parameter and the *Edge Weight* parameter, and therefore may not alter the end result if the regions defined by these other parameters is too narrow.

The *Propagation Weight* parameter controls how aggressive each iteration will be in determining surrounding pixels that may fit the criteria of the filter. The higher the *Propagation Weight*, the further the filter searches on each iteration, and therefore the faster it fills a region that matches the criteria. Having a high propagation may also cause the to jump gaps that may not be desired, so it is prudent to reduce the propagation weight in such cases. The *Propagation Weight* should be greater than 0 to avoid reduction of the volume.

The *Edge Weight* parameter will control how much effect the edges in the data volume effect the propagation pathway. The Higher the *edge weight*, the more the filter avoids filling in edges, though the propagation may cause the expansion to fill around the edge. It should be noted that the filter is very sensitive to this parameter, so that it should be increased with small increments. Too high of a *edge weight* could cause holes to appear in what may seem like homogeneous regions.

On a final note, when the segmentation level set filter is run with the *edge* and *curvature weight* at zero and high enough iterations, the result will likely seem similar to the Confidence Connected filter, but will require much more compute time to reach it. This can be useful to try when first using this filter on new data, as it will help tune the threshold range better. With that in mind, suggested parameters for novices are: high *Threshold Range* (above 2.5, but 5 might be best), high *Propagation Weight* (Use max unless doing fine tuning or very coarse data), and low *Edge Weight* (less than 1 unless very smooth data). The filter is not as sensitive to the *Curvature Weight* and the *Iterations* can be increased as much as desired.



2.2.2 Data Filters

Arithmetic

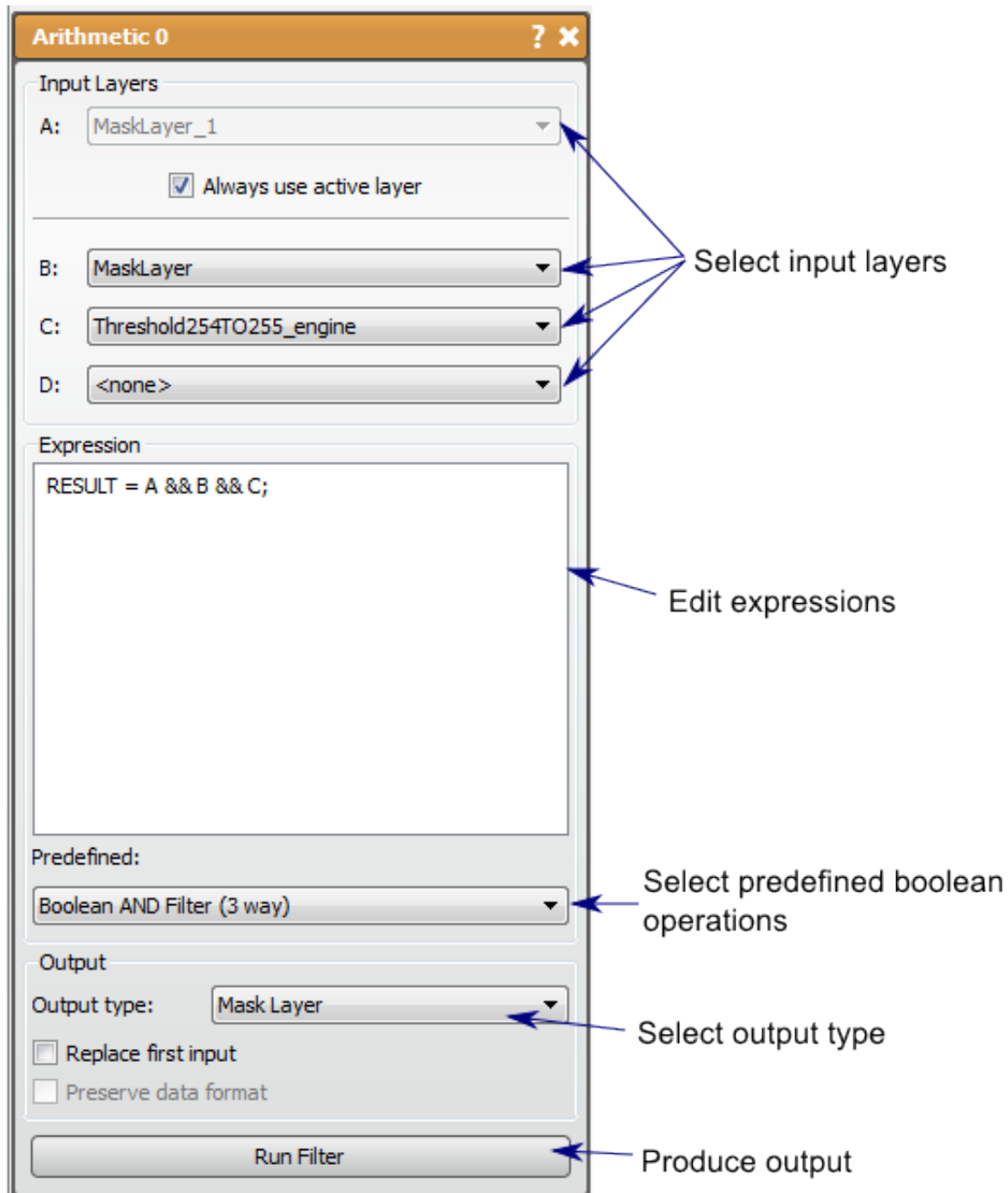
This filter runs basic boolean operations on a pixel by pixel basis.

Detailed Description

The Arithmetic filter allows the user to run basic arithmetic and boolean operations on both image and mask data on a pixel by pixel basis. This filter can handle up to four inputs and can be used to perform calculate any expression provided by the user (correct syntax is `RESULT = \<expression\>;`). There are also many saved expressions that exists as separate filters or tools, the main difference is the functionality to handle more than two inputs.

Any operation can be performed on either type of data, but it should be noted that if a boolean operator is performed it uses zero and non-zero values as the only two possible cases. This applies to using data layers and also values that may be the summation of mask layers.

This filter also enables outputs of both data and mask layers. The filter will switch the output the match the input if the input is changed, but it can be changed afterward. If there are multiple values in the mathematical result, but mask layer output is chosen, the filter will perform a threshold to try and capture the relevant data in a mask, usually capturing values over zero in the data.



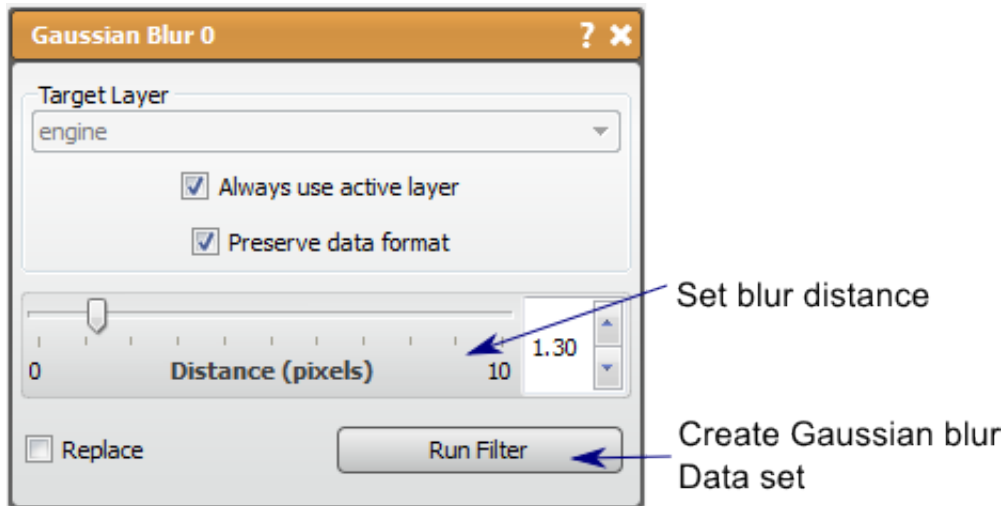
Gaussian Blur

This filter performs a smoothing on the data layer.

Detailed Description

The Gaussian Blur filter performs a smoothing on the data layer using a discrete Gaussian kernel. For every pixel (except the edges), this filter performs a weighted averaging of the neighboring pixels where the weights are a 3D Gaussian distribution centered at the pixel. This filter will reduce noise by averaging it out, but will also reduce edges.

The *distance* parameter set with the slider bar in the tool window determines the size of the neighborhood so that the bigger the distance, the greater the area of averaging and thus more blurring. Since the *distance* parameter is the only exposed parameter, the variance is dependent on the distance.

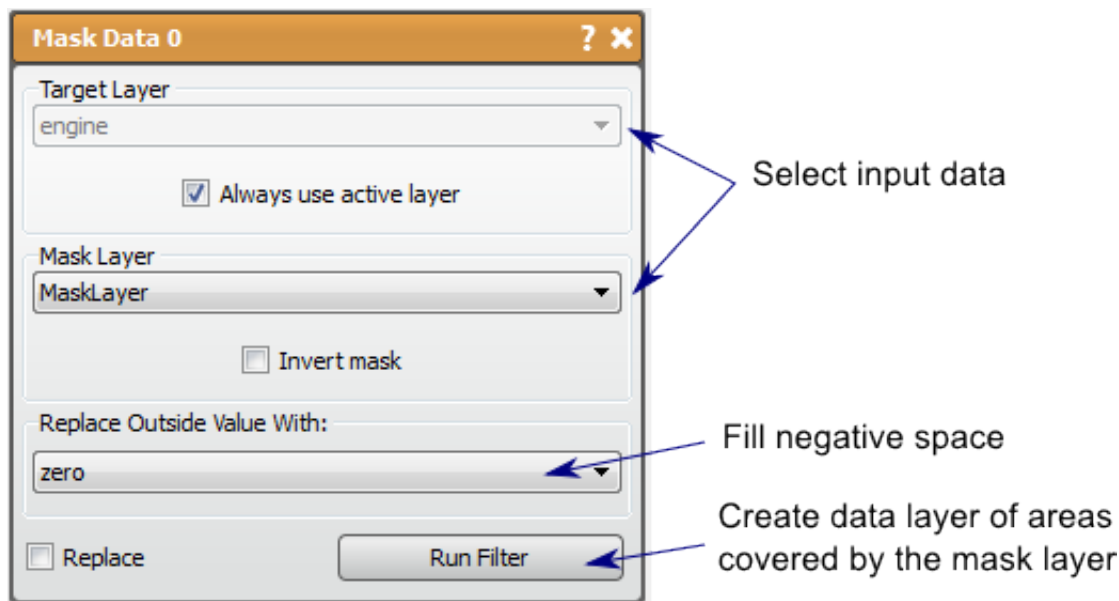


Mask Data

This filter will use a mask layer to discard data in the data layer.

Detailed Description

The Mask Data filter will use the second input of a mask layer to selectively discard (retain) data in a data layer. The pixels in the data layer that correspond to marked pixels in the mask layer will be kept, and the remaining pixels will be discarded by setting them to some background value. The background value can be set to zero, the original max or min value, or the new max or min value. This can be useful in isolating regions of interest and will eliminate noise from outside the region. This filter could also cause artificial edges to occur.

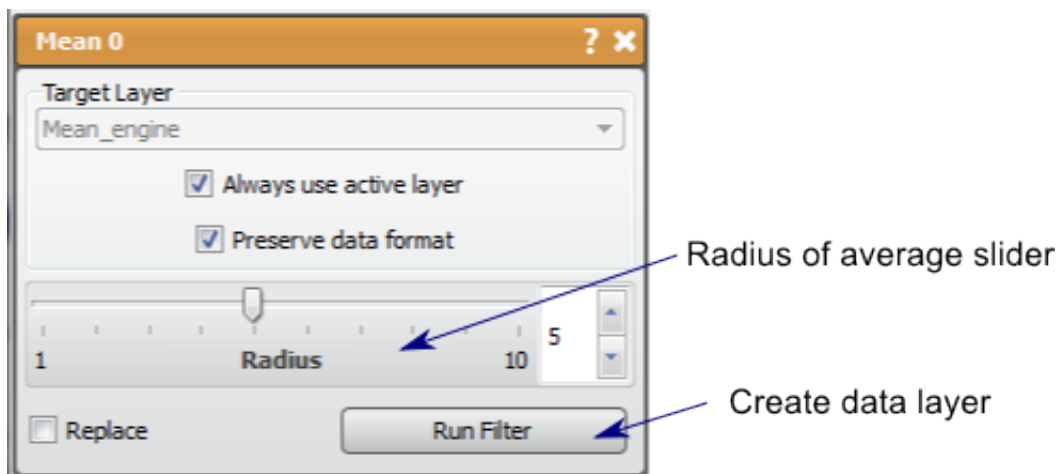


Mean

This filter performs smoothing on the data layer using a step function.

Detailed Description

The Mean filter performs a smoothing on the data layer using a discrete step function kernel. For every pixel (except the edges), this filter performs an averaging of the neighboring pixels. This filter will reduce noise by averaging it out, but will also reduce edges. The neighborhood size is controlled by the distance parameter, the greater the distance, the more smoothing.



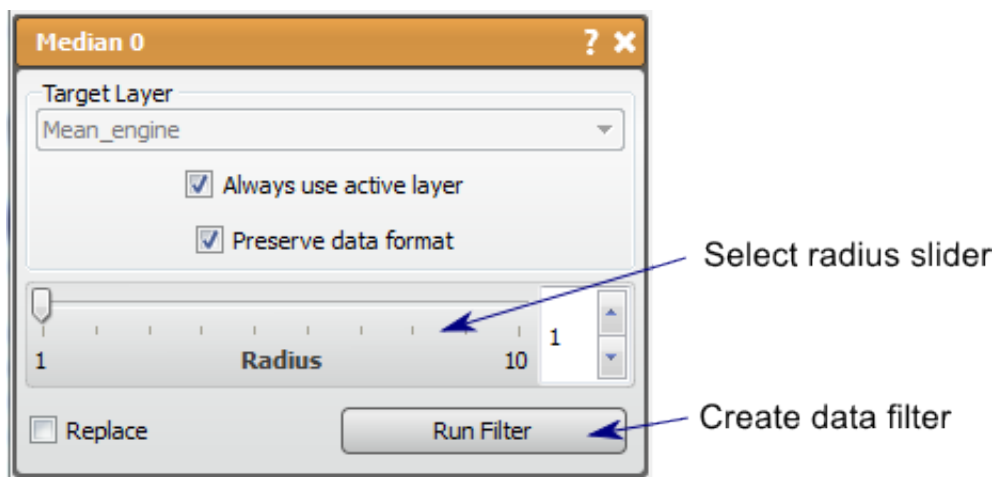
Median

This is a nonlinear filter for reducing noise but maintaining edges.

Detailed Description

The Median filter is a non-linear filter that will reduce noise (speckle), but may also preserve or enhance sharp boundaries. For every pixel (except the edges), this filter finds the median value of the neighboring pixels and uses this value to replace the original pixel. This causes the edges to be preserved, if the neighborhood is appropriately sized, while eliminating subtle variations.

The neighborhood is controlled by the *distance* parameter. Though increasing the distance will also increase the smoothing, it is because the size requirement for regions to not be eliminated. Clear divisions of large regions will increase in contrast and smaller regions will be smoothed out.

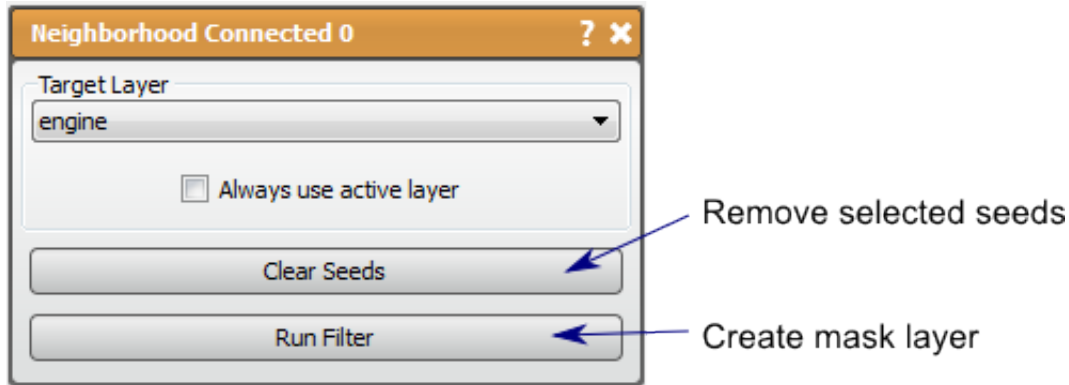


Neighborhood Connected

This filter creates a mask indicating pixels within a value range and connected to seed points.

Detailed Description

The Neighborhood Connected filter is like a combination of the Threshold tool and the Connected Component filter. The user places seed points in the volume using the 2D viewers (**left mouse** to place, **right mouse** to remove). Then the filter will create a mask layer marking the pixels that are within the data value range of the seed locations that are also connected to the seed points. This tool is useful in segmenting long and winding structures as long as there is sufficient contrast between the desired structure and the surrounding medium.



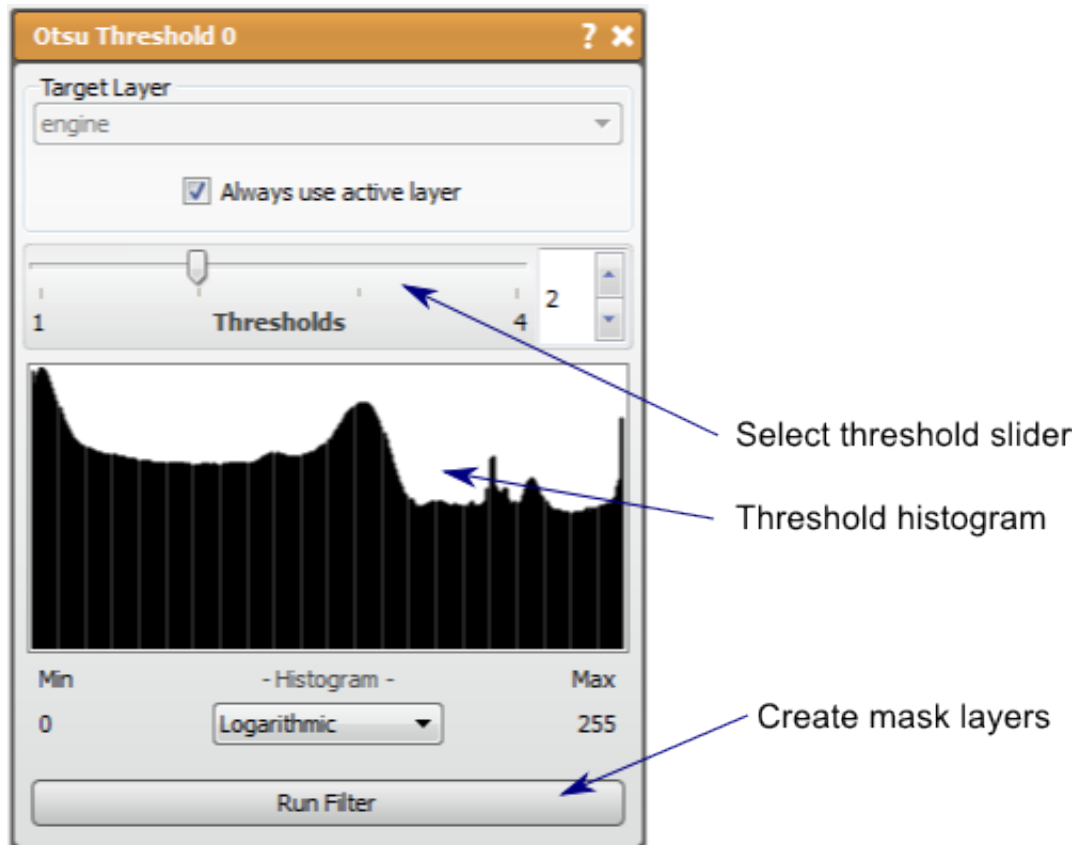
Otsu Threshold

This filter creates a mask of regions data clusters.

Detailed Description

The Otsu Threshold filter attempts to cluster data values in a data layer based upon how similar they are and create mask layers identifying the regions of similar data. This filter provides a good quick segmentation in many cases. There is a [Wikipedia article](#) containing a good description of the Otsu threshold algorithm.

Using the Otsu Threshold filter is simple. The only parameter in this implementation is the number of thresholds (1 to 4), or the number of divisions in the volume. The results of this filter are mutually exclusive (no output will mask the same pixel as another layer) and collectively exhaustive (the whole volume will be mask by one of the output layers) so that there will be one more mask layer generated than thresholds used. A histogram is presented to aid the user in choosing the threshold number.



2.2.3 Large Volume Tools

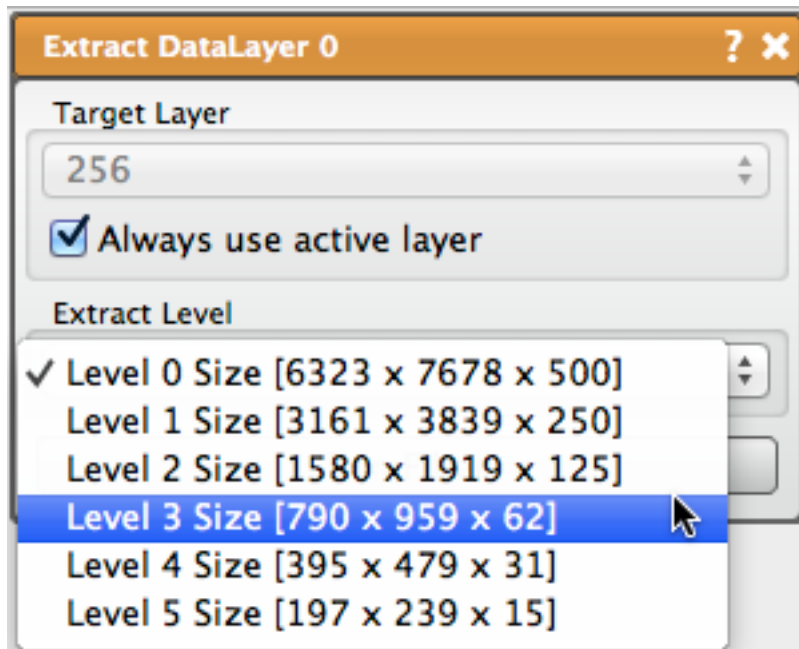
Extract Data Layers

Select one of the levels of detail from the large data volume and convert it into a normal data layer.

Detailed Description

Once a volume is converted, the user can run any filter on the data. Currently, data layers should not have more than 4096 pixels in a single dimension, as data layers are not rendered as bricks, but as a single texture.

It is expected that the ExtractDataLayer and Crop tools will often be used together, with a user cropping a region of the large volume and then selecting an appropriate level of resolution to convert the large volume layer into a normal data layer.



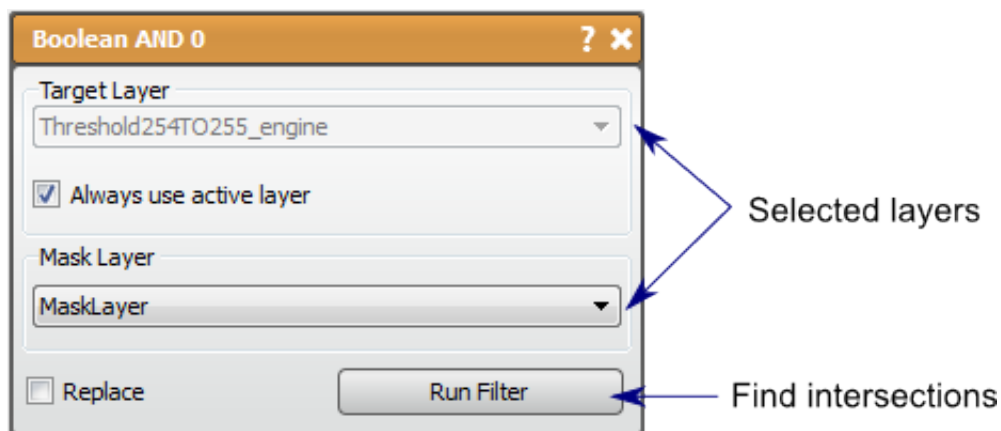
2.2.4 Mask Filters

Boolean AND

The Boolean AND filter finds the intersection of two masks.

Detailed Description

The Boolean AND filter will input two mask layers and output the intersection of the two masks, i.e., the pixels that are masked by both of the two input masks.

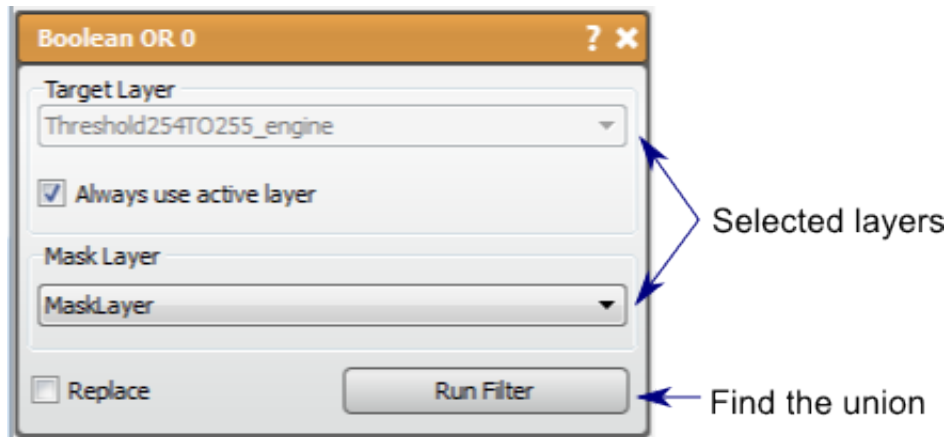


Boolean OR

The Boolean OR filter produces the union of two masks.

Detailed Description

The Boolean OR filter will input two mask layers and output the union of the two masks, i.e., the pixels that are masked by either of the two input masks.

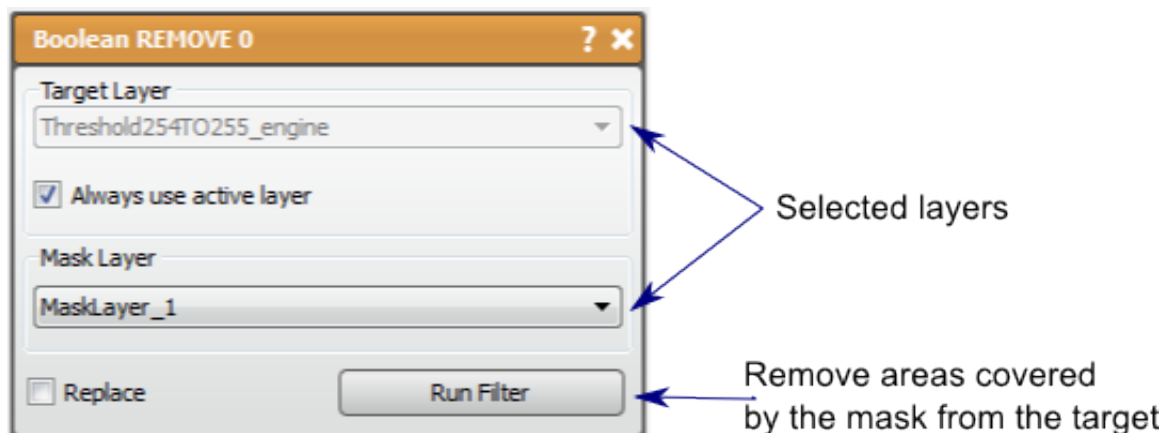


Boolean Remove

The Boolean REMOVE filter removes the intersecting region of two masks from the first.

Detailed Description

The Boolean REMOVE filter will input two mask layers and use the second input to remove the intersecting regions from the first. The result of this can be thought of as the first region minus the second region.

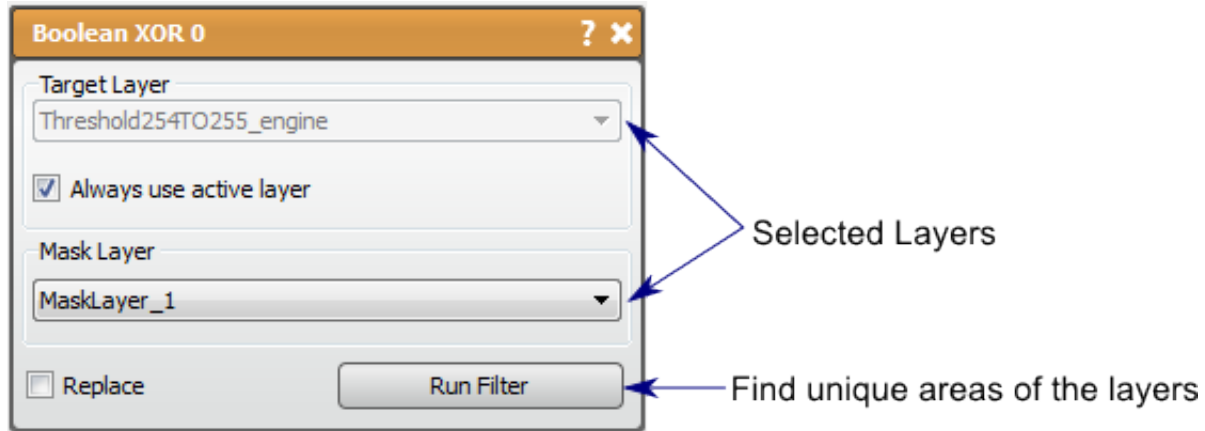


Boolean XOR

This filter produces the unique regions of two masks.

Detailed Description

The Boolean XOR filter will input two mask layers and output the unique union regions of the two masks, i.e., the pixels that are masked by exactly one of the two input masks. This filter is the same as removing the result of the Boolean AND filter from the result of the Boolean OR filter.

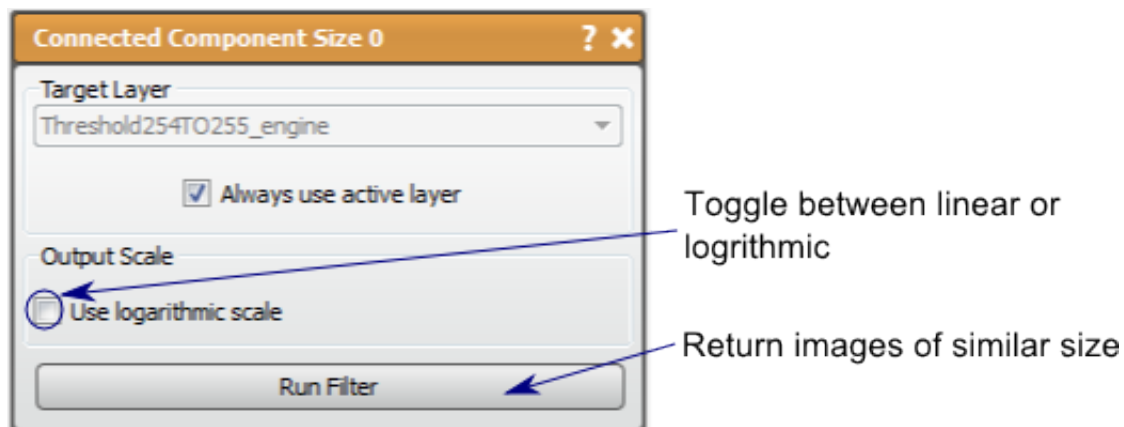


Connected Component Size

Similar to the Connected Component filter, this filter will return a image data layer but the data value will correspond to its size (logarithmic or linear scale).

Detailed Description

The Connected Component Size filter is similar to the Connected Component Filter, but will return a image data layer labeling each connected region with a data value corresponding to its size (logarithmic or linear scale). This allows the user to manipulate the connected regions in the volume based on size. For example, thresholding real data can yield several regions of interest that are correct maskings of the data, but may also capture small regions of noise in other areas of the scan. These small regions of noise can be eliminated using the Connected Component filter by using the Threshold tool on the result.



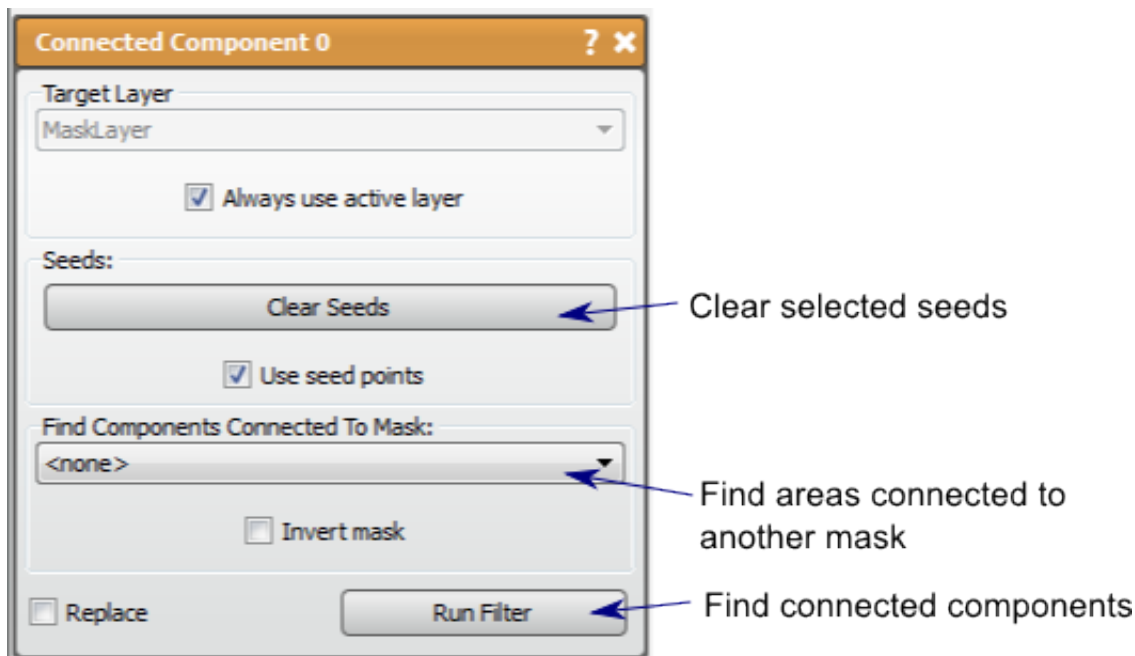
Connected Component

This filter finds connected data regions in a mask layer that are coincident to the defined seed points.

Detailed Description

The Connected Component filter finds connected data regions in a mask layer that are coincident to the defined seed points. This filter can be used to eliminate mask data that is not connected (separated by a blank pixel) from a desired region and can thus reduce noise in the segmentation.

The seed points are chosen by **left clicking** in the 2D viewers or deleted by **right clicking** on existing seed points. Multiple connected regions can be selected with these seed points. Another mask can be used as a seed. Using a mask layer second input field will cause the filter to find regions from the first input that are overlapping or next to any data in the second input. Additional seed points can be used or ignored with the use seed points option.



Fast Binary Dilate Erode

This filter will make a label mask bigger or smaller.

Detailed Description

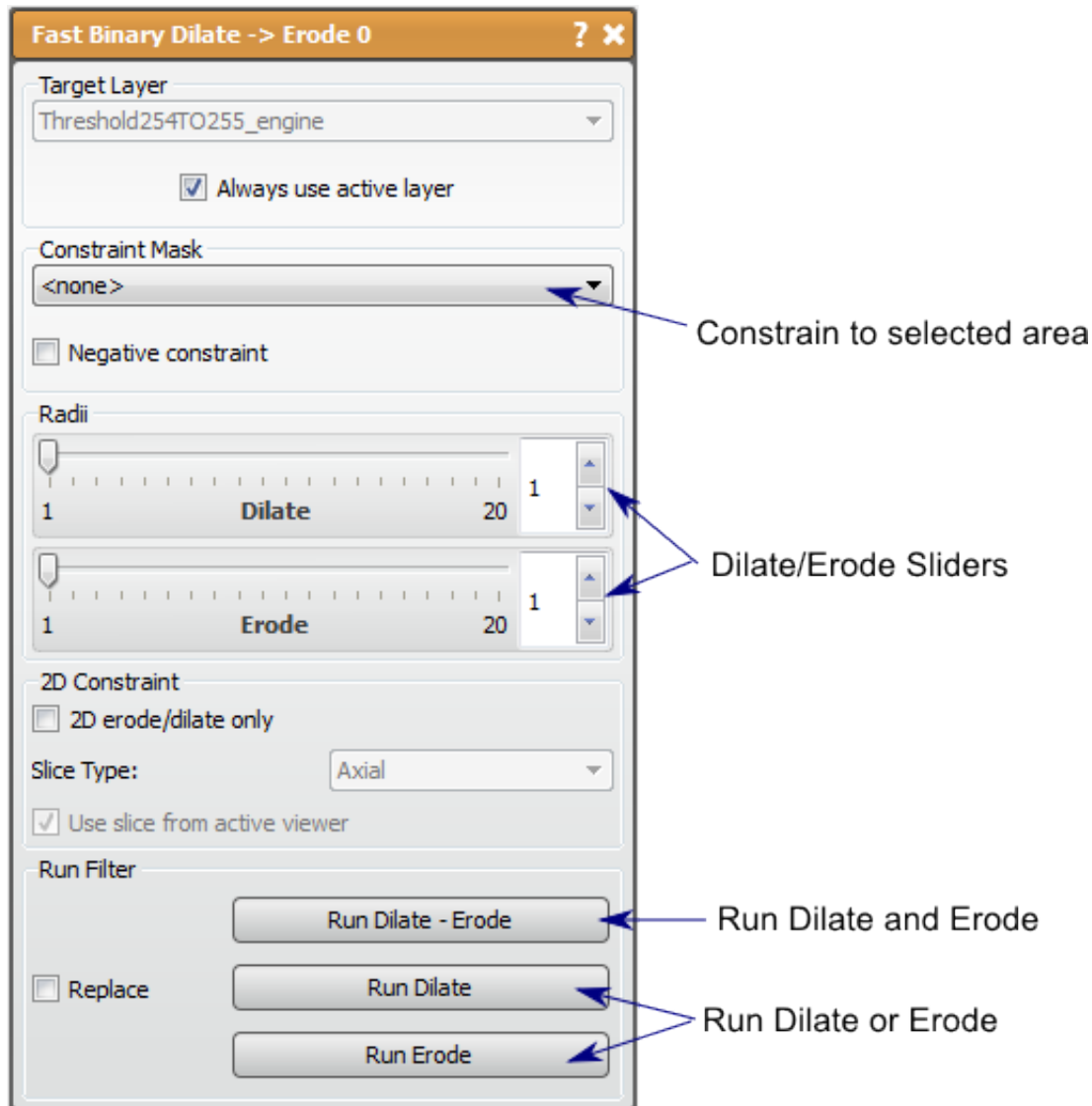
The Fast Binary Dilate -> Erode filter will make a label mask bigger (dilate) or smaller (erode). This filter is very useful in smoothing a mask layer, as performing a dilation and erosion on a mask volume will fill in some of the details on the surface. This can be desirable if noise is the detail being filled in. The opposite (erode then dilate) will also smooth, but it will reduce protruding details in the volume. Either type smoothing should be done carefully as important details may be smoothed in addition to the noise of the segmentation. The amount of smoothing is directly proportional to radius of each step. The radii are controlled by the slider in the tool window.

There are several features of this filter that allow high level manipulation of the dilate and erode functions. First is the ability to add a constraint to the algorithm. The constraint will not allow the dilation or the erosion of the original mask to exceed the boundary of the constraint mask. Another feature is to dilate and erode in a specified plane, which can

be specified in the viewer or with the drop down menu (disable the *use slice from active viewer* option) An example of when this is useful is when there is a much bigger spacing in one direction, therefore the 3D dilation extends much further in space in that direction and can cause unwanted closing.

It should be noted that the edge of the volume provides complications to this filter. If the mask layer is touching the edge of the volume and erode is performed, the edge slice will perform a 2D erosion only, so that the mask still connects to the edge. This can become a problem when the original mask data is near the edge, but not touching, then *dilate -> erode* is run so that the dilate causes the mask to touch the edge. The erosion will then leave the mask touching the edge of the volume.

This filter and Smooth Binary Dilate -> Erode are similar in function, but are slightly different in the results. As the name suggests, Fast Binary Dilate -> Erode is the faster filter of the two, but the solution may produce corners in the data instead of smooth surfaces. The runtime of Fast Binary Dilate -> Erode is dramatically less and will the results will be indistinguishable with moderate radii and number of eroding and dilating steps.

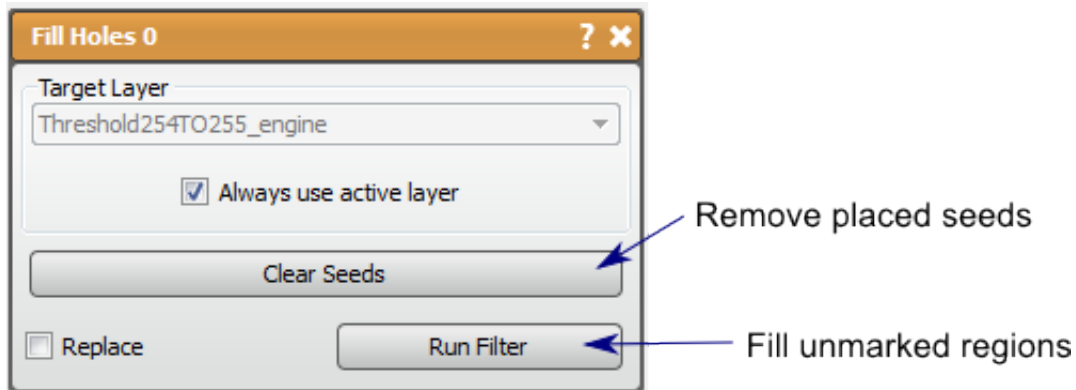


Fill Holes

This filter can be used to fill unmarked regions in a mask layer.

Detailed Description

The Fill Holes filter can be used to fill unmarked regions which are surrounded by a data in a mask layer. This is useful when automated segmentation techniques leave false negative noise, or regions that should have been marked but aren't. This filter will only fill regions that are completely surrounded with mask data. Therefore, if the filter appears to not work at filter, check the holes and make sure they are not exposed to the background. The seed points can be used to if there is a hole that is on the edge, but the hole must not be the majority of the edge, i.e. the background. In this case the seeds are used to identify the background. Seed points are not needed otherwise.



Smooth Binary Dilate Erode

This filter will make label masks bigger or smaller with additional smoothing to the mask.

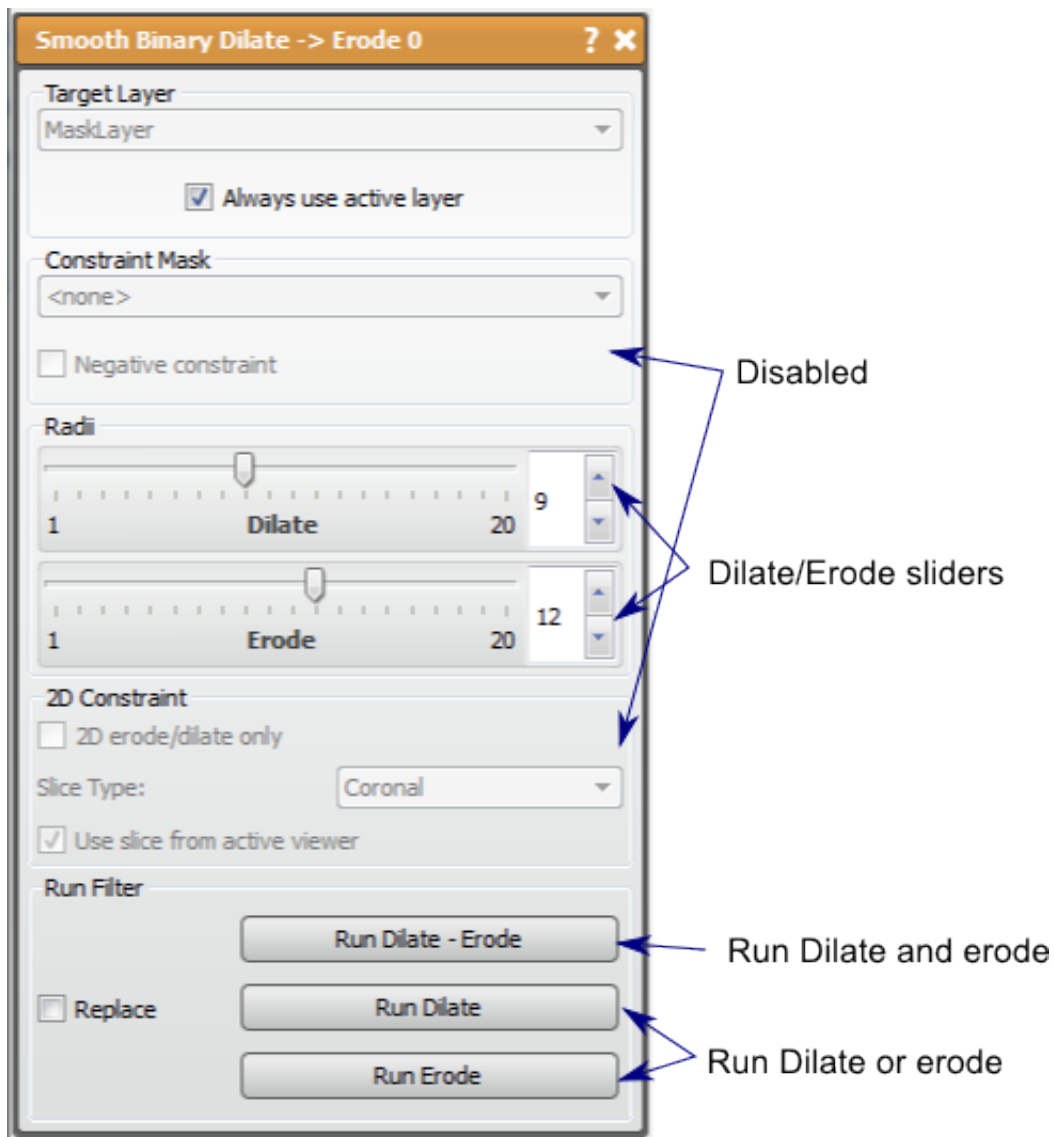
Detailed Description

This filter will make a label mask bigger (dilate) or smaller (erode). This filter is very useful in smoothing a mask layer, as performing a dilation and erosion on a mask volume will fill in some of the details on the surface. This can be desirable if noise is the detail being filled in. The opposite (erode then dilate) will also smooth, but it will reduce protruding details in the volume. Either type smoothing should be done carefully as important details may be smoothed in addition to the noise of the segmentation. The amount of smoothing is directly proportional to radius of each step. The radii are controlled by the slider in the tool window.

Though there appears to be several features similar to the Fast Binary Dilate -> Erode filter, these have been disabled due to complexity issues from this filter.

It should be noted that the edge of the volume provides complications to this filter. If the mask layer is touching the edge of the volume and erode is performed, the edge slice will perform a 2D erosion only, so that the mask still connects to the edge. This can become a problem when the original mask data is near the edge, but not touching, then *dilate* -> *erode* is run so that the dilate causes the mask to touch the edge. The erosion will then leave the mask touching the edge of the volume.

This filter and Fast Binary Dilate -> Erode are similar in function, but are slightly different in the results. As the name suggests, Smooth Binary Dilate -> Erode will produce a smoother surface after each step due to an additional correction in the algorithm. This additional step adds to runtime dramatically. You should use this algorithm if the radii used are large, or if the erosion and dilation are performed multiple times each.



2.2.5 Tools

Copy/Paste

This tool allows the user to copy and paste data from one mask to another.

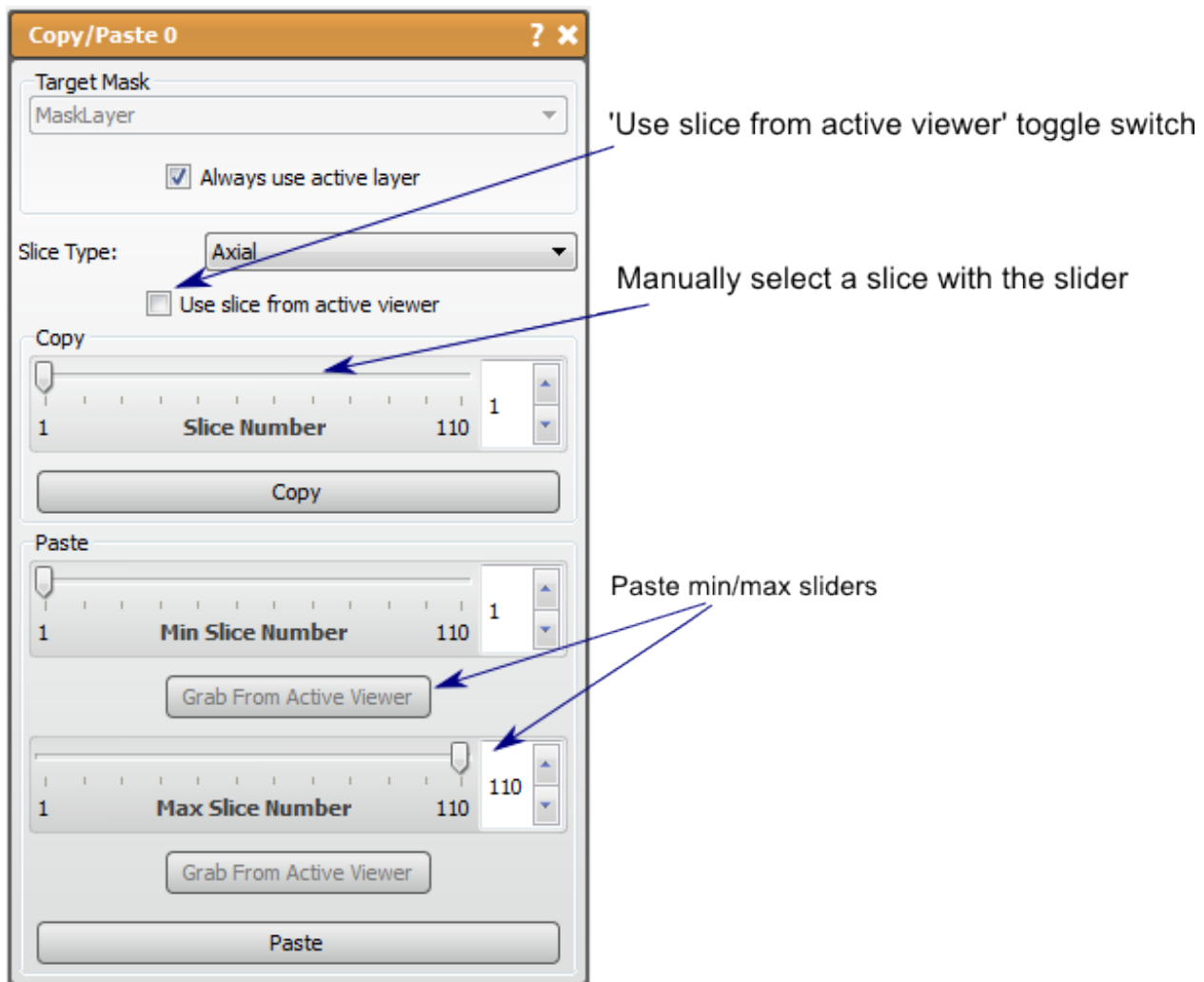
Detailed Description

This tool uses only mask layers as inputs. It is used to modify mask layers to repeat a slice of interest. This tool is similar to the basic features found in Copy/Paste Mask Slice. The key difference between the edit functions and this tool is that this tool provides more flexibility with the pasting options, offering pasting anywhere from one slice to the whole volume. The editing options only offer the extremes of pasting a single slice or punching a copied slice through the whole volume. This tool is most useful for pasting multiple slices but not necessarily the entire volume.

The Copy/Paste tool has two functions, copy and paste. The copy buffer only saves one copied slice at a time. This means that only one slice can be pasted, though it can be pasted to many slices. The slice to be copied can be specific in two ways, from the viewer (when *use slice from active viewer* is checked) or using a slider and specifying the direction (make sure *use slice from active viewer* is unchecked). You should be careful using either option to make sure the correct plane is active/selected. To copy the slice, press the *copy* button near the middle of the tool window.

The paste option is a little more complicated, as the user must specify the slice boundaries for the pasting. This can be done using the provided sliders or by navigating the slice and pressing the grab from viewer. The user will need to ensure that the same plane is chosen for the pasting and the copied slice. Also, it should be noted that the software will not constrain the maximum to be more than the minimum, or vis versa. If the maximum is lower than the minimum, the tool will still use the slices described and the slices will be pasted. Press the *paste* button when the correct slice range is specified to finish.

This tool will modify the original layer without an option to create a new layer with the action.



Crop

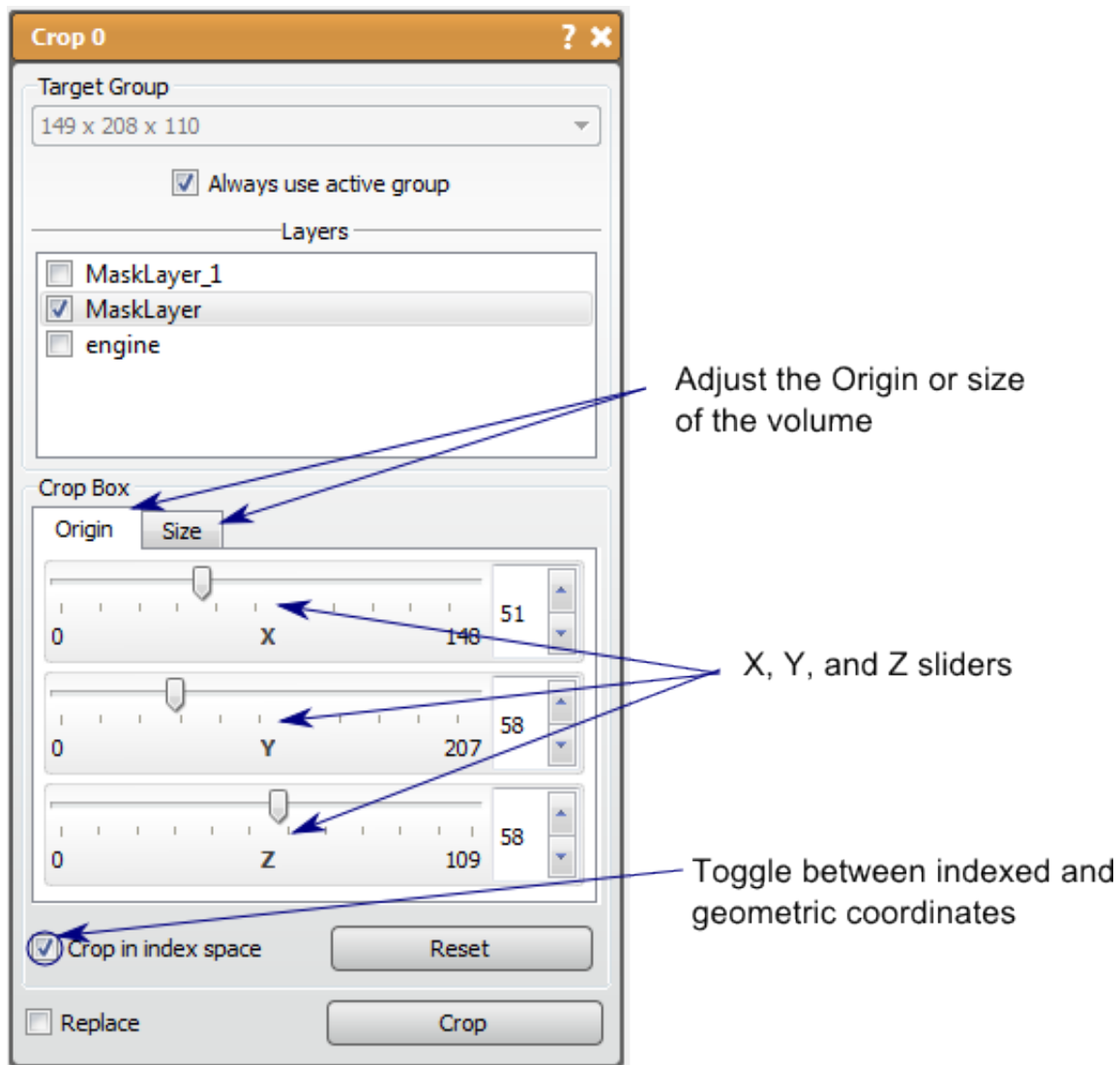
The Crop tool will crop one or more layers to a specified size. This tool will clip multiple layers from the same group to the same size by choosing them in the list of layers near the top of the window. The Crop tool also works on the new Seg3D large volume data layer format.

Detailed Description

The size of the new volume can be specified using the sliders in the tool window, or by adjusting the widget in the 2D viewing windows. The sliders in the tools window will specify the origin and the size of the cropping region. The buttons over the slider specify which parameter is displayed. It should be noted that you can display the indexed or geometric coordinates by checking/unchecking the *crop in indexed space* option near the bottom of the window.

The Crop widget is shown as a red box in the 2D viewers. This can be adjusted in any plane by clicking and dragging on the edge to change the size, or in the middle to change the origin. The mouse pointer will change to indicate which function will be affected. Press *Crop* to finish cropping the volumes.

Currently, data layers should not have more than 4096 pixels in a single dimension, as data layers are not rendered as bricks but as a single texture.

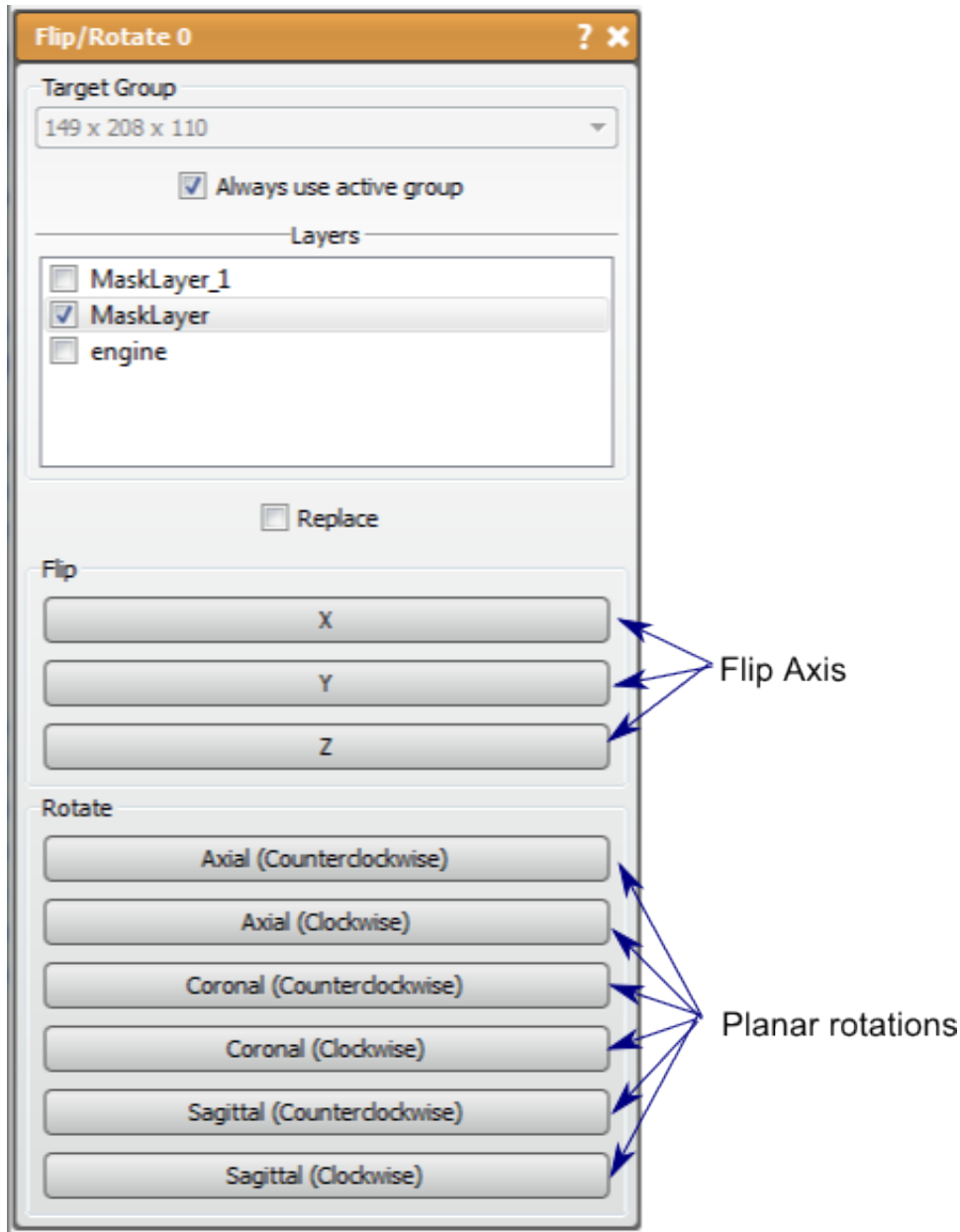


Flip/Rotate

This tool will flip or rotate one or more layers around an axis.

Detailed Description

The Flip/Rotate tool will flip or rotate 90 degrees one or more layers around a specified axis. This tool will flip/rotate multiple layers from the same group to the same size by choosing them in the list of layers near the top of the window. After choosing the desired layers to flip/rotate, the action is performed when the button for that action is pressed. There are three flipping options (flipping each axis) and six rotation options (clockwise and counter-clockwise in each plane).



Implicit Model

This filter creates a segmentation by generating an implicit surface represented by a radial basis function (RBF) interpolation.

Detailed Description

In our algorithm, an implicit surface that is represented by a radial basis function (RBF) interpolation is used to generate a segmentation [TOBrien02].

As an input, the algorithm is provided with a set of points on the boundary of the object that needs to be segmented from the images. Each slice should contain at least three points for our algorithm to return a reasonable segmentation. Our algorithm constructs an approximate function whose value vanishes at the given points and whose value is positive inside the object and negative outside.

In order to construct such a function, a radial basis function interpolation technique is used. The value of the function is set to be zero at the given points. On each slice, the polygon with the smallest perimeter passing through all points is constructed (we use an approximate algorithm for the Euclidean traveling salesman problem described below). This polygon is used to determine approximate normals on the surface of the object. We use the approximate normals to construct two sets of points inside and outside the polygon (at some constant distance from the given points). We then set the value of the function to be some positive and negative constant inside and outside the polygon, respectively. The thin plate RBF spline is used to interpolate the points and construct a function, $f(x)$, whose level set for $f(x)=0$ approximately constructs the surface of the object.

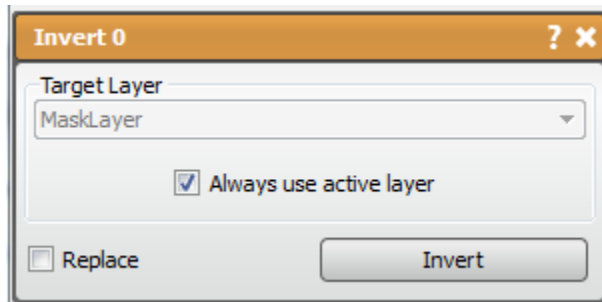
The construction of the polygon is described in this paragraph. The minimization of the perimeter of a polygon passing through a set of points is the Euclidean traveling salesman problem (TSP). On each slice, the given points are used to construct a graph in which the points are the nodes and the distance between them are the weights of the corresponding edges. The approximate algorithm for the Euclidean TSP begins by constructing a minimal spanning tree (MST) of the graph. The nodes on the tree with an odd number of neighbors are then matched such that the sum of their edge weights is minimized (we use a brute-force algorithm since the number of points are small). These edges (that connect the matched nodes) are added to the tree, and the new graph (the MST + matched edges) is traversed in the depth-first search (DFS) manner to obtain the polygon.

Invert

This tool will make a negative of the input data.

Detailed Description

This tool will invert the data in a layer to its minimum and maximum, making a negative of the input data. If performed on a mask layer, this tool will be a mask of everything in the volume besides the original mask. If performed on image data, the intensities will be reversed so that the dark areas look light and vice versa.



Measure

The Measure tool will calculate the distance between two points chosen in the viewer windows. The points, line segments, and measurements in the 2D viewers and record the measurements in a table in the tools window. This tool will calculate and retain several different distances allowing the user to copy the measurements to other software (as a text or into a spreadsheet) using the two copy functions in the tool window (*selection* or *all data*).

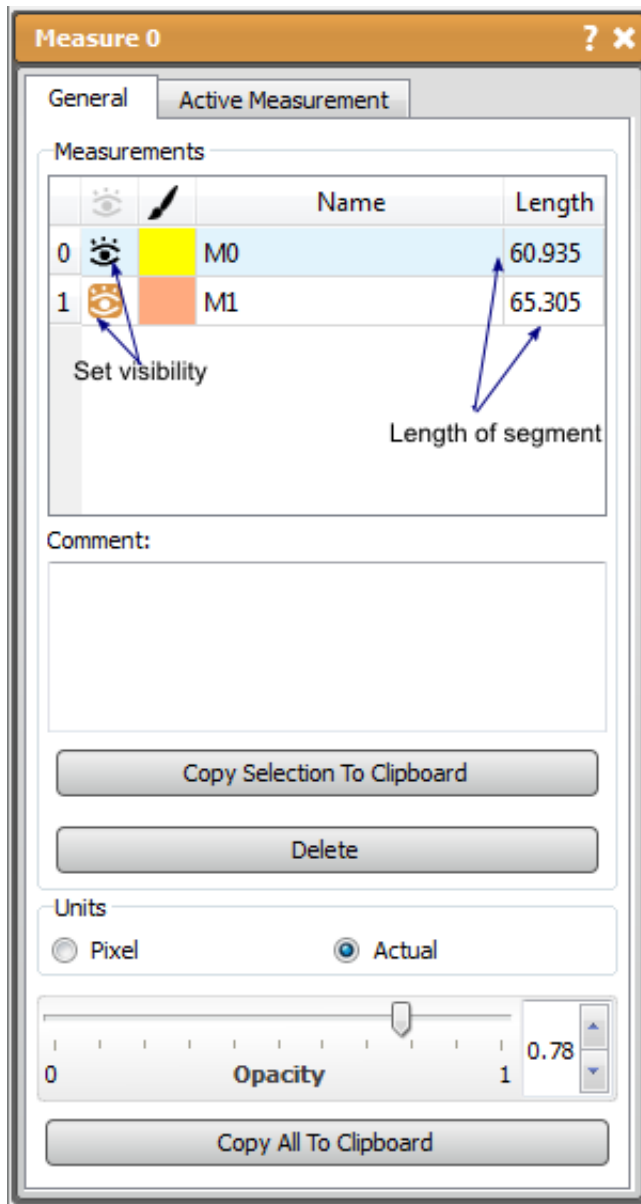
Detailed Description

The user can choose which measurements are visible, change the color, name, and add comments of each of the measurements by **double clicking** on the parameter in the table, or by highlighting the measurement in the table and clicking the active measurement tab. A new measurement automatically becomes active. Using this tab, the user can also center all the active viewers (with picking from other planes allowed) on the points that are in the measurement.

Back in the general tab, the active layer can be deleted and the opacity of the visualizations in the viewers can be adjusted. Both tabs allow switching between pixel (indexed) and actual (geometric) distances. When this is changed in the active measurement tab, it changes all the measurements.

Table 2.11: Keyboard and Mouse Actions in the for the Measure Tool

Action	Function
ctrl/cmd	snap to axis
middle mouse button	snap to slice

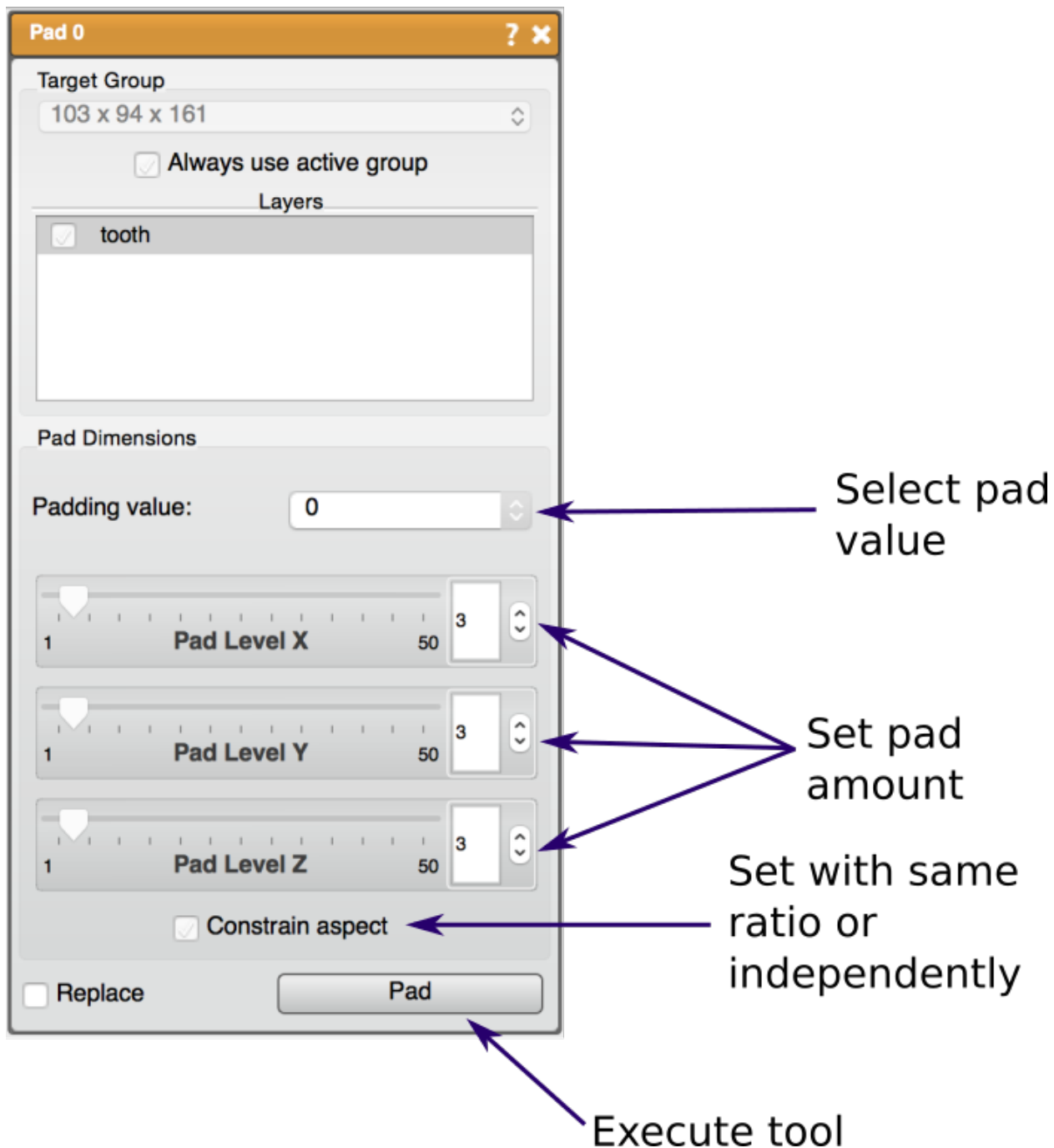


Pad

The Pad tool allows the user to pad one or more volumes. This tool can be used on both mask and data layers.

Detailed Description

The Pad amount is set manually using the sliders. There are options to resample with the same ratio in every direction (check *Constrain Aspect* option) or to resample each direction independently (must uncheck *Constrain Aspect* option to enable). Pad values can be zero, the dataset min or max.



Paint Brush

The Paint Brush tool is a tool that allows full manual creating and editing of mask layers. This tool allows the user to specify explicitly which pixels to include in the mask label. This tool can only manipulate existing mask layers (can be empty).

Detailed Description

The Paint Brush size can be changed with the scroll wheel or in the tools window using the slider. The **left mouse** button paints and the **right mouse** button erases, but this can be reversed by checking the *Erase* option.

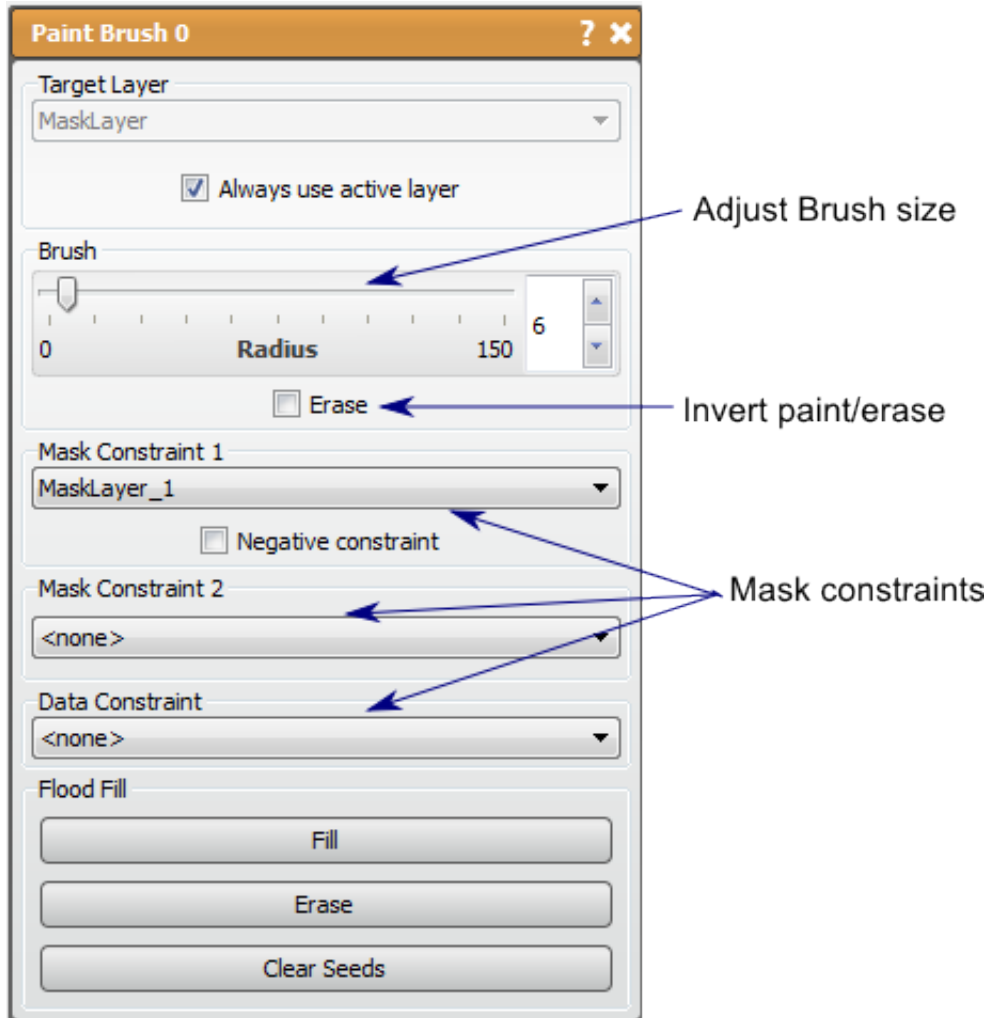
You can constrain the Paint Brush with other mask layers (up to two) or with a data layer and threshold limits (this is the same as creating a threshold mask layer and using it as a constraint) to limit the pixels that can be painted.

There is a *flood fill and erase* function which will fill an area in the slice that is completely surrounded with the mask layer, or delete a connected region. If no seed points are used with these functions, the entire slice will be filled or erased. Seed points are chosen with **alt+left mouse** button.

It should be noted that since this tool uses the scroll function to change the Paint Brush size, it will not also change the slice in the viewer. To change the slice use the arrow **up/down** or **shift+scroll**.

Table 2.12: Keyboard and Mouse Actions in the for the Paint Brush Tool

Action	Function
left mouse	Paint
right mouse	Erase
scroll up/down	increase/decrease brush size
alt+left mouse	add seed point
alt+right mouse	remove seed point
C	clear seed points
F	Paint flood fill
E	Erase flood fill



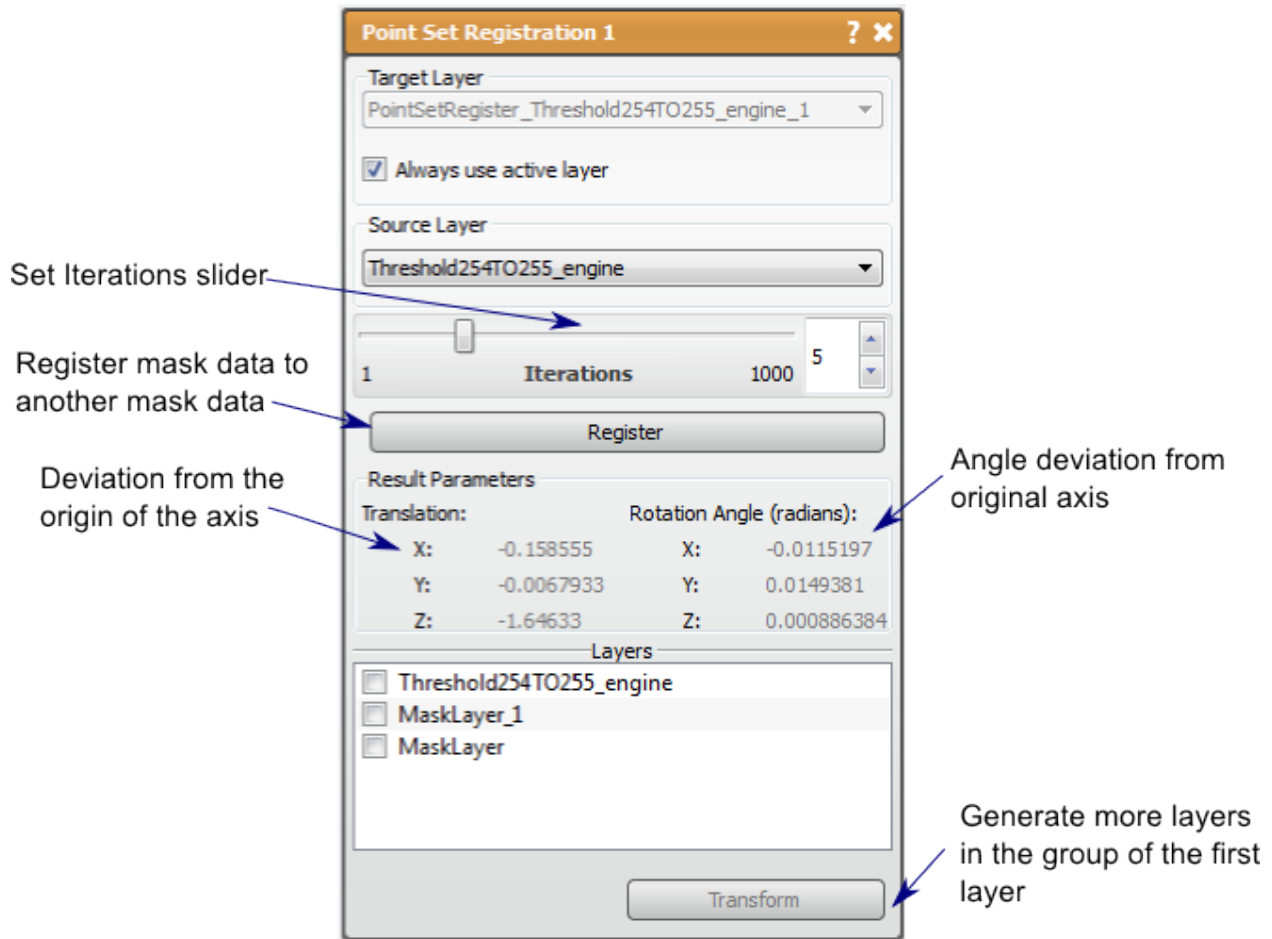
Point Set Register

This tool will register one mask layer to another.

Detailed Description

The Point Set Registration tool will register one mask layer to another mask layer. This is currently the only tool or filter that will take inputs from two different layer groups as the main purpose of this tool is to allow the user to map a segmentation from one grid to another with a similar mask layer. The tool attempts to register the entire mask data to another mask data using the iterative closest points method (rigid registration) and will save and display the transform generated for use on the other layers in the registered layer's group. Since this is an iterative method, the user can set the number of iterations.

To run this registration, choose for the first input the mask layer in the destination frame (to register to) and the mask to change (register) as the second input. The idea is that these would be in separate layer groups. Clicking the register button will create a new layer in the group of the first input and the transformation will be saved and displayed (translation and rotation). There will be a list of other mask layers from the second input's group with which the user can select the layer on which to perform the saved transform. Upon pressing transform, more layers will be generated in the group of the first input layer.



Polyline

The Polyline tool allows the user to define a region of the layer to modify by adding or erasing the data in the region defined by the polyline.

Detailed Description

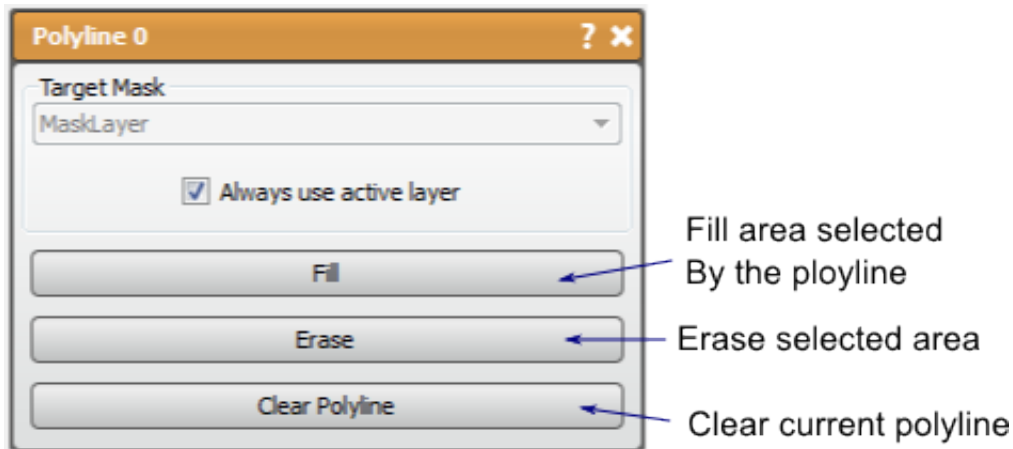
The Polyline tool is a tool that allows manual creating and editing of mask layers. The polyline tool allows the user to define a region of the layer to modify by adding or erasing the data in the region defined by the polyline. This tool can only manipulate existing mask layers (can be empty).

The Polyline is created by clicking with the **left mouse** on the 2D viewer. The polyline can be modified in any plane, but will only fill/erase in the active plane. The polyline will try to order the new point in between two existing points to make as smooth a shape as possible. This means that the user should try to place a few points around the edge of the shape desired to create a coarse shape and then fill in the details as needed. points can be moved with the **left mouse** button (pointer will turn into a hand) and deleted with the **right mouse** button. The whole polyline can be moved with the **shift+left mouse** on one of the points.

Once the desired region is designated by the Polyline, the user can fill the area or delete the mask data in the area. This tools then is like the Paint Brush tool, where the brush can be shaped in any way. The polyline will remain as the slice changes, so that the user can modify the existing polyline instead of creating a new one. If desired, a new polyline can be created by clearing the polyline and starting fresh.

Table 2.13: Keyboard and Mouse Actions in the for the Polyline Tool

Action	Function
left mouse	add or move point
right mouse	erase point
shift+left mouse	move Polyline (on one of the points)
F	Paint flood fill
E	Erase flood fill



Resample

This tool allows the user to resample one or more volumes to change the resolution.

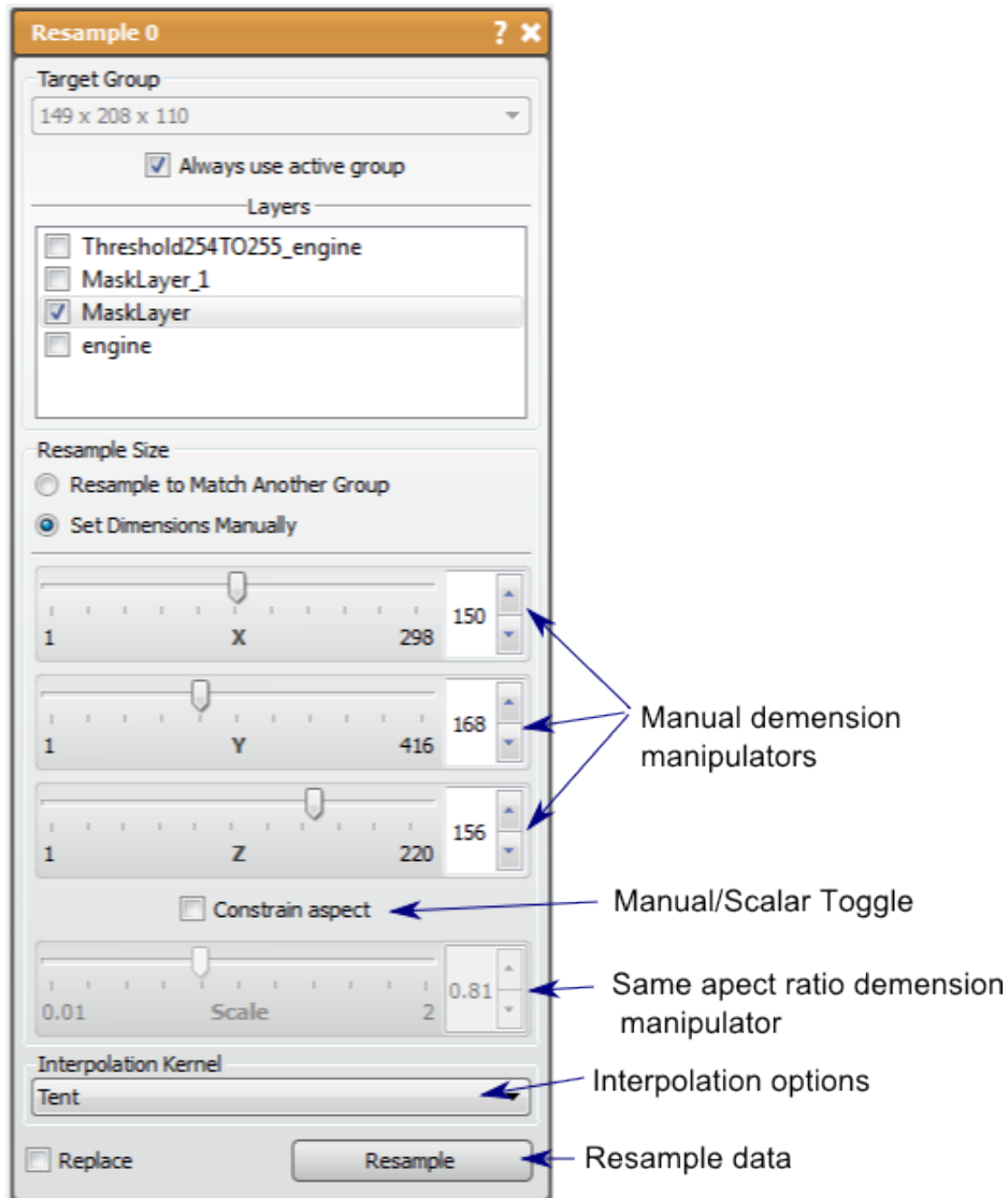
Detailed Description

This tool can be used on both mask and data layers.

The resampling can be set manually using the sliders, or by trying to match the element number of another layer group. To manually choose the resampling, there are options to resample with the same ratio in every direction (check *Constrain Aspect* option) or to resample each direction independently (must uncheck *Constrain Aspect* option to enable). To resample to match another group, you must have another layer group loaded. You may choose the desired group from the drop down menu and the padding value.

Interpolation of the data is often needed when resampling, especially with image data. There are several options for interpolation in the Resample tool: Box, Tent, Cubic (Catmull-Rom and B-spline), Quartic, and Gaussian which are provided by the Teem library, and Linear, B-spline and Nearest Neighbor from the ITK library. You may choose the interpolation method in the drop down menu near the bottom of the tool window.

Resampling a layer to match the geometry of another group can also be done by clicking (left mouse button) on a layer while holding down the Shift key, and dragging the layer to the destination group.

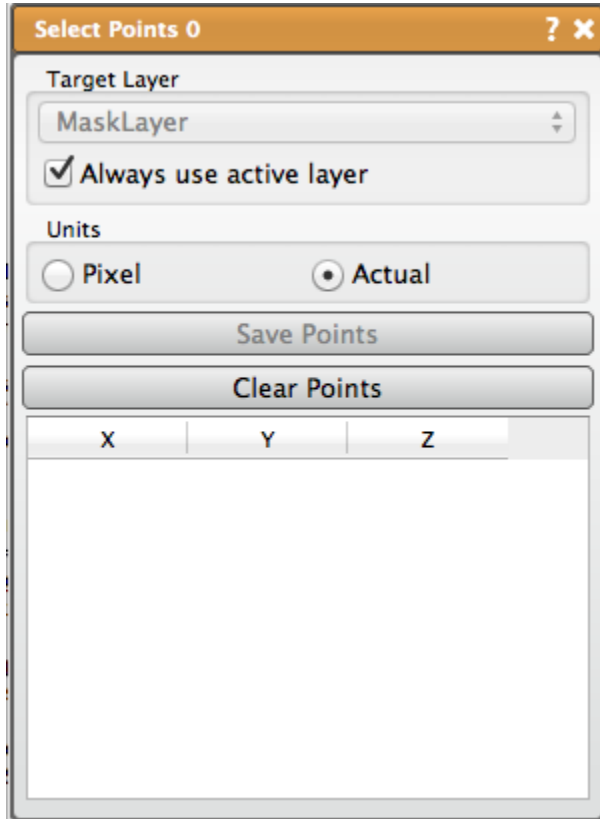


Select Points

Select and export a set of points.

Detailed Description

This tool selects a set of points either in pixel or actual (world) coordinates and save as a text file. It works with either data or mask layers. Clicking on a layer in one of the 2D views places a point. Secondary (*right mouse* button) click deletes a point.



Speedline

The Speedline tool is similar to the Polyline tool in that you use lines to identify a region to segment, but the Speedline tool attempts to alter that path between set points to find edges in the image data.

Detailed Description

The Speedline tool, which is based on the [Live-Wire algorithm](#), is similar to the Polyline tool in that you use lines to identify a region to segment, but the Speedline tool attempts to alter that path between set points to find edges in the image data. The Speedline tool computes the shortest path from a source to a target pixel using a cost function that uses the following parameters, which can be found in the tool interface: *Gradient Magnitude* (image gradient), *Zero Cross* (image zero-crossings), *Gradient Direction* (boundary smoothness constraint). The *Face Connectiveness* option is used to toggle the use of a local neighborhood. The *Image Spacing* option is used to toggle the use of a neighborhood weighting scaling factor.

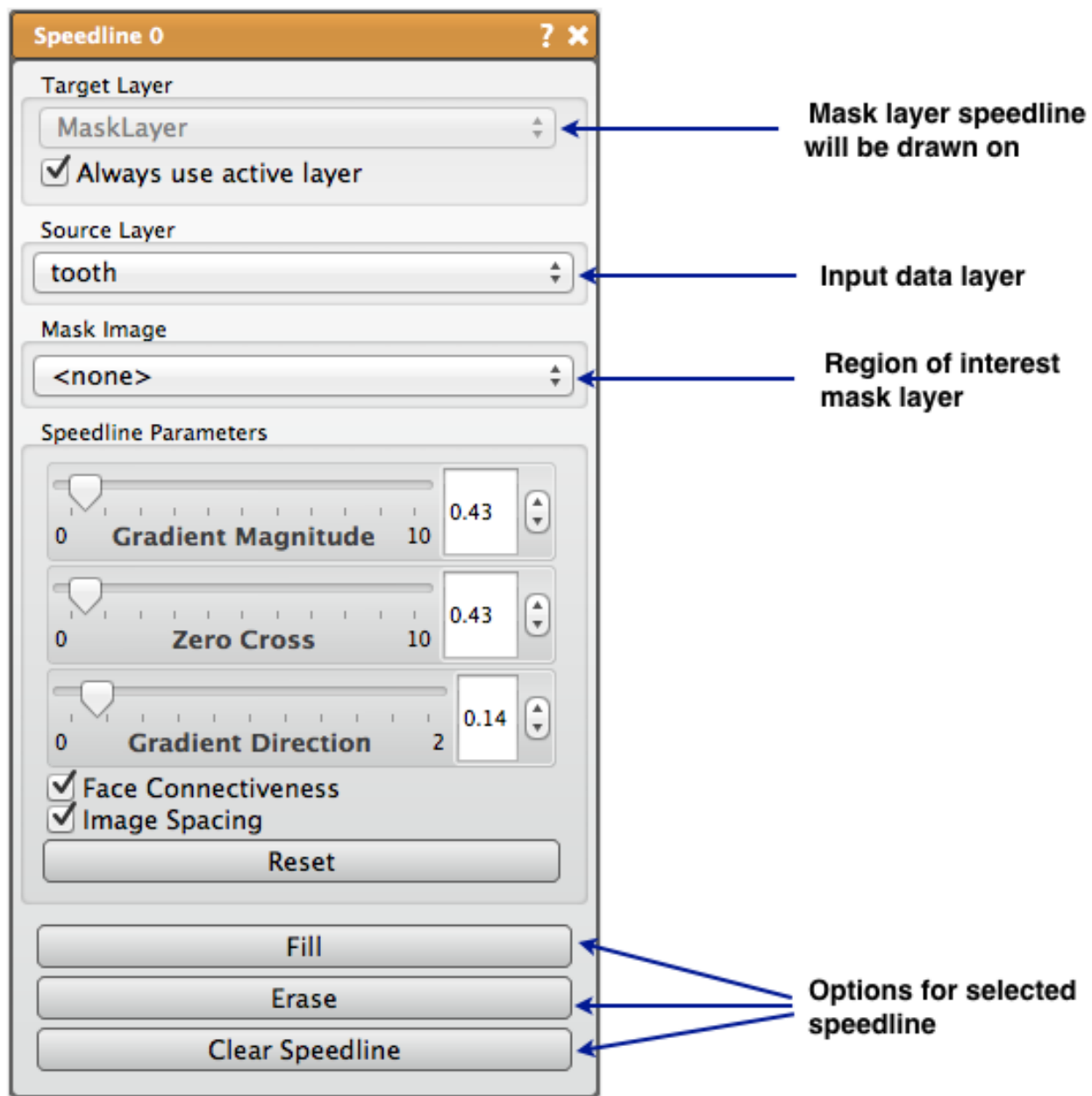
The primary input of the Speedline tool is a mask layer which will be altered. The second input is a data volume. Smoothing the input data volume is recommended before using the Speedline tool. An additional mask input can be used to constrain the Speedline to a given region of interest. Now the Speedline selection can begin.

The selection of the the Speedline is virtually identical to the selection of a Polyline. The user will place points and the software will create a shape with those points. In the Speedline case, a new point will be ordered to create as smooth a shape as possible, and the path of the connection will attempt to follow high intensity regions of the speed image (edges of the original image). Once the desired shape is created, the user can fill the region, or erase it to the mask layer input.

It should be noted that if the active slice changes, the points will not change, but the connections will still attempt to follow the edge regions of the images. This can therefore be used to segment multiple slices quickly if the slices do not dramatically change. The Speedline points can be cleared or reset in the tool window. The user can also set the iterations and termination value to use in the path selection algorithm.

Table 2.14: Keyboard and Mouse Actions for the Speedline Tool

Action	Function
left mouse	add or move point
right mouse	erase point
F	Paint flood fill
E	Erase flood fill
C	Clear Seed points



Threshold

This tool generates a mask label from an image data volume and the value range to select.

Detailed Description

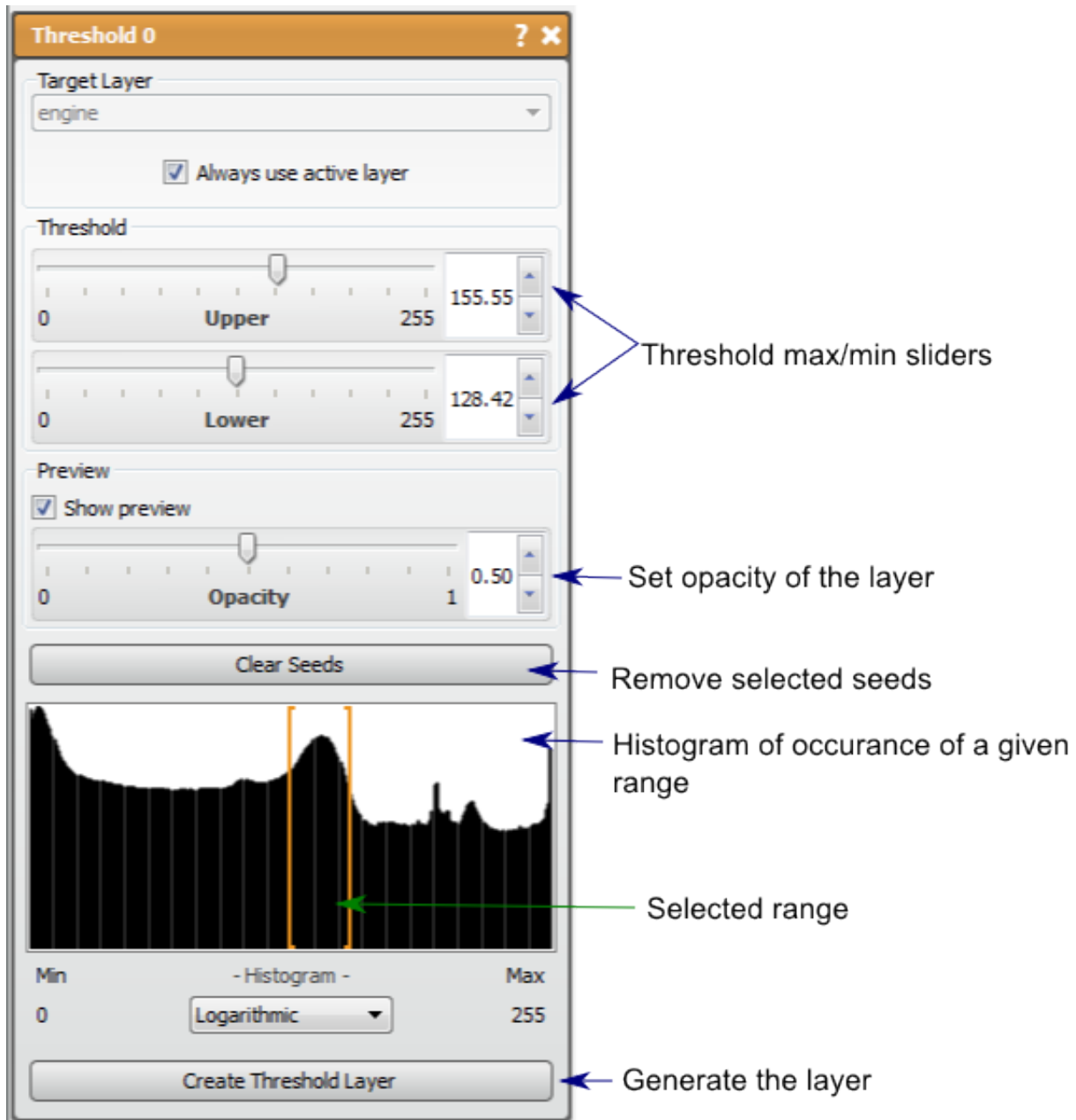
The Threshold tool generates a mask label from an image data volume and the value range to select. The mask layer created will be a segmentation of the regions in the image data that have intensities in the range specified. This is one of the most basic semi-automatic tools used in segmentation as it is fast but can also separate contrasted regions very effectively.

The threshold range can be set in two ways: the slider provided in the tool window will set the maximum and minimum for thresholding or seed points can be placed throughout the volume and the values at those points will set the range to threshold. A preview of the threshold regions will be displayed in the 2D viewers (can be disabled in the tool window). The opacity of this preview can be set using the provided slider. The threshold range will also be shown with the image histogram in the tool window. The threshold range and histogram can also be based on a logarithmic scale (more homogenous) or linear scale.

Using the Threshold tool can provide a very good rough segmentation of the region of interest, but may require some editing. The Paint Brush and Polyline tools are manual ways to remove excess regions in the mask, but the user should also try other mask filters such as Connected Component and Connected Component Size (and another Threshold) to remove stray regions of selected noise.

Table 2.15: Keyboard and Mouse Actions in the for the Threshold Tool

Action	Function
M	toggle visibility of the preview mask



Transform

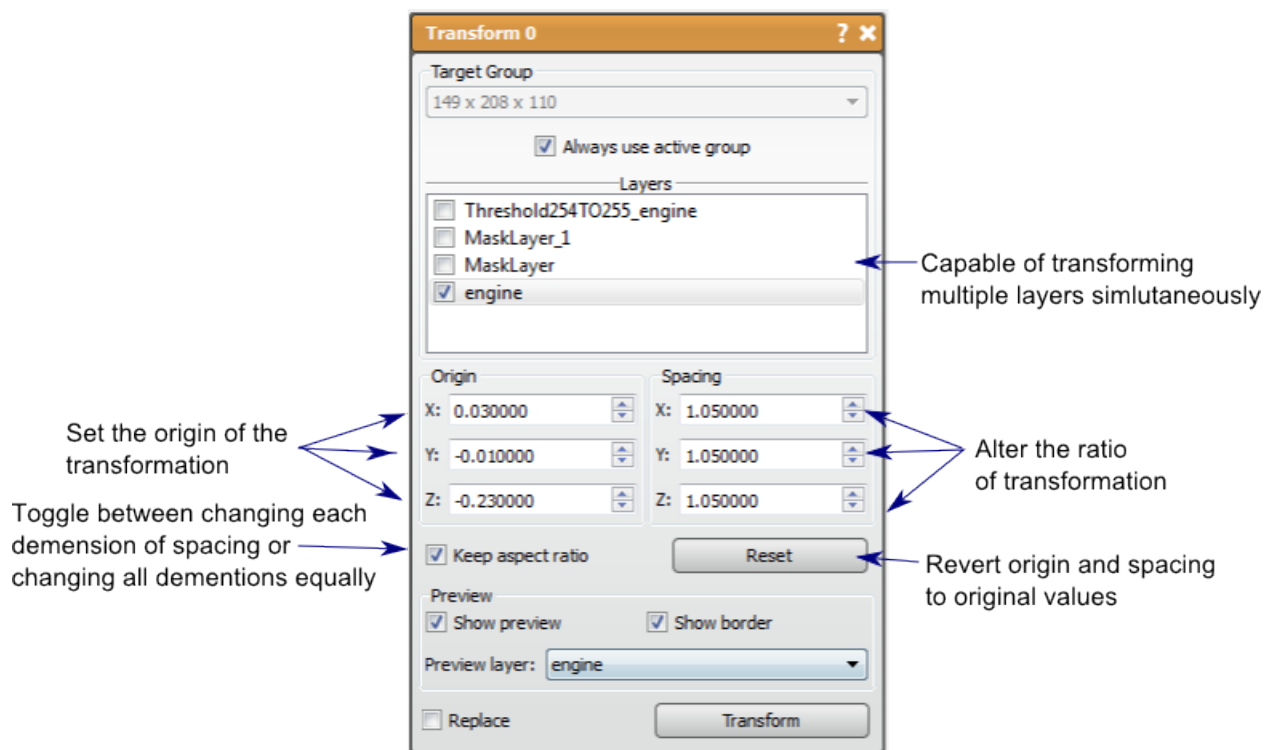
This tool can be used to translate and scale one or more layers.

Detailed Description

The Transform tool can be used to translate and scale one or more layers. This tool can be used on both mask layers and image data layers. This tool allows the user to change the origin (translate) and the spacing (scale) of multiple layers in a group. A key use of this tool is to manually transform one layer to match the origin and spacing of a different group, thus including the layer in that group.

The origin and spacing can be changed in the fields provided, or editing the box widget in the 2D viewer. To change the box widget, **left mouse** click and drag in the middle to move the origin, **left mouse** click and drag on the edge to change the spacing. The aspect ratio of the spacing can be fixed (*Keep Aspect Ratio* option) or the spacing dimension can change independently. The Transform parameters can also be reset so that the original origin and spacing are displayed.

A preview of one of the checked layers from the list will be shown in the 2D viewer, as well as the box widget. Both these can be hidden by unchecking the *show preview* and *show border* options, respectively. If more than one layer is checked, you may choose which layer to preview with the drop down menu provided.



2.3 Seg3D Tutorial

2.3.1 Overview

This tutorial describes step by step instructions to segment Ascending and Descending Aorta in cardiac MRI and describes the segmentation of skull, gray and white matter in a head dataset.

Software Requirements

Seg3D

Seg3D is distributed as a binary download for Linux, Windows, and OS X. Please visit the [SCI software portal](#) to download the latest Seg3D. Any version of 2.0 or higher will do.

Required Datasets

This tutorial uses two datasets. The first one is a set of two 3D scans, one is an anatomical MRI of the atria and the second one is a delayed enhanced MR image of the same region. Both images have been registered. The second data is a brain data set that contains a preoperative MR image of the brain and a post surgery CT image of a pediatric patient.

The datasets are available in the SCIRunData zip files. Please visit [SCIRun Data](#) to download the datasets. The datasets for this tutorial are inside the Seg3D directory. The direct linked to the zipped SCIRun Datasets can be found [here](#).

2.3.2 Basic Orientation

Starting Seg3D

When you start Seg3D, you will be greeted by the welcome screen (Fig. 2.28). As you will see, you have some options: you may load a recent project from the list (likely to be empty), open a project that you have saved on your machine, start a new project, take a quick look at a single segmentation or image file, or you may quit Seg3D if you have changed your mind.

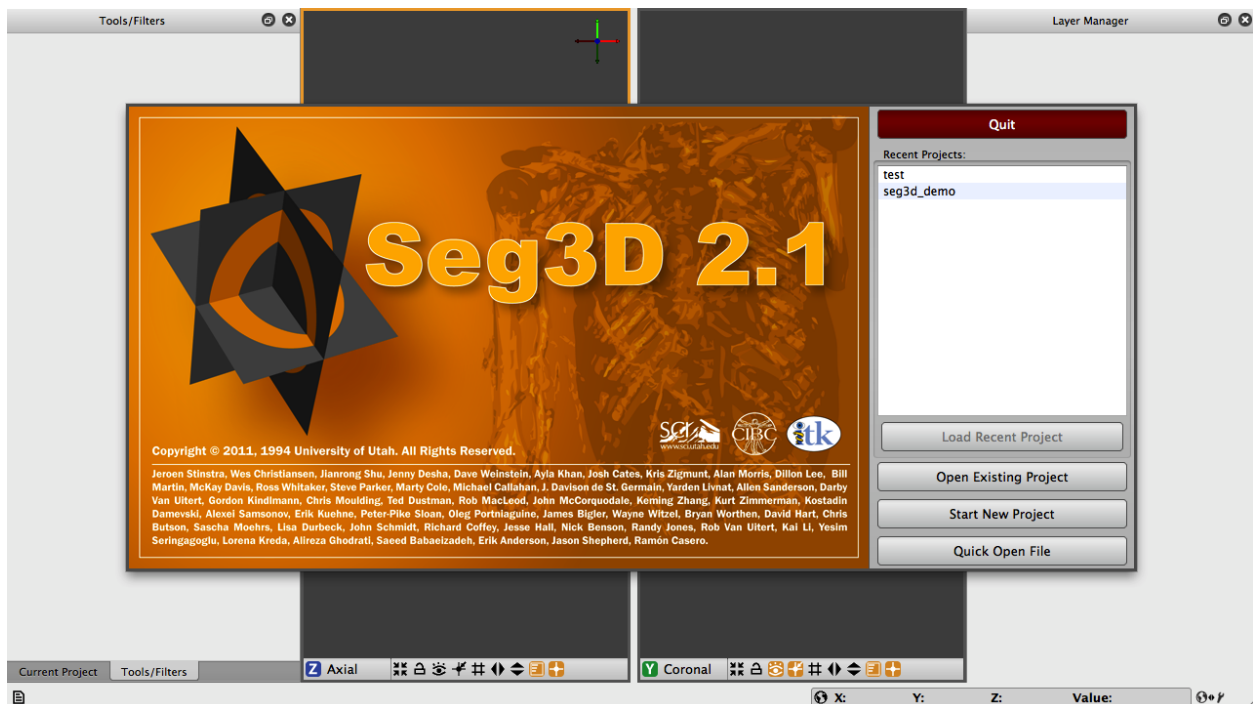


Fig. 2.28: Seg3D welcome screen.

Seg3D handles your data mainly in the form of projects which are similar to those used in other applications. A project consists of a group of files and with this set up, Seg3D is able to track and save your data that you are working on,

the settings that you are using, and the tools and filters that you are using. This is very useful, especially as your segmentations become larger and more complicated. For now, we will begin by starting a new project. Once you choose to open a new project, you will be asked to choose a name and location for the project. Test_1 will do for a name and the default location will be a directory that was created during installation called Seg3D-Projects. You can use this directory to store future projects.

Loading A Dataset

In order to get a good feel for the Seg3D program, one needs to load a dataset. Please open up the file menu and select the **Import Layer From Single File...** menu item as indicated in Fig. 2.29. This opens up a file browser. Browse to the Heart_DataSet directory and select this one to load the DEMRI.NRRD file.

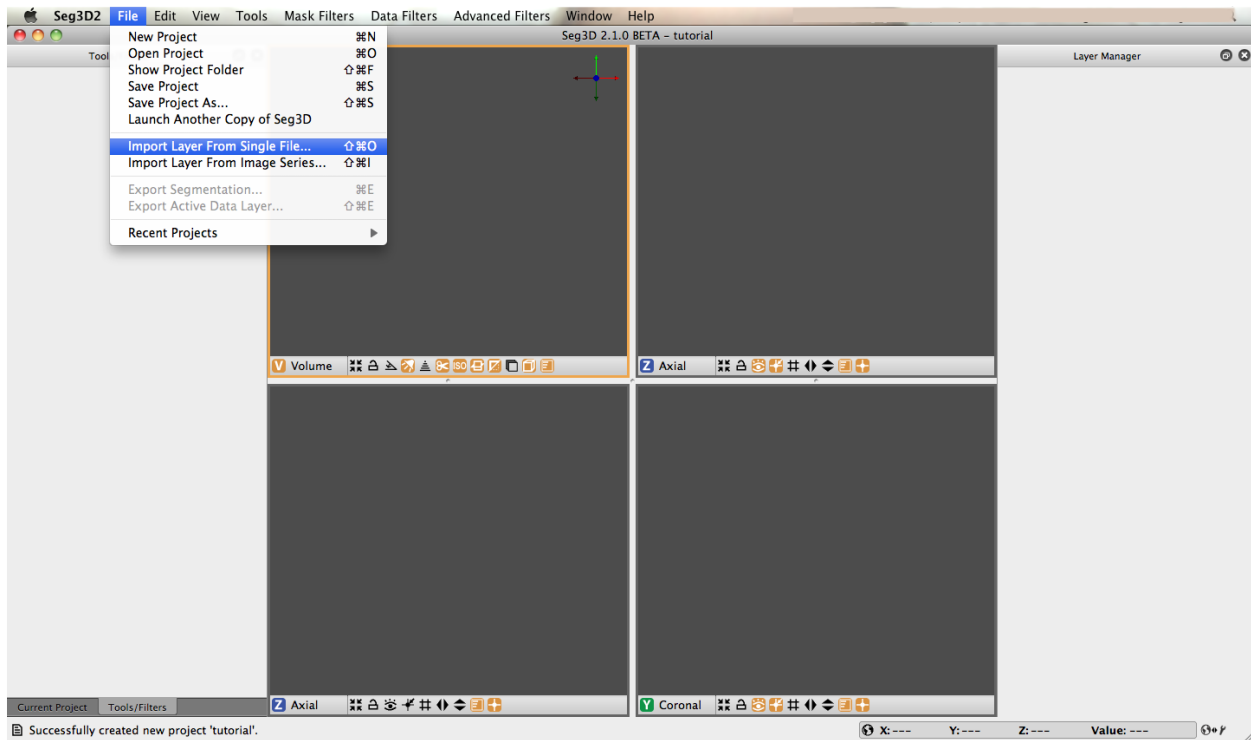


Fig. 2.29: Loading a volume.

The layer importer widget will then appear (Fig. 2.30). This widget is for distinguishing how the data from the file is read and the widget is especially needed when loading previously saved label masks. For this dataset, choose the only available option, which is **Data Volume**. This will load the file as a gray scale image volume.

Note: we converted the datasets for this demo into the *NRRD* dataset format as it does not contain any patient information. For more information on NRRD filetypes, refer to the [NRRD Format](#). Seg3D reads *DICOM* files as well and any image file format supported by the Insight ToolKit, as well as Matlab files. For additional supported filetypes, view the filetype drop-down menu that appears while browsing to open a file. The browser decides which file format to use based on the extension of the file.

Once the file has loaded one can see the data being displayed in the 3 slice views and the 3D viewer. The layout of the Seg3D program is highlighted in the Fig. 2.31. On the top left one has a tool menu that will show the options of the currently loaded tool and is currently empty. On the right one can see the layer menu. Each layer in Seg3D is a separate volume and the order in the Layer Editor depicts the order in which datasets are being shown. In the center one has the three orthogonal views and one 3D viewer and to the lower left one can browse the position of the active dataset and sample the value.

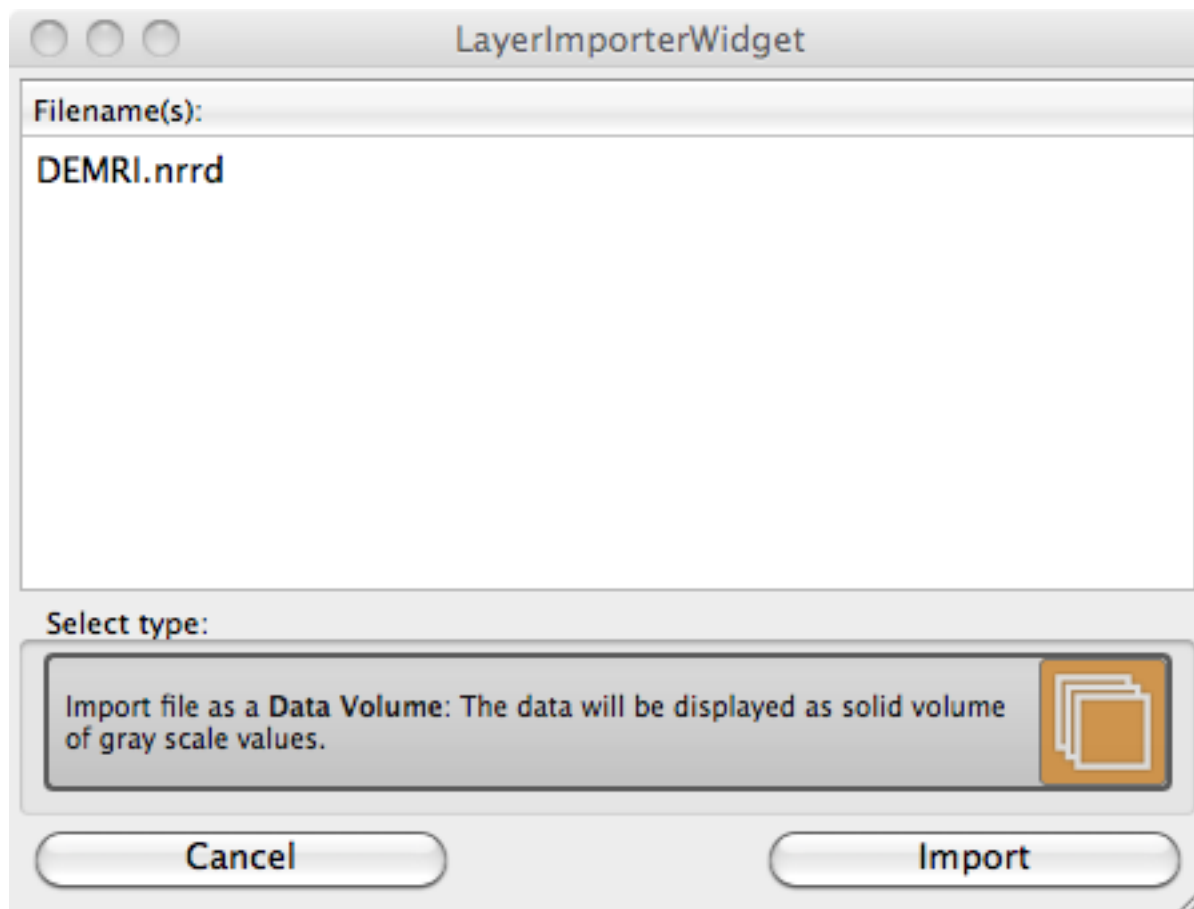


Fig. 2.30: Chose type of data to load.

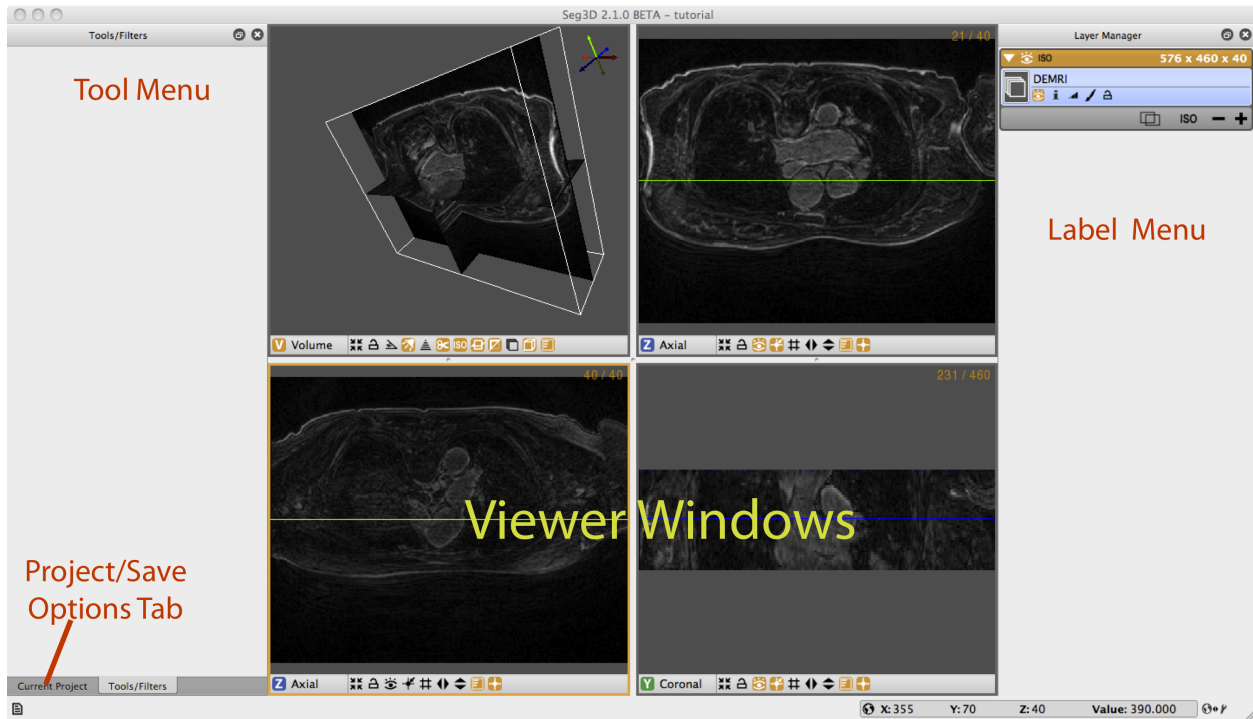


Fig. 2.31: The main Seg3D window.

Some tools require data to be represented in a histogram, which would be shown in the tool menu. The histogram shows pixel intensities in a cross section of the image, and the data from the histograms is used for image filtering. The data in the histogram can be represented in either a logarithmic or a linear format, with the linear format showing linear values, which often results in more dramatic intensity gradients, and the logarithmic format showing an alternative, with lower intensities receiving a greater emphasis.

Navigation

Note: These operations are based on a setup that involves a three-button mouse. For other configurations, see **Keyboard Shortcuts** in the Help menu (Fig. 2.32).

Brightness

Move the mouse up and down while pressing left mouse button (only when a volume layer is selected).

Contrast

Move the mouse left and right while pressing left mouse button (only when a volume layer is selected).

Pan Volume

Press shift and move volume while clicking left mouse button (Volume view only).

Zoom

Move the mouse up and down while pressing the right mouse button (Volume view only).

Rotate

Move the mouse while pressing the middle mouse button in the 3D viewer (Volume view only).

Move slice orthogonal slices

Move the mouse while pressing the middle mouse button in the 2D view windows. This will change the location of the two colored lines, which represent the location of the two other slice windows.

Viewer Shortcuts:		General Shortcuts:	
Navigation & Control –		File –	
Toggle Lock Viewer:	L	New Project:	COMMAND+N
Toggle Picking State:	P	Open Project:	COMMAND+O
Increase visible slice number by 1:	↑ or >	Show Project Folder:	COMMAND+SHIFT+F
Decrease visible slice number by 1:	↓ or <	Save Project:	COMMAND+S
Increase visible slice number by N:	SHIFT + ↑ or >	Save Project As:	COMMAND+SHIFT+S
Decrease visible slice number by N:	SHIFT + ↓ or <	Import Layer From Single File:	COMMAND+SHIFT+O
Make layer above active:	←	Import Volume From Image Series:	COMMAND+SHIFT+I
Make layer below active:	→	Export Segmentation:	COMMAND+E
Compute Isosurface:	O	Export Active Data Layer:	COMMAND+SHIFT+E
Viewer Modes –		Edit –	
Set viewer to show the Volume View:	V	Undo:	COMMAND+Z
Set viewer to show the X Axis:	X	Redo:	COMMAND+SHIFT+Z
Set viewer to show the Y Axis:	Y	Copy Mask Slice:	COMMAND+C
Set viewer to show the Z Axis:	Z	Paste Mask Slice:	COMMAND+V
Visibility –		Punch Through:	COMMAND+P
Toggle active layer visibility:	SPACE	View –	
Toggle volume lighting visibility:	H	Full Screen:	COMMAND+F
Toggle isosurface visibility:	I	Only One Viewer:	ALT+0
Toggle overlay visibility:	T	One and One:	ALT+1
Toggle grid visibility:	G	One and Two:	ALT+2
Volume View –		One and Three:	ALT+3
Pan:	Left Mouse Button	Two and Two:	ALT+4
Zoom:	Right Mouse Button	Active Tool Shortcuts:	
Rotate:	Middle Mouse Button		
Volume Rendering –			
Add/Move Transfer Function Control Point:	Left Mouse Button		
Delete Transfer Function Control Point:	Right Mouse Button		
Move Transfer Function Feature:	SHIFT + Left Mouse Button		

Fig. 2.32: The keyboard shortcuts as found in the help menu.

Move slice depth

Use the mouse wheel to move to the next slice. Or use the up and down arrow keys, or the < and > keys, to navigate through a slice.

Painting

In painting mode the mouse wheel is the size of the brush, hence to move through slices use the up and down arrow keys, or the < and > key.

Plane visibility

Press space to switch off viewing the image and space again to switch it on (all views, masks remain visible).

Navigate through the data and rotate the 3D image.

Changing Views

To alter the views that are displayed in the window. Press on the Red, Green, or Blue tab at the top of each Viewer to show a different viewer. By clicking on the labels one loops through the four main modes of each window.

In the **Views** menu one can further specify the layout of the views. One can choose from:

Only one viewer

Just showing one window.

One and One

Two viewers side by side.

One and Two

Make one big viewer and two smaller ones.

One and Three

The default

Two by Two

Four views of equal size.

Two and Three

Make two viewers big and three smaller.

Three and Three

Make six viewers.

There is also an option for a full screen mode which will fill the entire computer screen with the Seg3D window. The top menu will pop down when the cursor is moved to the top of the screen.

Loading a second dataset

Open the **Import Layer From Single File...** menu once more and now load the Angio.NRRD dataset by choosing **Data Volume**. After loading this dataset, two datasets are present in the menu on the right. You can see that because the data sets are in a different group, *i.e.*, they each have an orange header, the volumes are slightly different. By clicking on the *i* button for each volume you can see that the origins are slightly different. You may correct this using the **Transform** tool. With this tool you can change the origin of one volume to match the other. The corrected volume will then join the other into a single group. This step is not needed for the purposes of this tutorial, but tools and filters that use two layers as inputs cannot use layers of different groups.

Press on the eye to show which layer is visible. As volume layers are full data layers they completely clutter the underlying image. Switch on the visibility (using the eye) of both layers and toggle the visibility of the top most layer. The top most is the one found on top of the menu. The blue highlight set the selected layer. Select the top layer and press space and press space again (put the cursor in the Viewer Window). This option can be used to quickly view the difference between layers.

2.3.3 Simple Operator Based Segmentation Strategies

DEMRI Dataset

In this section we will use the paint and polyline tools to segment a simple structure, the aorta, from the DEMRI dataset. Delayed enhancement MRI (DEMRI) makes use of the contrast enhancement that gadolinium (Gd) provides in MR imaging. In this scan Gd was injected via IV into the patient's venous system. As the Gd washes through the system it enters and exits tissues at variable rates. In this case, 15 minutes after administration, the Gd has washed into and out of most tissue. Enhanced tissues that continue to sequester Gd are assumed to have disrupted, or altered vascular perfusion. This method has been used to detect cardiac ischemia, infarct, and scarring. This particular DEMRI dataset was used to evaluate post-operative scarring in the left atrium. To get started, we will be looking at the ascending and descending aorta.

Label Masks

Seg3D has two types of layers that can be operated on by different tools and filters. These are volume and label masks. Volumes are image data that has been loaded in, or produced by one of the filters. The DEMRI and Angio datasets loaded in the first section are volumes, and appear left aligned on the Layer Menu. If you select the + button below either of these volumes a new layer with the name 'Mask Layer' appears indented above the volume. This is a label mask, and you may rename it anytime by clicking on the name of the layer. The label mask displays the segmented regions of interest as a colored opaque layer. When using any tool or feature it is important to know whether it operates on a volume, label mask, or both.

Paint Tool

The paint and polyline tools allow the user to directly select regions of interest for segmentation. The paint tool presents as a circle icon of adjustable radius that can select regions of interest by holding down the left mouse button and moving the icon over the desired object.

Begin by selecting Paint Brush under Tools on the menu bar. When the circular icon appears, experiment with the tool by moving it over the three orthogonal image planes. You will notice that the icon is elliptical over the Coronal and Sagittal planes and is circular over the Axial plane. Click on the i button under DEMRI on the Layer Menu and information about the volume will open. Next to Spacing is listed the voxel dimensions (0.62, 0.62, 2.5). With respect to the Axial plane this means the slice thickness is nearly 4X the size of X-Y spacing. When using other features in Seg3D it is important to remember that the anisotropy of your data is preserved.

Prepare to segment by clicking on the + widget under DEMRI on the Layer Menu to create a label mask. Begin segmentation of the aorta by navigating the Paint icon over the **Descending Aorta** on the axial plane. Adjust the diameter of the tool with the scroll wheel, or by holding the right mouse button until it just fills the lumen of the aorta. Once you have positioned and sized the icon click the left mouse button. The circle icon should then fill with the same color of the selected label mask. Proceed to the next few slices painting the lumen of each one (Fig. 2.33). Next, create a new label mask, and expand the diameter of the Paint icon so it encircles the outer diameter, or superficial surface of the aorta. Now paint the superficial surface of the same slices. It is possible to erase painting on the label mask by holding the right mouse button and moving the icon over the region to be erased.

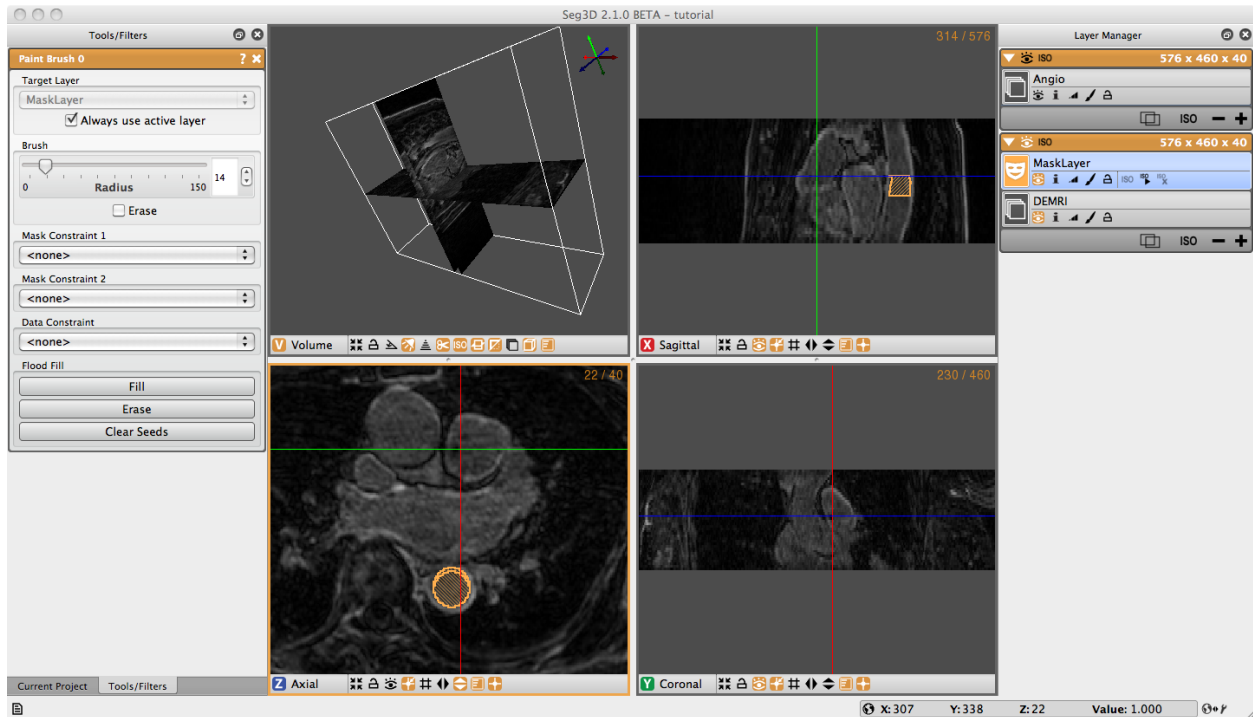


Fig. 2.33: Painting the lumen of the descending aorta.

Polyline Tool

The Polyline Tool may be better suited for segmentation of complex shapes. The polyline tool is operated by selecting points, with the left mouse button, around the perimeter of the region of interest. Each point added is connected to the point before it and to the first point selected making a closed polygon. Both tools display the selected regions of interest in label masks.

Open the Polyline Tool under the Tools menu on the menu bar. Without changing the label mask, begin clicking points around the superficial surface of the **Ascending Aorta** (Fig. 2.34). If you are not satisfied with the location of a point you can move it by moving your cursor over the point, when the cursor turns into a had or a thicker cross, you may move the point by left clicking on it and dragging it to the desired location. Or, you can simply delete the point by right clicking on the point when the cursor changes. The polyline tool will automatically try to place the point in the order that completes the smoothest shape, so when your are outlining a concave shape, make a convex shape, then add more points to make it concave. Once you have traced the superficial surface all the way around click fill on the Tool Menu, or simply press F on the keyboard. The region surrounded by the polyline should fill with the color of the selected label mask. Navigate to the next slice, and notice that the polyline does not disappear. If the polyline still matches the region of interest, simply hit F. To adjust the polyline hold shift and left click on one point and drag to move the entire polyline, or move individual points as described before. If you want to start over hit Clear Polyline button in the tool options on the left. It is also possible to erase regions of the label mask with the Polyline tool by creating a polyline around the region to be erased and selecting erase on the Tool Menu. After creating the superficial surface label masks hide the masks by clicking on the eye widget, select the lumen label mask, and use the Polyline tool to create lumen masks on the **Ascending Aorta** (Fig. 2.35).

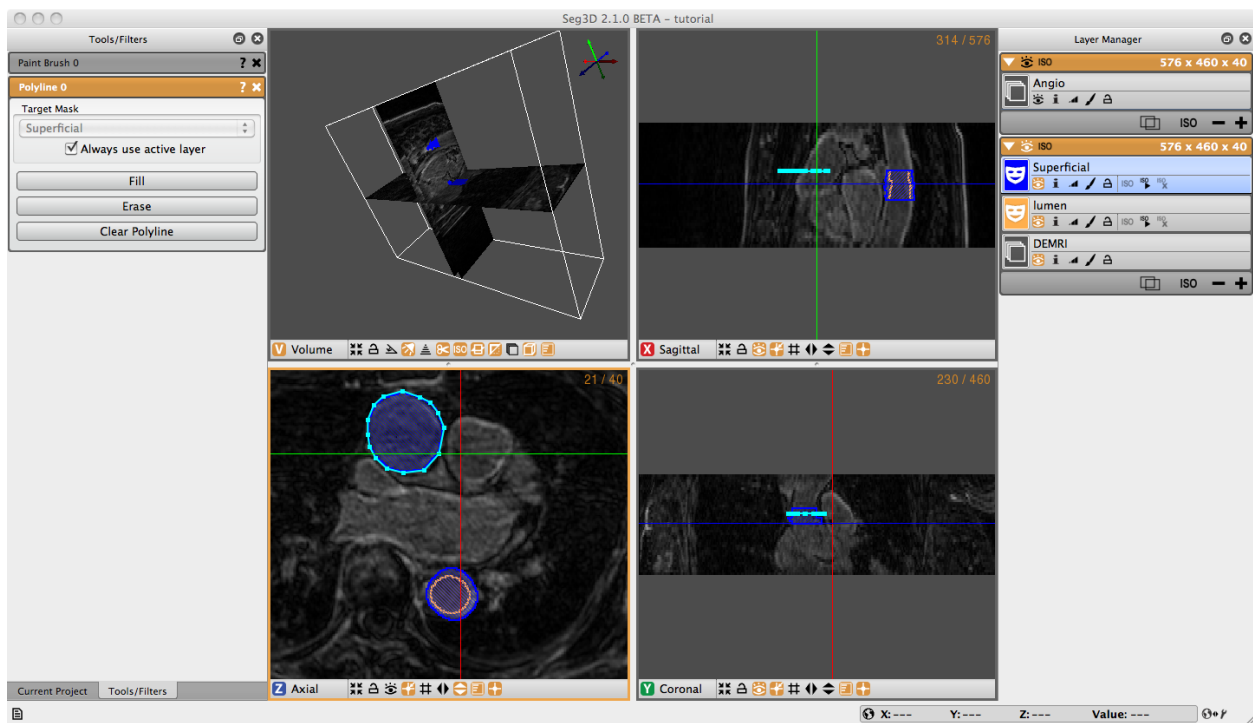


Fig. 2.34: Polyline for masking the superficial surface of the ascending aorta.

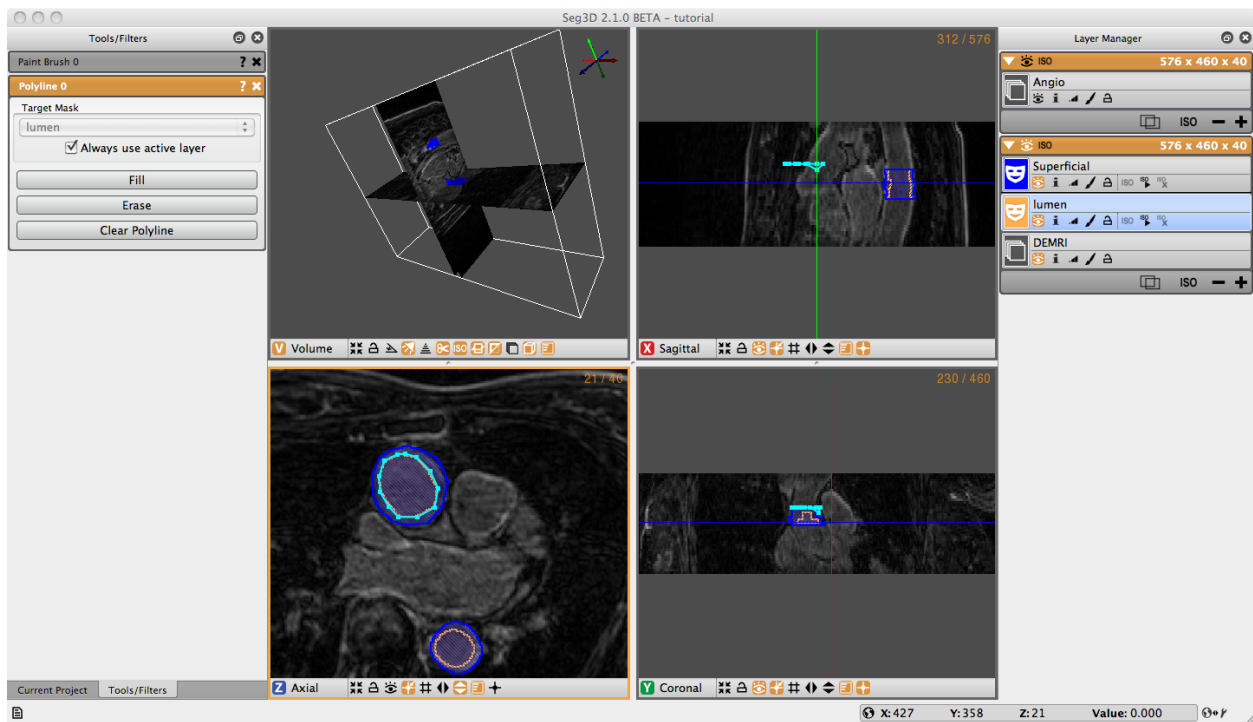


Fig. 2.35: Polyline for masking the lumen of the ascending aorta.

Logical Operators

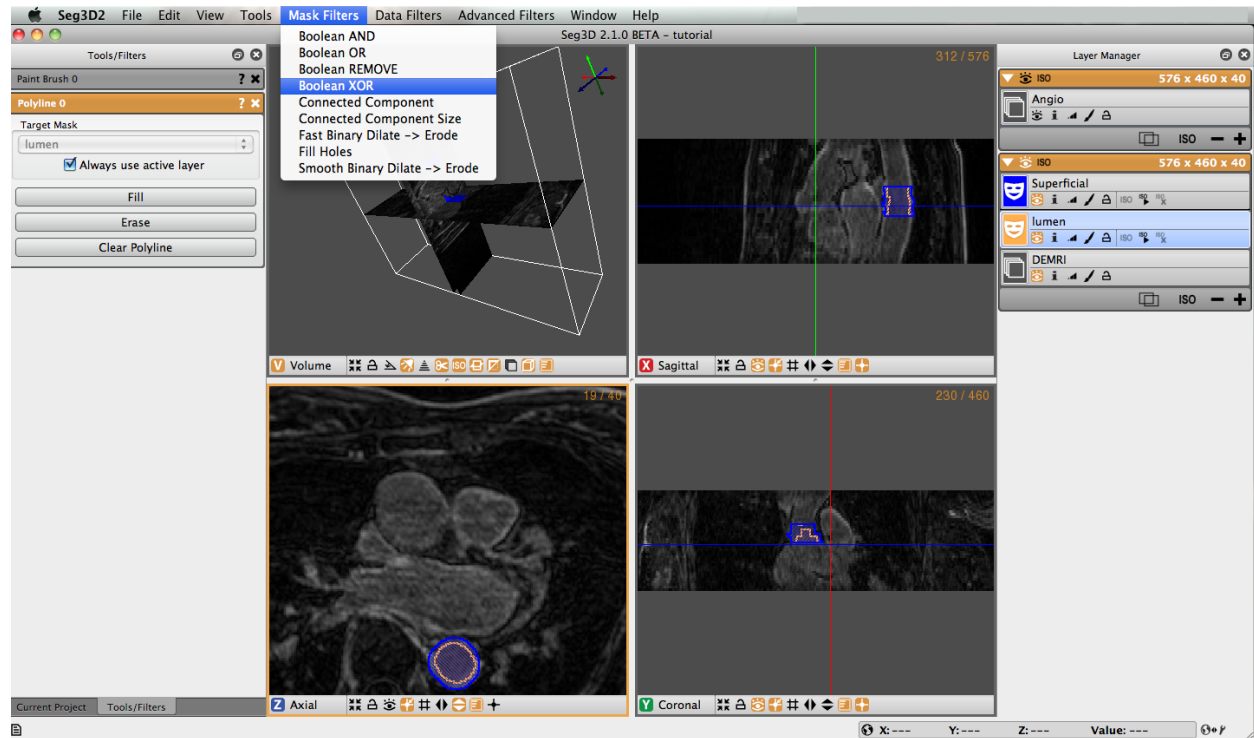


Fig. 2.36: Creating label mask of region of interest with logical operators.

Once you have masked the lumen and superficial surface of both the ascending and descending aorta we can use logical operators to mask the real region of interest, the aorta wall. Under Mask Filters on the menu bar, select Boolean Xor. Highlight either of the label masks. You will notice that the name of the highlighted filter will appear in the target layer field. You can change this by unchecking the 'Always use active layer' option. Choose the other label mask in the Mask Layer option by selecting it from the list of eligible layers, or by holding down shift and clicking on the face color image of desired layer and dragging it to the option list. Click on the Run Filter button, this will create a new label mask that only covers the aorta wall (Fig. 2.36). The resulting layer is shown in Fig. 2.37. Notice there is an option called Replace on the bottom of the tool menu. Most of the filters have this option. It will replace the target layer with the result of the filter. This is very useful when you do not need the unmodified layer after you run the filter.

Isosurface

To create a three dimensional isosurface of the Xor label mask, simply click the create isosurface icon ('ISO ') for the layer. It may sometimes be necessary to delete an isosurface that has already been created if you run a filter with the replace option ('ISO x'). Once Xor is the only layer highlighted, click the create isosurface icon on the Xor layer window. Now the 3D View in the upper right corner should display the 3D isosurface (shown in figure above). Navigate around the volume with the normal zoom and pan volume short keys. Holding the alt/option key and clicking (or middle click) on the volume will allow you to rotate the object. Experiment with the widgets in the bottom of the 3D viewer to turn on and off various elements of the display.

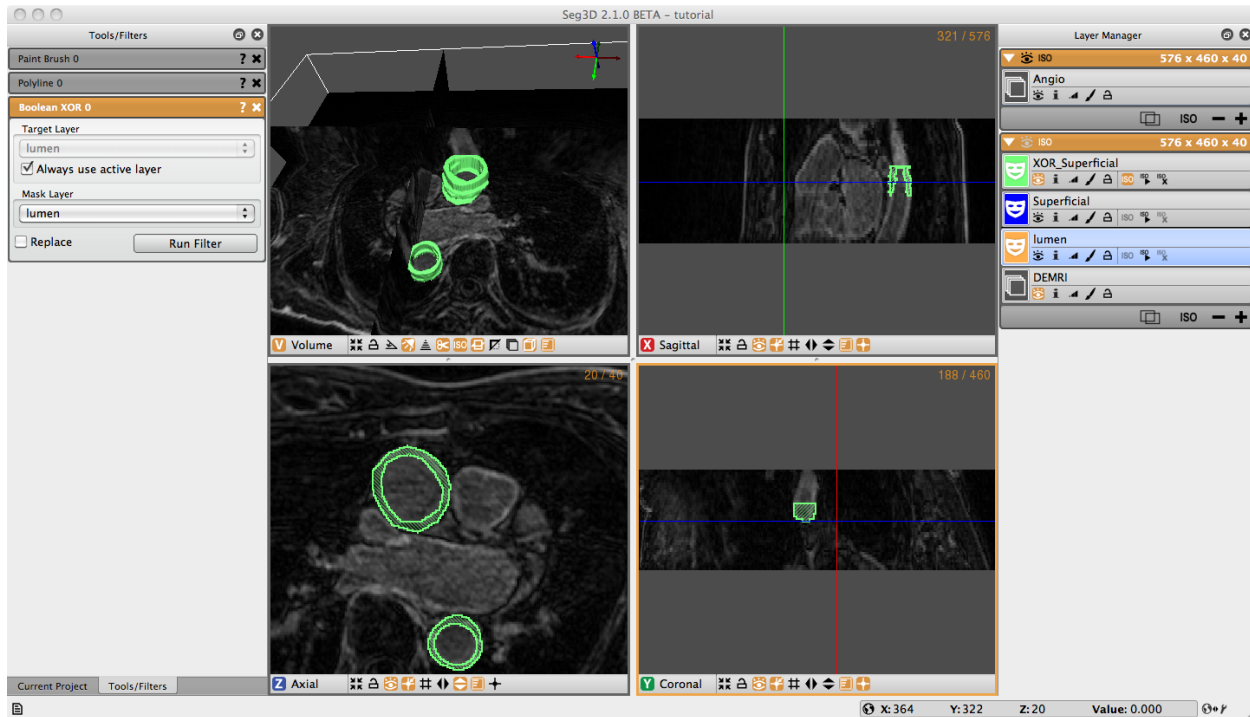


Fig. 2.37: Creating isosurface of label mask.

Mask Data

To extract the data covered by the label mask select the Mask Data option under the Data Filters menu. Choose the label mask you want to use to extract data with and click Set Label Mask in the Tools Menu. Select the Data layer you want to extract data from (DEMRI in this case), which should appear in the Target Layer field. Then choose the Mask Layer you wish to mask the data with (XOR_Superficial) from the drop down menu or by dragging the layer to the field. Set the background value to the desired setting (min_value), then click the Run Filter button. A new data layer will appear in the Layer Menu (called MaskData_DEMRI). This should show only the region of interest from the masked volume (Fig. 2.38). You may need to toggle off the viewing of the mask layers to see the data.

Saving

At this point there are multiple options for saving the work that you have done. Under the File menu are the options: Save Project, Save Project as ..., Export Segmentation, and Export Active Data Layer. There are hotkeys for these functions and many others that can be seen by clicking the help menu then Keyboard Shortcuts. Save Project/Save Project as ... will save the work that you have done to this point in for later restoration. The Export Segmentation option saves the label masks that you have created. Export Active Data Layer will save the image data selected in the Layer Menu. Individual mask and data layers may be exported by right clicking on the layers and choosing Export Segmentation/Data as... and then choosing the file type. As apparent, the export data types available are image stacks (bitmap, png, tiff, dicom), matlab v7.3, and NRRD (see <http://teem.sourceforge.net/nrrd/>). The preferred data format for CIBC software is NRRD, and is required for using BioMesh3D, a tetrahedral mesh generation software also developed by CIBC <http://biomesh3d.com>.

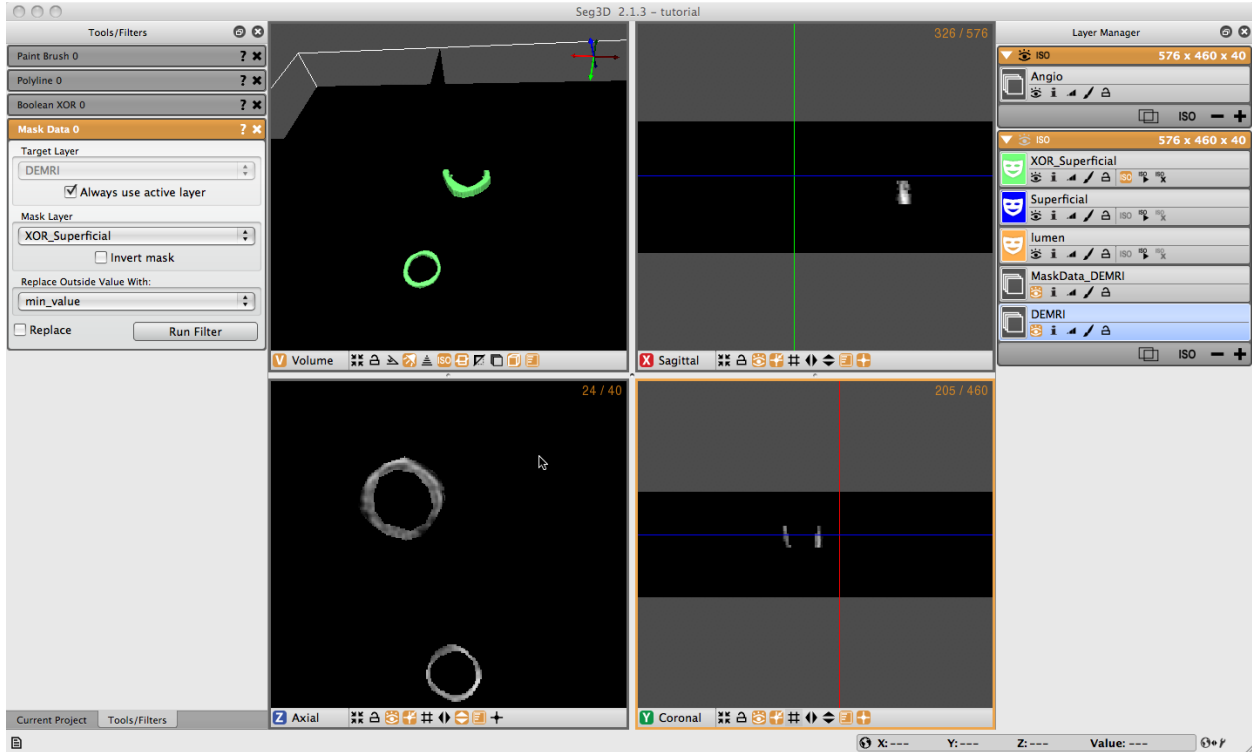


Fig. 2.38: Using label mask to extract data from volume.

2.3.4 Filters and Threshold Tool

Loading Angiogram

The second cardiac dataset that is part of this example is the angiogram. For this part of the tutorial load the Angio.NRRD file. Use the **Import Layer From Single File...** menu in the File Menu to load this dataset. An example of this dataset is depicted Fig. 2.39.

This dataset was previously registered to the DEMRI.NRRD dataset. As both datasets have the sampling distance and coordinate system. One can use label mask from the DEMRI dataset in the Angio dataset and vice versa.

Filtering data

As we want to use the threshold tool on the Angiogram data to segment out the inner wall of the both ascending and descending aortas, the data noise in the MR image needs to be suppressed. In this example we use a Median Filter and Gaussian Blur to filter the data.

Open up the **Median Filters** from the Filters menu and select the Angio data layer as the target. The radius indicates the number of the neighboring pixels that is used to derive the median from. In this case the default setting of one will be sufficient. Run the filter and compare the results by pressing space to toggle on and off the selected layer, which in this case should be the top most layer that was just generated if there are no mask layers present.

The second stage of filtering uses the **Gaussian Blur** from the **Data Filters** Menu. Again the default settings will do. Select the output from the median filter and run this filter. After filtering the data will look like the results depicted in Fig. 2.40.

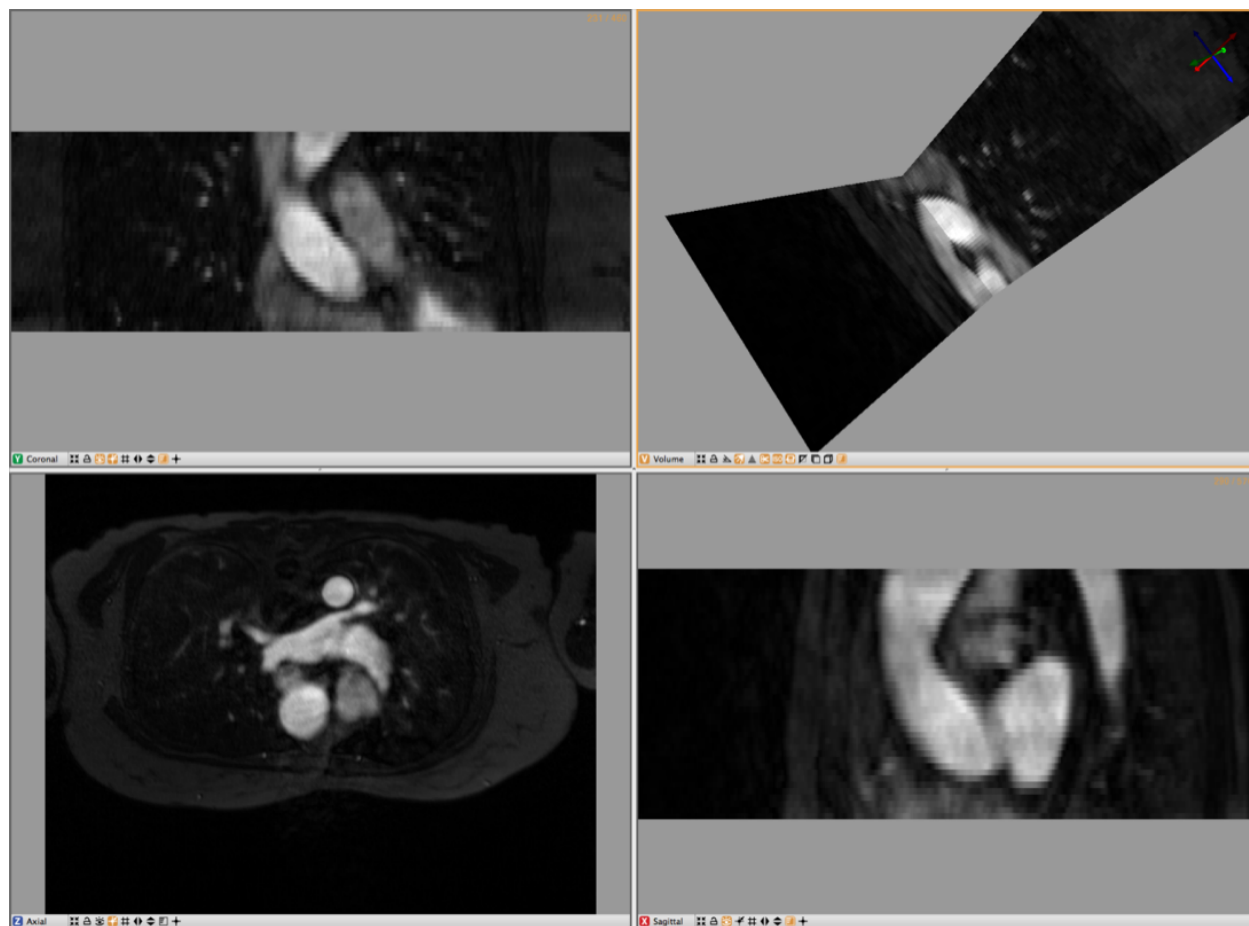


Fig. 2.39: Angiogram.

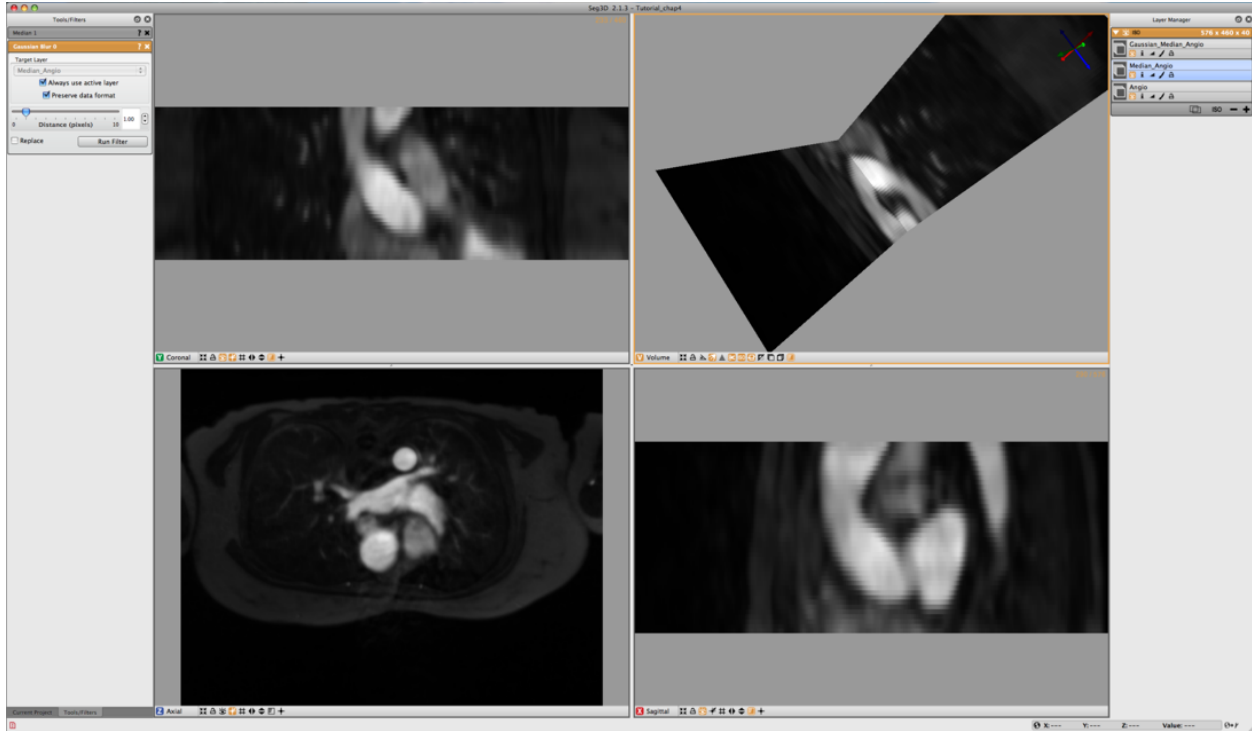


Fig. 2.40: Median and Gaussian filtered Angiogram.

Threshold Tool

To select the blood from the segmentation, use the **Threshold Tool** from the Tools menu. Once this tool has been opened, choose the filtered angio as the target and use the sliders to adjust the threshold values. In order to do this, switch off the visibility of both the filtered Angio datasets as well as the original Angio dataset and make the DEMRI visible. This will allow you to interactively generate a segmentation of the blood using the DEMRI as a visual confirmation, and filtered angio data for the threshold. Once the segmentation looks good press **Create Threshold Layer** in the Threshold Menu to generate the label layer. (see Fig. 2.41) The threshold tool however segments both aortas and the left atrium. To select just one of the of the vessels, select the newly created label layer and then select **Mask Data** in the Data Filters Menu.

Now create a new layer by pressing on the + symbol in this layer. This will generate a new empty layer with the same dimensions. Now open up the **Paint Brush** in the Tools Menu. Select the Mask Layer for the Target Layer in the Paint Brush Menu. Then under **Mask Constraint 1** select the Threshold Angio and start painting the Ascending Aorta. As a mask has been set the painting will be restricted to previously segmented blood. (see Fig. 2.42)

Paint a couple of slices and then generate the isosurface of the vessel. Once you have captured the vessel to your satisfaction, you may export the segmentation as described in the previous chapter.

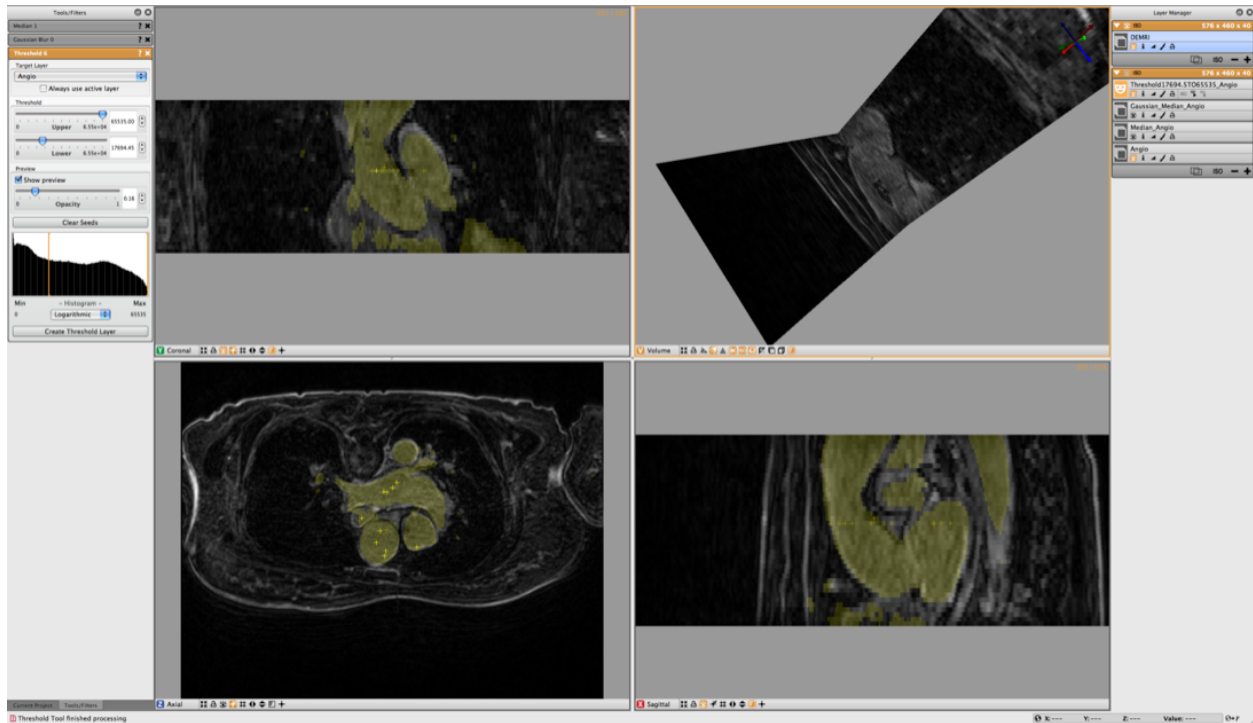


Fig. 2.41: Thresholding Angiogram while looking at DEMRI.

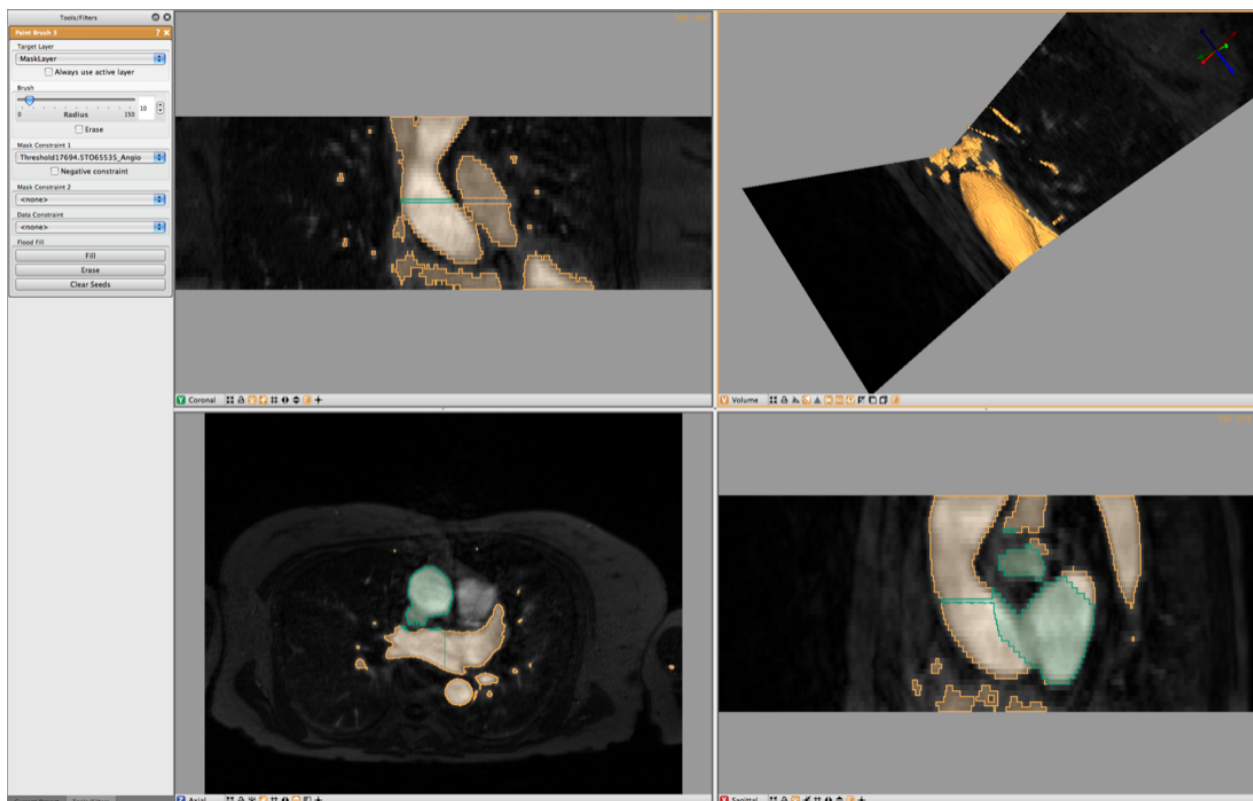


Fig. 2.42: Painting a label while masking from another label.

Otsu Threshold

Another thresholding tool available in the Data Filters menu is the Otsu Threshold. The Otsu Threshold uses an image intensity histogram to determine adequate threshold levels and segment the image based on the determined threshold levels. A demonstration of the Otsu Threshold will be done on the CT image that was taken on the 15 year old pediatric patient. The file can be found in the SCIRun datasets (see [Software Requirements](#)). We will be working with the CTbrain50.NRRD file found in the SCIRunData/Seg3D/Brain_DataSet directory.

To use the Otsu Threshold, first select the **Otsu Threshold** tool from the **Data Filters** menu. The tools window will then show the information on the Otsu Threshold tool. The target layer may be selected, and the number of thresholds can be selected, from zero to four different thresholding layers. Below the number of thresholds is the histogram representing the intensities of the pixels of the image (Fig. 2.43). The menu box below the histogram allows the user to select whether they would like a logarithmic or a linear graph.

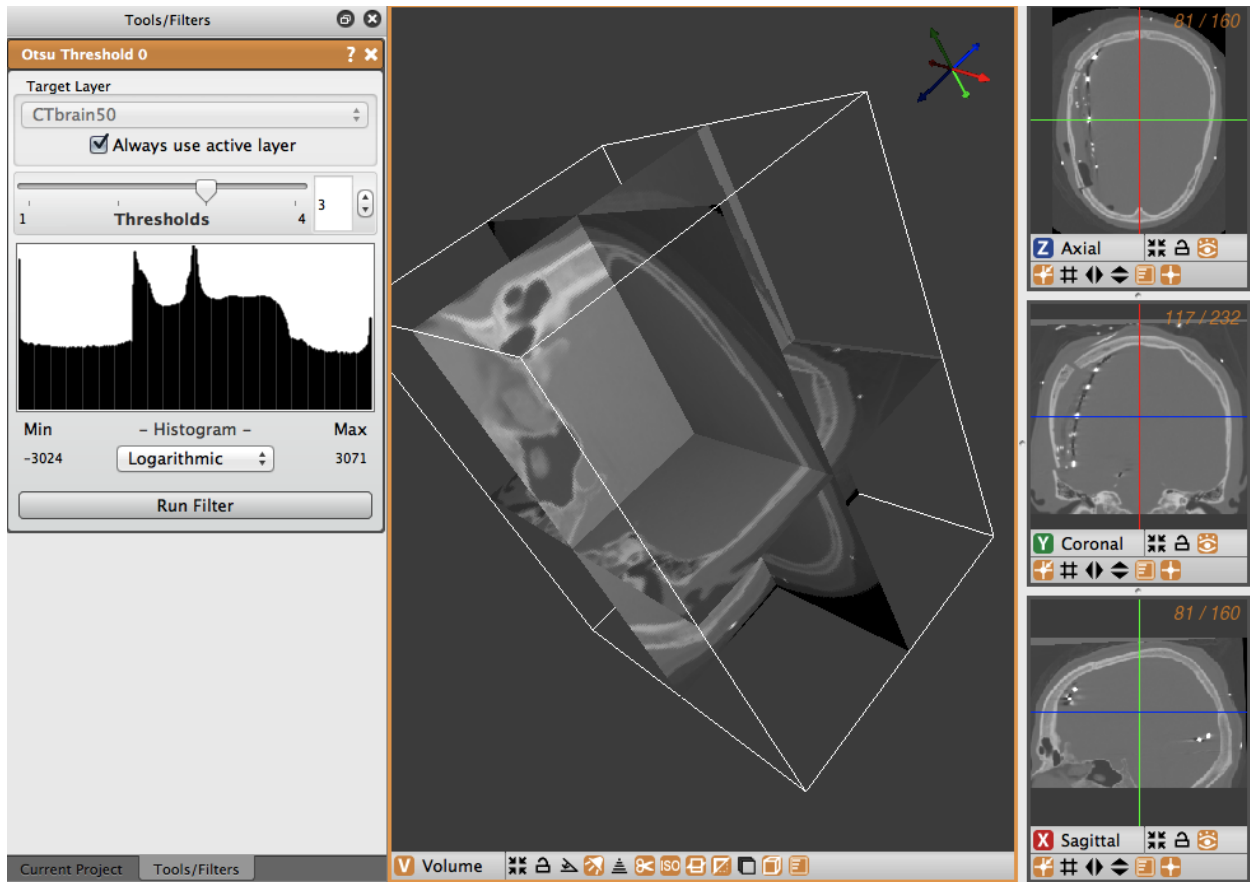


Fig. 2.43: Brain data before Otsu thresholding.

To use the Otsu Threshold tool on the brain CT image, select the tool and then choose three as the number of thresholds. Select logarithmic for the display of the histogram, and run the filter. The thresholded layers will appear in the layers view on the right. The number of layers corresponds to the number of thresholds selected in the tool view. In the brain data, the tool created different layers for dense matter like bone, other less dense matter, and air (Fig. 2.44).

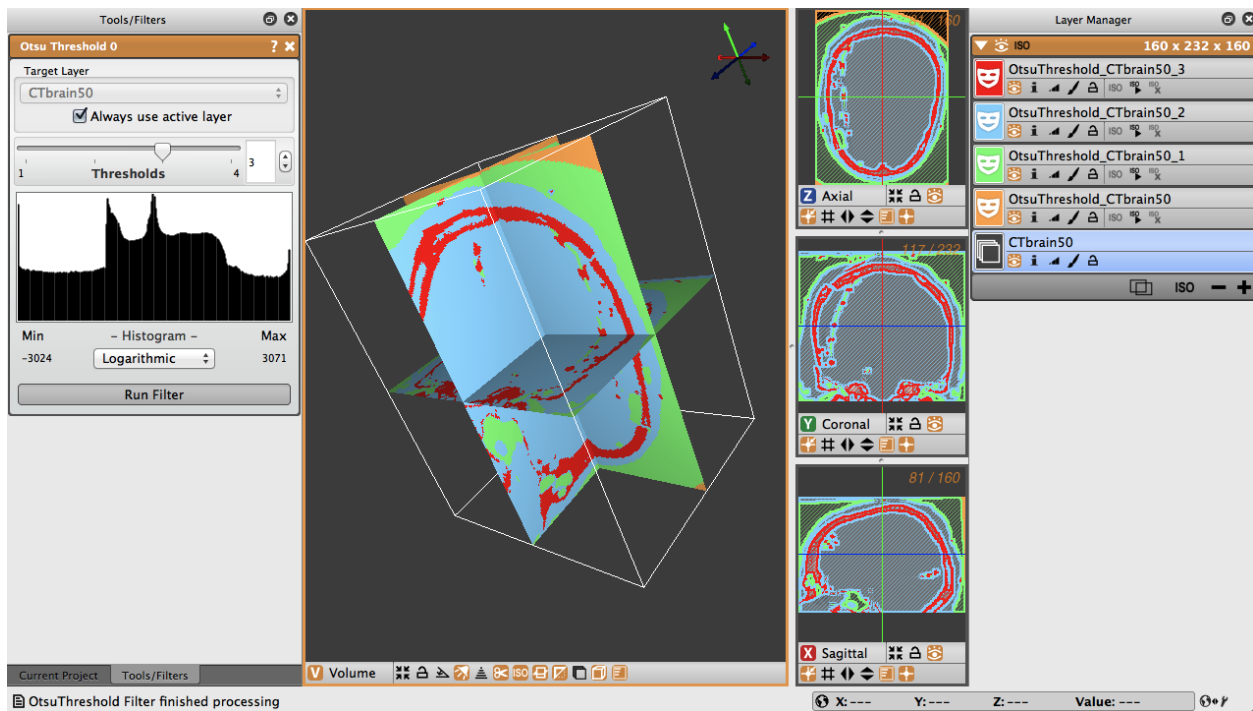


Fig. 2.44: Brain data after Otsu thresholding.

Canny Edge Detection Filter

An advanced tool for edge detection is the Canny Edge Detection Filter. This filter computes the edges using a Canny Edge Detection Filter, which involves smoothing the image with a Gaussian filter, calculating the second directional derivatives of the smoothed image, and performing a non-maximum suppression to find the correct extrema. The hysteresis thresholding is then applied to the gradient magnitude of the smoothed image to find and link edges. A demonstration of the Canny Edge Detection Filter will be done on the CT image that was taken on the 15 year old pediatric patient. The file can be found in the SCIRun datasets (see [Software Requirements](#)). We will be working with the CTbrain50.NRRD data file.

To use the Canny edge detection filter on the volume, first select the Canny edge detection tool from the Advanced Filters menu. In the tools window, the Canny edge detection tool will appear, allowing you to select the layer to run the filter on, as well as a distance slider. The distance slider sets the variance for the Gaussian smoothing algorithm. Below, there is a slider to set the threshold for hysteresis thresholding to find and link edges (Fig. 2.45).

The distance is the distance between edges, and the threshold is the pixel intensity threshold for distinguishing edges. Select the distance as 4 and the threshold as 40. When the layer, distance and threshold have been selected, run the filter to create a new layer with the Canny edges displayed (Fig. 2.46). Note the detection of edges between dense bone matter and less dense matter.

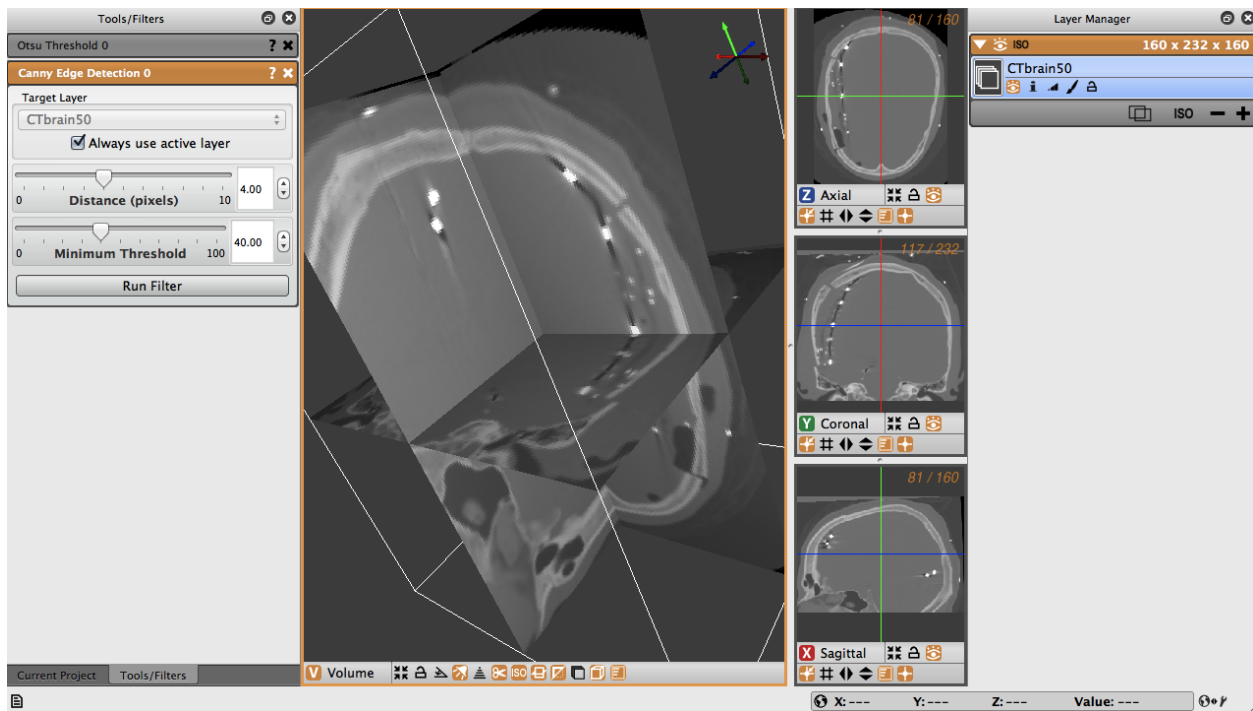


Fig. 2.45: Brain data before Canny edge detection.

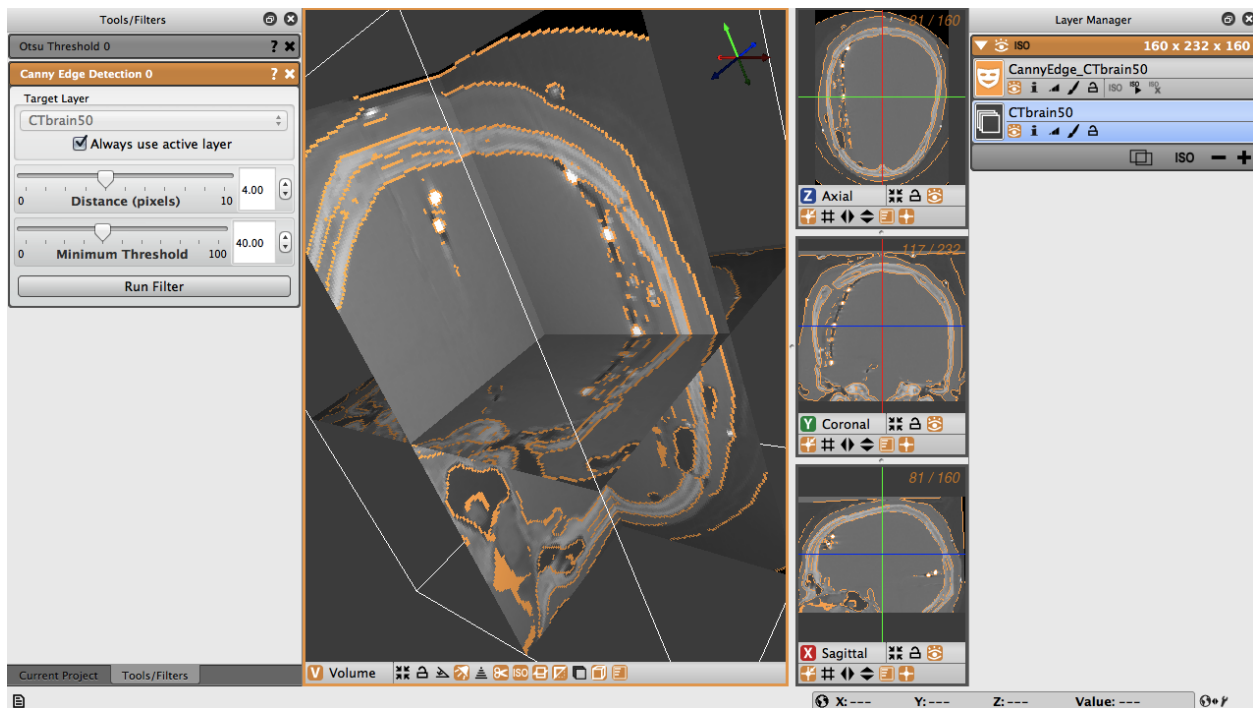


Fig. 2.46: Brain data after Canny edge detection.

Canny Edge Detection With Mean Filter

The Canny edge detection filter may also be run by first applying a mean smoothing filter prior to applying the Canny edge detection filter.

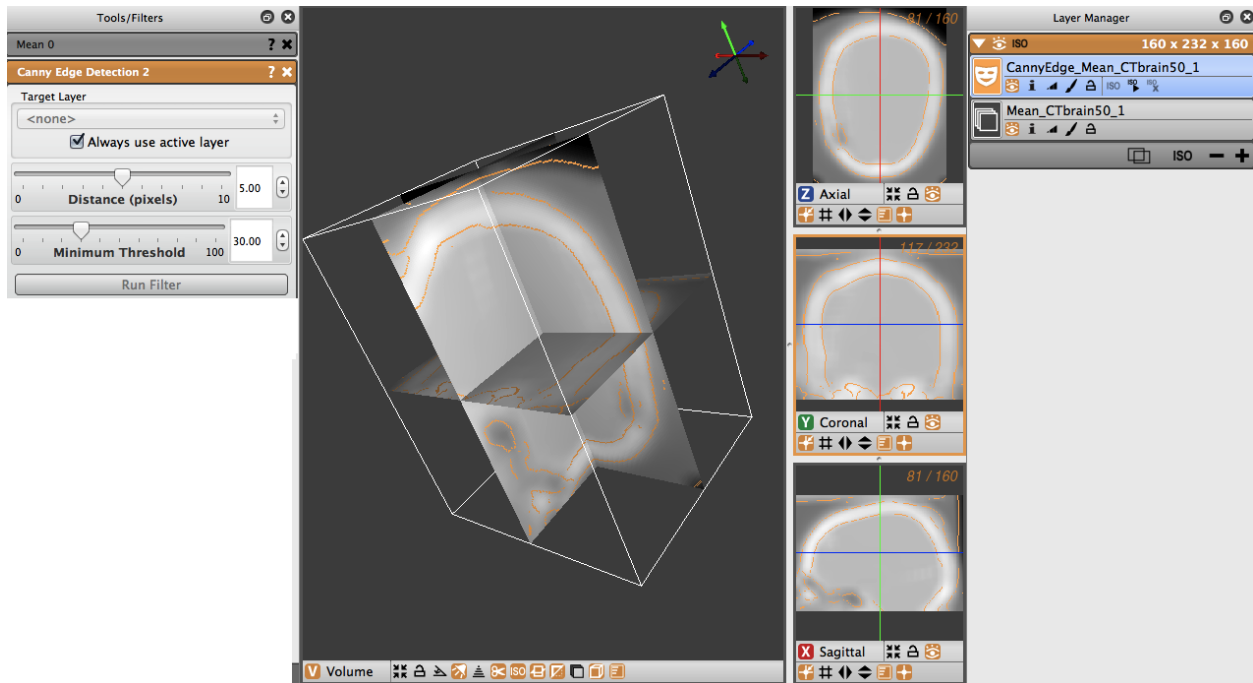


Fig. 2.47: Canny edge detection with mean filter

The image in Fig. 2.47 had a mean filter applied to it with a radius of 6, followed by a Canny edge detection filter with a distance of 5 and a threshold of 30. Notice how smooth the image is prior to the application of the Canny filter.

2.3.5 Speedline Tool

Speedline Tool Segmentation

In this example we will use the Speedline Tool (also known as the Live-Wire tool) to segment the left atrium. We start off with the Median Filtered Angiogram defined in the previous chapter. In this example, the data is further smoothed using the Gaussian filter. Using a smoothing filter is recommended before the using Speedline tool.

Open the **Speedline** tool from the Tools menu as shown in Fig. 2.48. Errors (highlighted in red boxes) are displayed in the **Speedline** dialog box explaining what needs to be done in order to proceed. The first error indicates that a mask needs to be selected in order to create the speedline. Push the plus button on the bottom right corner of the **Layer Manager** on the right of the screen. This will create a new layer mask and eliminate the first error message. Next, a data volume layer must be selected on which to apply the speedline. In the middle of the **Speedline** dialog box on the left of the screen is a drop down menu that has **<none>** selected. Click on the dropdown box and select a data layer.

Now we are ready to create the speedline. Insert a point on the border of the left atrium. Place a second point at some distance from the first and a third point even further away. With the third point in place, hover over a point until you get a hand icon. Now drag this point along the border and the tool will follow the edge of the volume. Add points at the corners and drag the speedline along the left atrial wall. An example of this is shown in Fig. 2.49.

Once the speedline has been completed, it can be filled in by hitting **F** or selecting **Fill** from the Tool Menu on the left. Make sure that the label mask you are using does not already have filled regions as the Fill operation will automatically

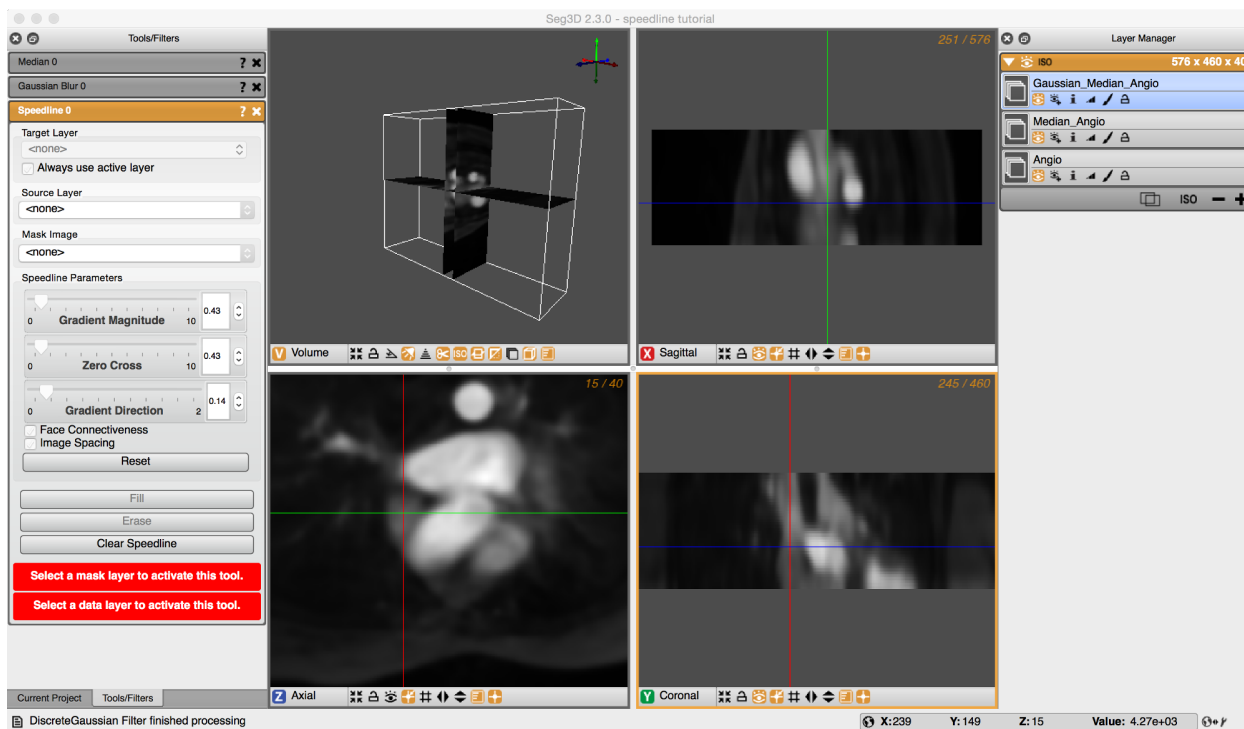


Fig. 2.48: The speed function displaying the gradient of the dataset.

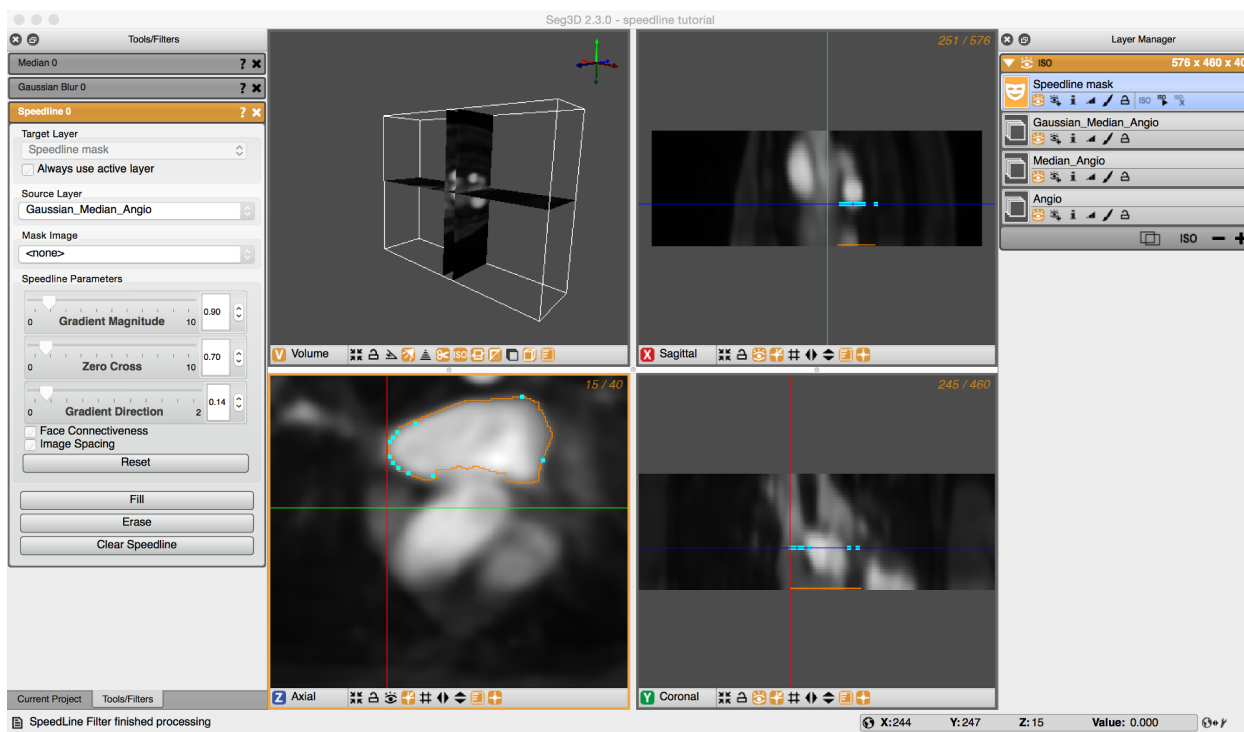


Fig. 2.49: Creating a speedline contour.

mask over any part that is within the speedline.

Now a segmentation is generated in one slice, proceed to the next slice by press < or >. The Speedline tool will try to accommodate it's shape to the new cross section. However, manual corrections can be made by moving points around using the left or right mouse buttons and dragging point around, or by adding additional points by pressing the left mouse button in the center of a segment. Sometimes removing points by pressing the right mouse button over a point will give a better result. Press **F** again to fill in the segmentation. (see Fig. 2.50).

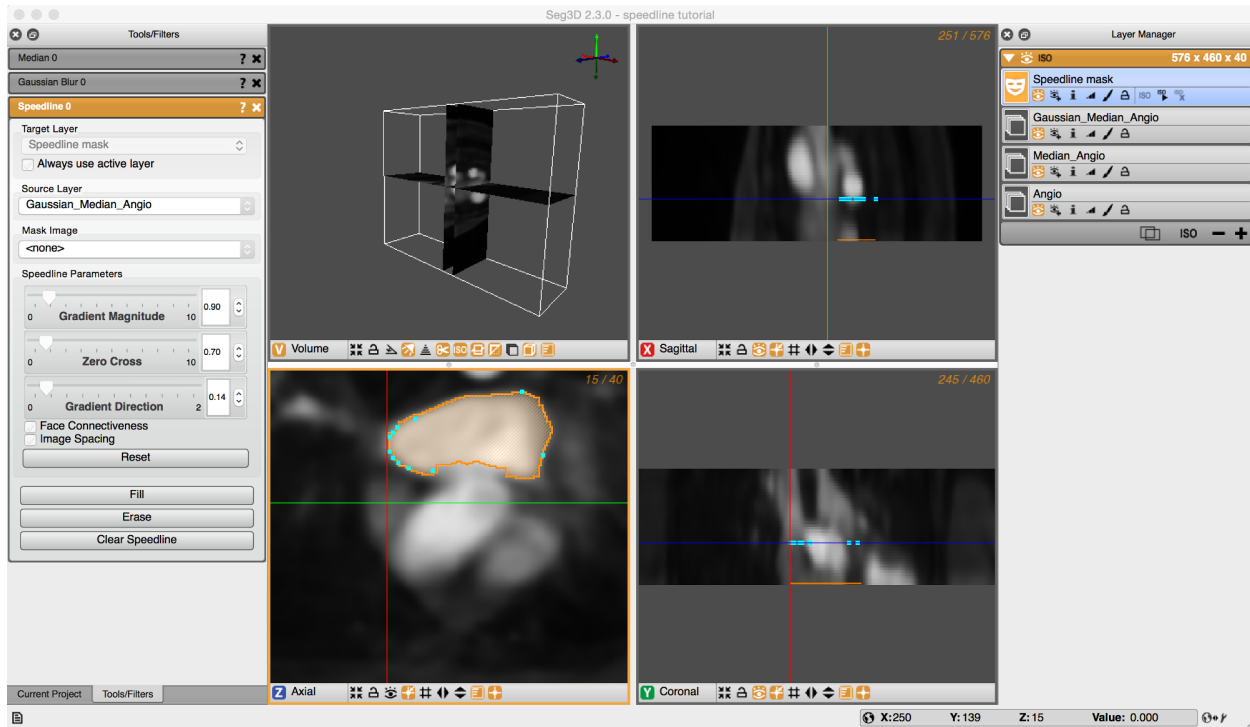


Fig. 2.50: Filling in a speedline contour.

By doing this sequentially for many slices, one can quickly segment a section of a structure. Once a portion of the structure has been segmented, press the **ISO** button below your selected label mask in the **Layer Manager** window on the right to visualize the results in the 3D.

This concludes the cardiac part of this tutorial. The next chapter will highlight some of the tools using a head dataset.

2.3.6 Segmenting a Brain Dataset

Loading Datasets

This part of the tutorial deals with segmenting a head dataset. The data that will be used is that of a 15 year old pediatric patient. This patient underwent surgery during which electrodes were implanted into the brain cavity. The image that we will be using are the MR image that was taken before the surgery and the CT image that was taken after. Note that one can clearly see in the image where the electrodes were inserted.

The final goal of this research is generate a patient specific model which can be used to find focus of brain activity during a seizure. For this a full volumetric representation of the head is needed. In this example we are extracting three tissues, namely the skull, the white matter and the gray matter. The remaining scalp and CSF can be obtained similarly, but is outside the scope of this tutorial.

Neighborhood Connected

Browse the SCIRunData/Seg3D/Brain_DataSet directory and load the following two files: CTbrain50.NRRD and MRI-brain50.NRRD. The first one is the post operative CT image and the second one is the pre-surgery MR image. After loading the datasets the Seg3D window should look like the one in Fig. 2.51.



Fig. 2.51: Loading the brain dataset into Seg3D.

Note that the data is not perfectly registered and that a part of the skull has been lifted to actually insert the electrode arrays.

To segment out the Skull we will be using the CT image. Select the CT image in the Layer Menu. Now from the Data Filters menu select the Neighborhood Connected filter. This filter needs seed points and finds every connected tissue of the same gray scale range. Hence placing seed points needs to be carefully. An example of this placement is given in Fig. 2.52.

Place the seed points with the left mouse button. While placing the seed points one can scroll to other slices. Each out of plain seed point will be shown less brightly. Place at least 10 seed points or so and then run the filter on the CT image. Be sure to select that one before running the filter.

The result should look like the ones in Fig. 2.53. To help understanding the 3D nature of the data, select the newly created label layer and run the isosurface by clicking the create isosurface icon ('ISO ') for the layer. This renders the isosurface of the current segmentation.

If parts of the skull are still missing, open up the **Neighborhood Connected** filter again and add a few more seed points. Note that all the previous seed points are still there to make it easier to add new points. To clear the seed points hit clear seed points in the filter menu (if needed). To add additional regions just place seed points in those areas. Use the isosurface rendering to check whether the skull is complete. Use the 3D view to see where the slices are and then put seed points where needed.

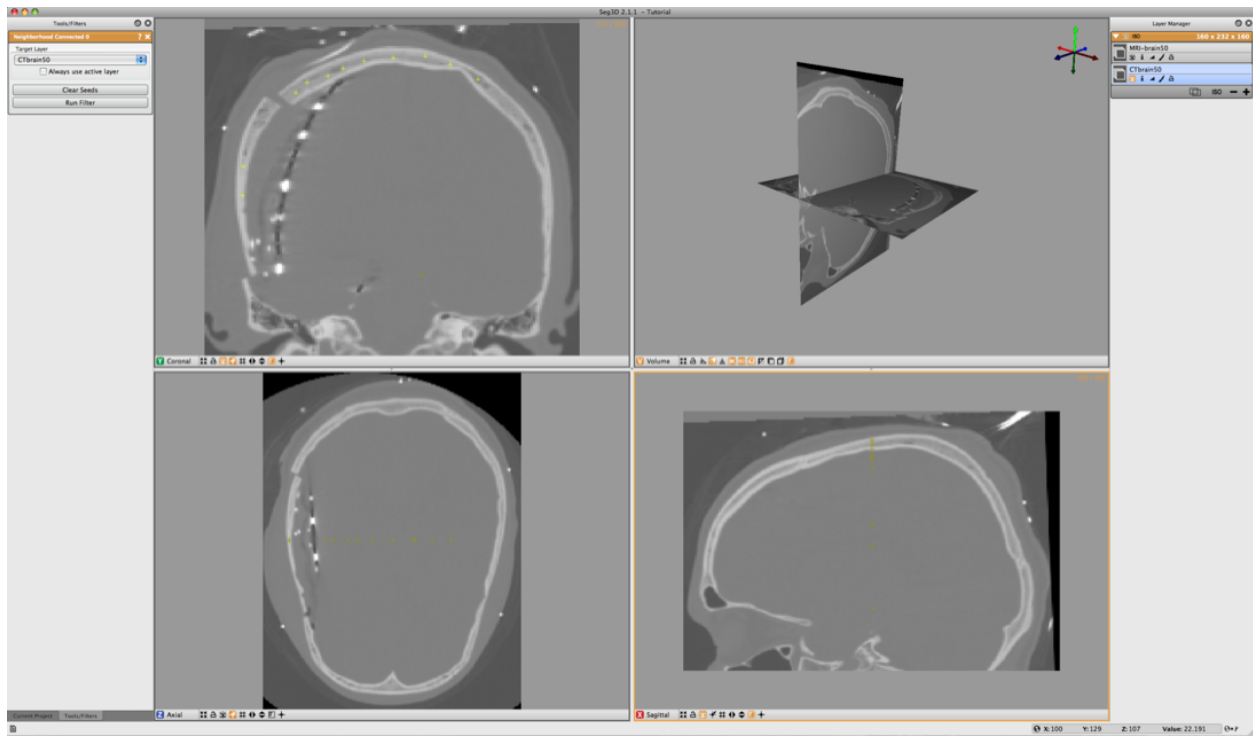


Fig. 2.52: Selecting seed points for Neighborhood Connected filter.

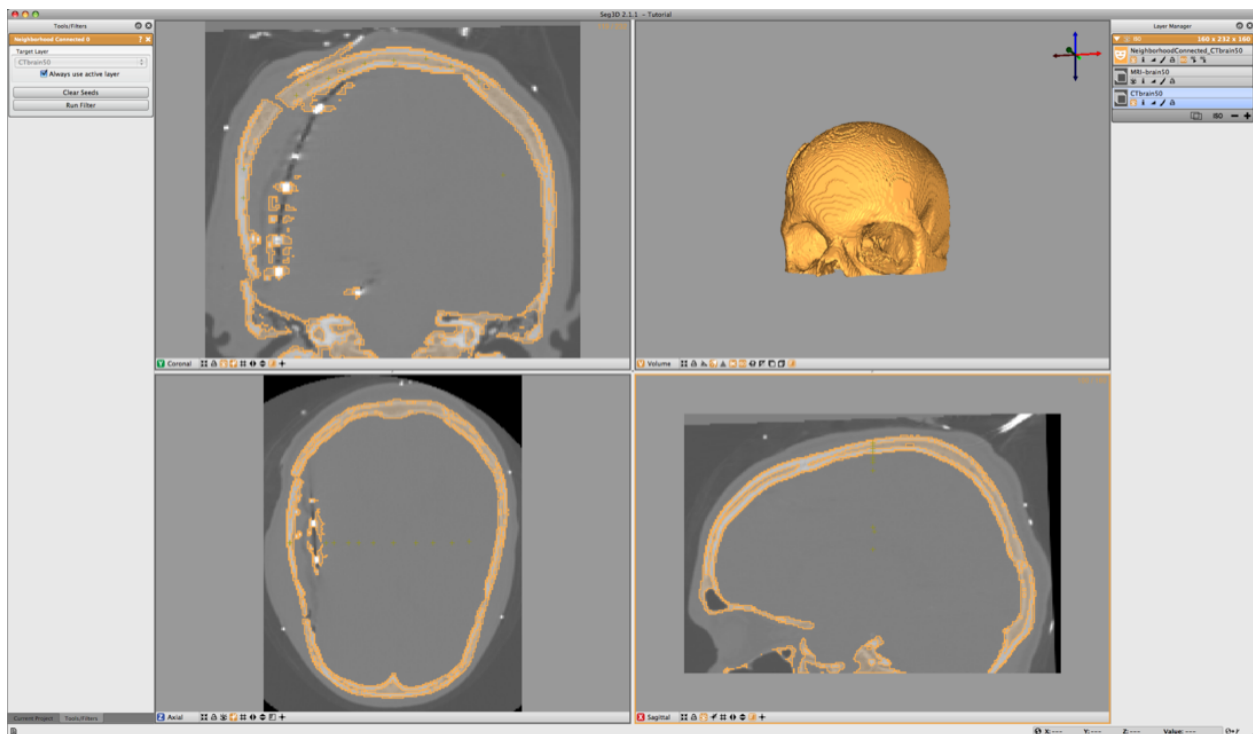


Fig. 2.53: Isosurfacing of Skull Segmentation.

Filtering MRI Data

The MR data contains a good data set for segmenting the gray and white matter. However MR images tend to have a bit of noise on them. Often running the data through the median filter will reduce the noise. Select the MR image in the Layer Menu and from the top menu select the **Median Filter** from the Data Filters menu. Run this filter and compare it to the previous MR image.

An easy way to do this is making all layers invisible, by clicking on the eye icons of the individual layers, and then make the original and the filtered MR image visible. Now select the top most image. Now the space bar will toggle between the two images. Navigate through the windows and look at the differences.

A second very useful filter for MR data is the **Intensity Correction Filter** from the Advanced Filters menu. This filter takes a while to run but filters the intensity in the data that have slight gradients due to the distance to the MR coils. The closer the image is to the location of the MR coil, the brighter the images generally are. This filter uses a polynomial model underneath to correct for this. We do this operation to make it easier to threshold the Gray and White Matter.

Apply this filter and again look at the differences. The resulting image is also given in [Fig. 2.54](#).

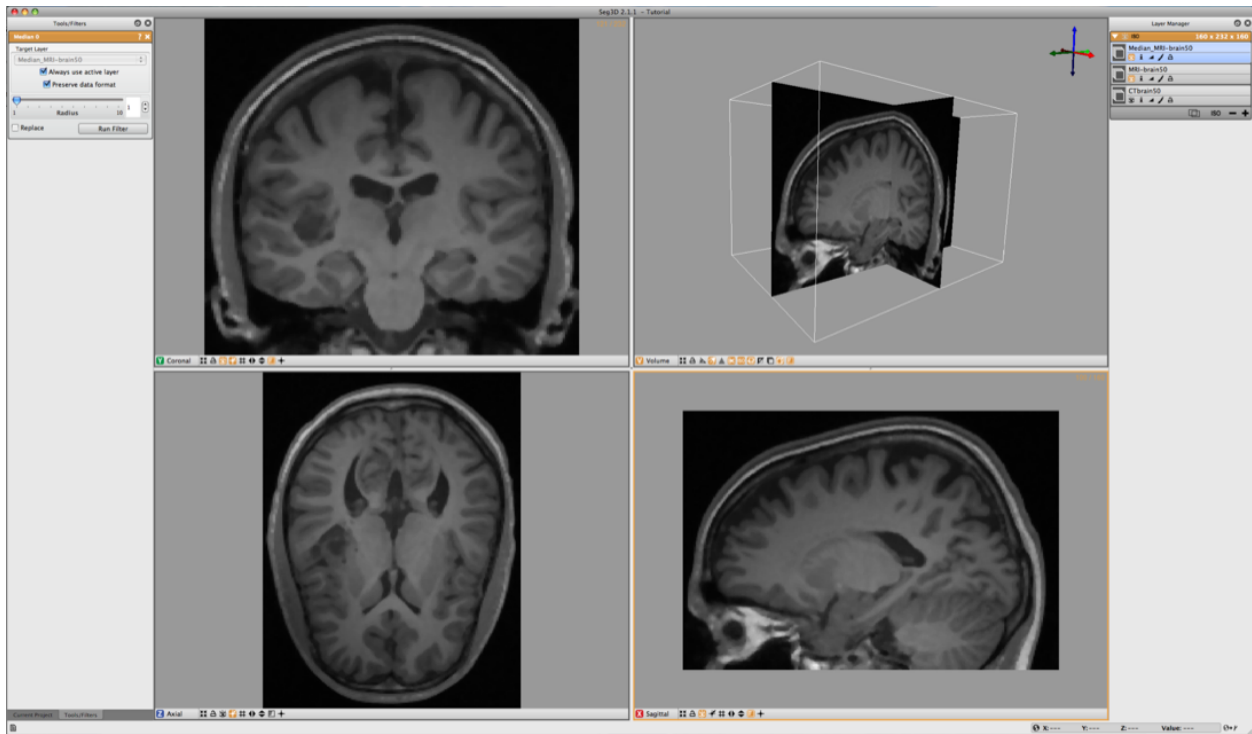


Fig. 2.54: Filtering the brain dataset.

Masking out Brain

To mask out brain, the easiest is to manually generate a mask and paint with a brush a rough segmentation of the brain cavity. The goal of masking out the data is to be able to threshold the white and gray matter inside the skull without picking up data from the scalp.

To jump start the process a segmentation is available in the dataset. This one was generated by manually painting the slices. Load the label layer through the **Import Layer From Single File...** from the File menu. Choose the Brain.NRRD dataset. Then click on Import file as a **Series of Masks**: There are two of these options and either one is valid because this data file has only one mask layer. After loading the dataset Seg3D should look like the image depicted in [Fig. 2.55](#).

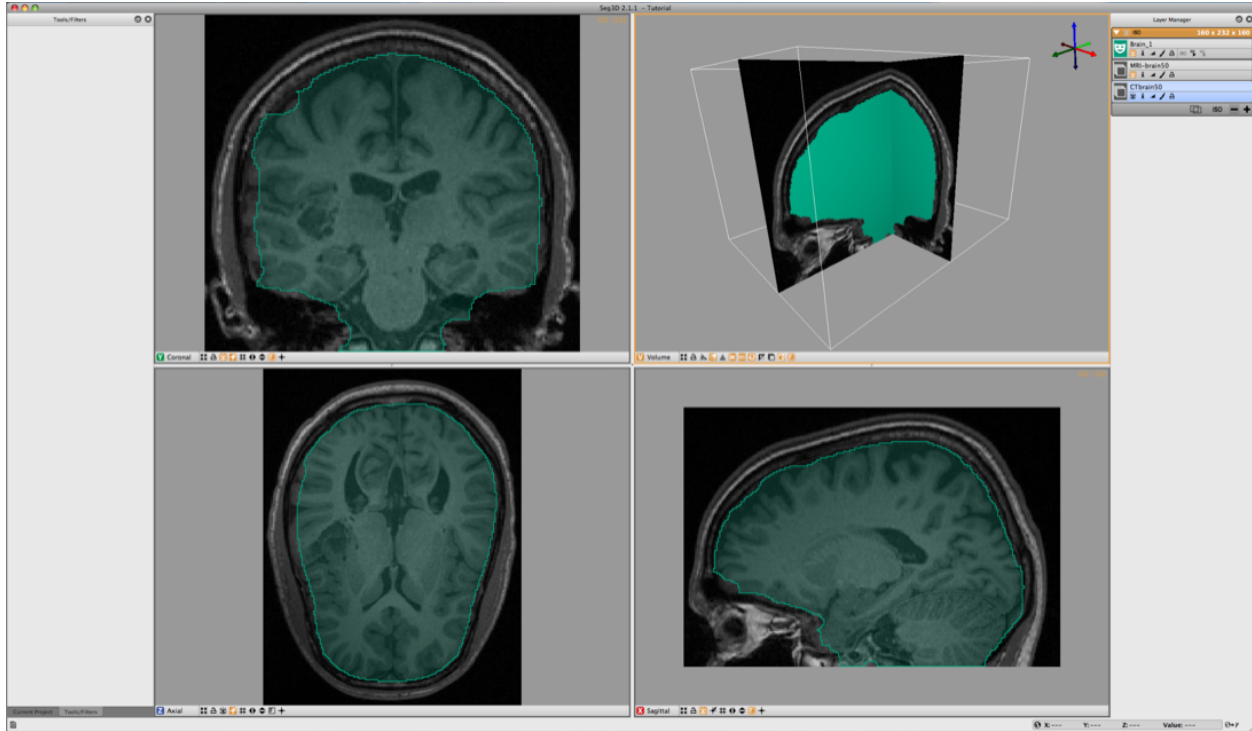


Fig. 2.55: Load a pre-made rough segmentation of the brain.

Now use the **Mask Data** in the Data Filters menu to set the Brain segmentation as a mask. Select the MRI-brain50 for the **Target Layer** and Brain_1 for the **Mask Layer**. Then click on **Run Filter**. This will create an image where all the scalp and skull are removed and only what is segmented is left as seen in Fig. 2.56. This mask will now be used for segmenting the gray and white matter.

Thresholding Gray and White Matter

To segment the white matter, we select the **Threshold Tool** from the Tools menu. Before using the tool make sure to turn off the Masking layer, by selecting “The Eye” underneath Brain_1.

Now select the Masked Data Volume. With the **Threshold Tool** insert a few seed points in the white matter. The threshold tool will now segment all the points that are within the range of the seed points. Keep adding points until the white matter has been marked. (see Fig. 2.57). Once the segmentation is good enough press **Create Threshold Layer** in the menu on the left and a new masking layer will be created.

Repeat the same procedure for the gray matter (see Fig. 2.58). Finally one can use the boolean operators to subtract the white matter from the gray matter. Select **Boolean AND** in the Mask Filters menu. Select one of the masks as the **Target Layer** and the other as the **Mask Layer** and click on **Run Filter**.

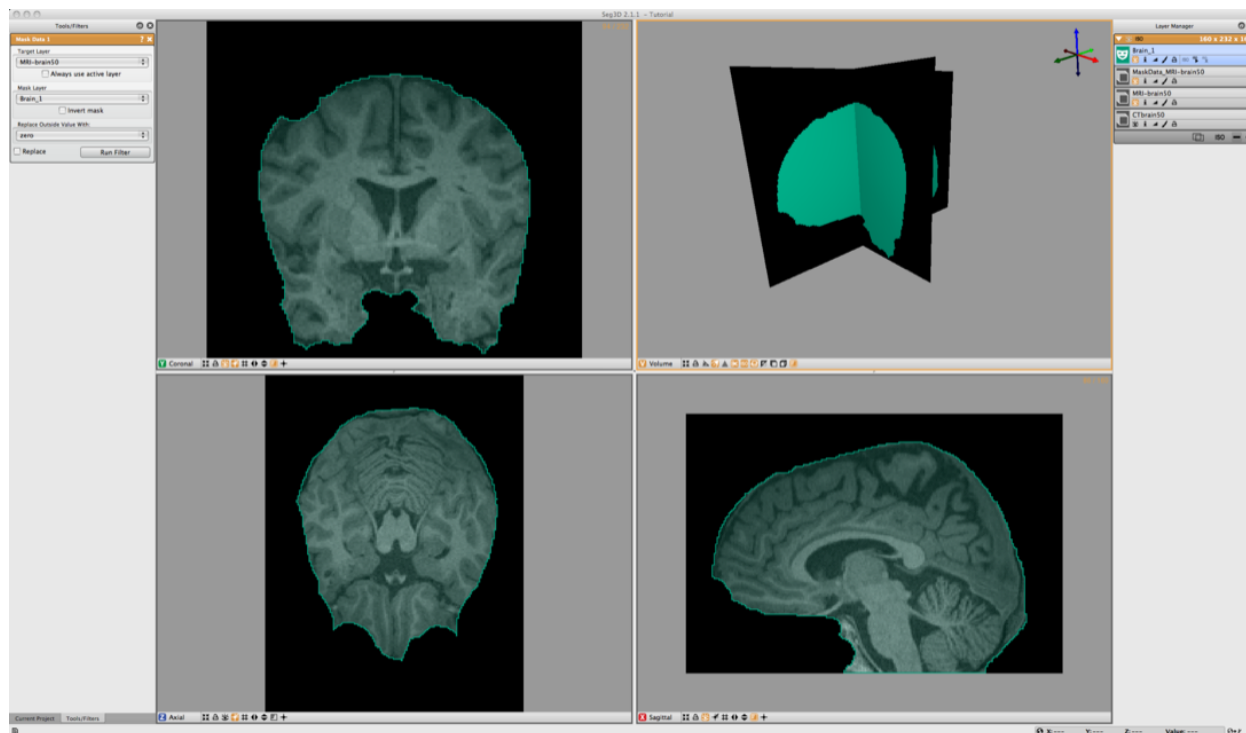


Fig. 2.56: Mask brain data with hand painted segmentation.

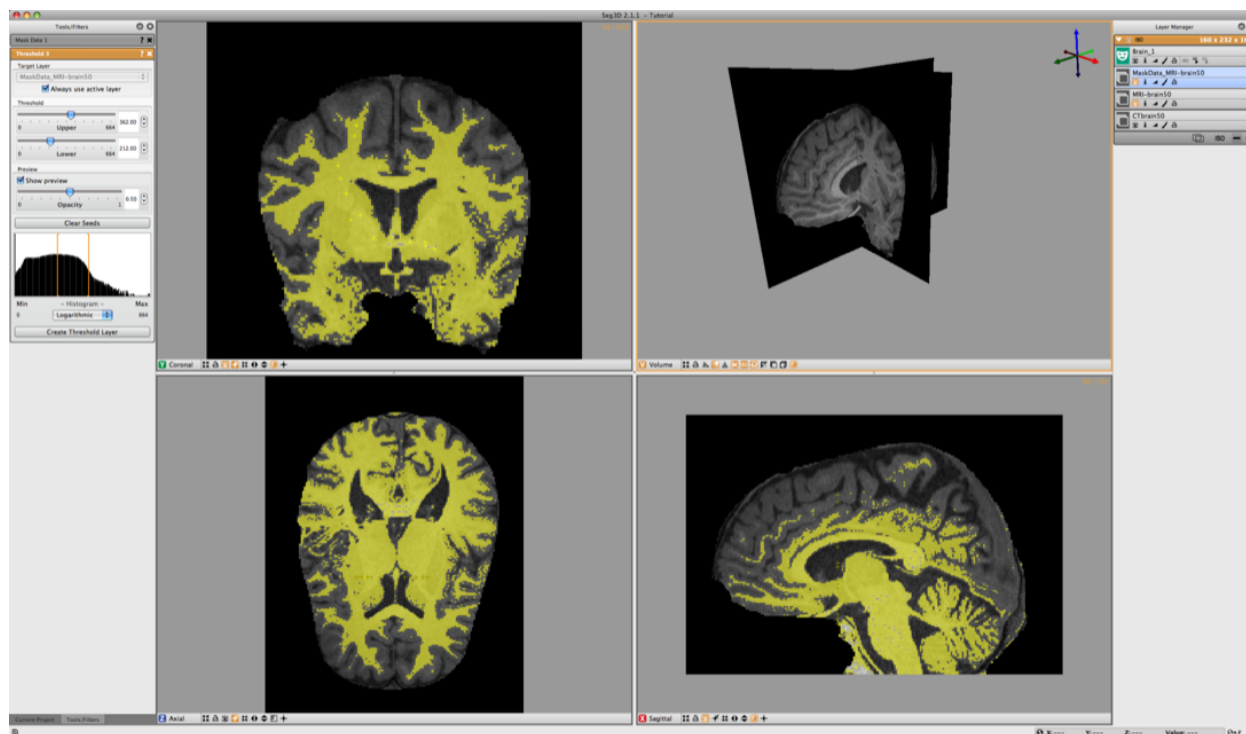


Fig. 2.57: Thresholding white matter using threshold range.

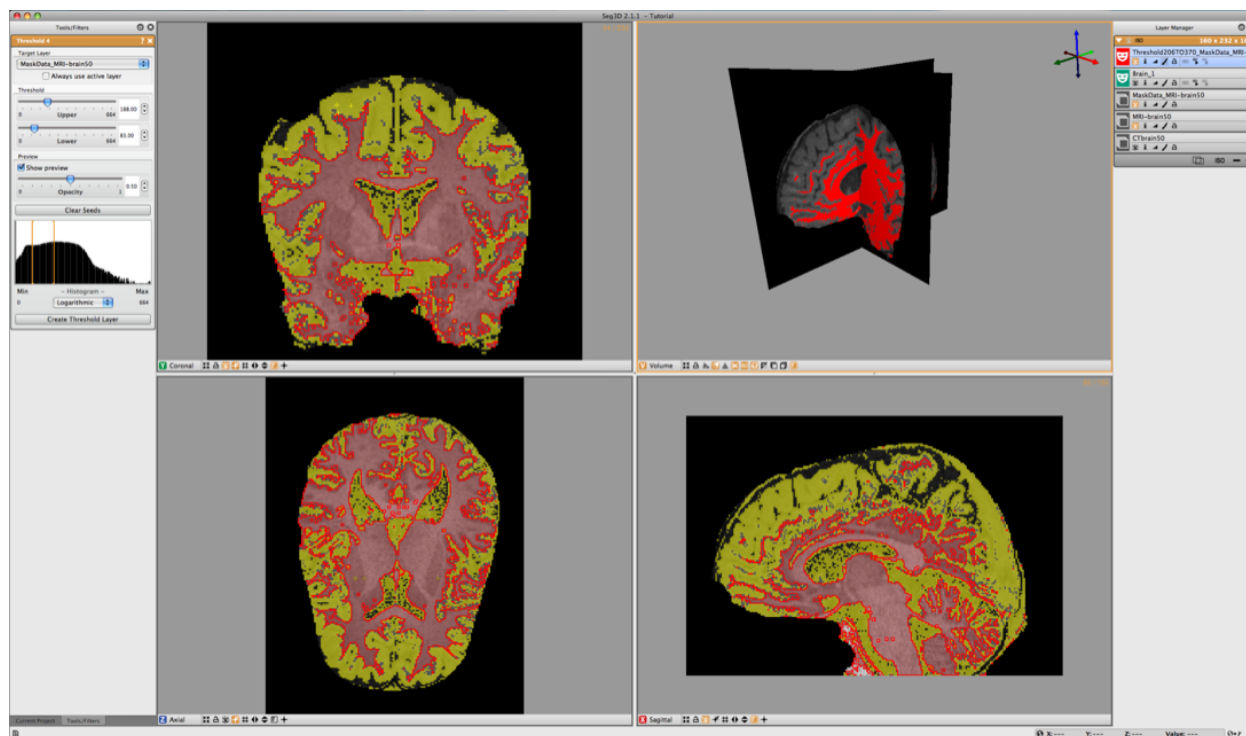


Fig. 2.58: Thresholding grey matter using threshold range.

2.3.7 Conclusion

This concludes the tutorial. Feel free to explore other structures in the head and make a full segmentation. Please explore the use of the other tools presented in this tutorial and incorporate them into your segmentation strategy if they are useful. Most of the tools and filters not presented utilize a similar structure and are relatively self explanatory; try them out and use them in your project. Contact us at seg3d@sci.utah.edu with questions or bug reports and we will try to answer your questions.

2.4 Video Tutorial

2.5 Build Seg3D

2.5.1 Installing Seg3D from source

Compiler Requirements

At least C++11 64-bit compiler support is required.

Windows

The current source code must be compiled with the 64-bit version of Visual Studio 2013. Visual Studio 2015 support is coming soon. If building the installer using CPack, NSIS is required. The code has not been tested under Cygwin or MinGW compilers.

Mac OS X

The source code base was built with Xcode 5-7 as well as GNU Make and Ninja and works for both environments on OS X 10.8+. If building the installer using CPack, PackageMaker is required.

Linux

The code base has been tested for use with GCC, and this is the recommended compiler for linux. GNU Make and Ninja build systems can be used. **Compiler must support C++11.**

Dependencies

Qt

Seg3D 2.4 And Newer

Before building Seg3D, please make sure that **Qt 5.9** has been installed on your system.

Windows

A typical Qt 5.9 build is as follows:

- Qt will need to be built from the command line using [Visual C++ command line tools](#). Troubleshooting tips can be found [here](#).
- Perl and Python are required before building Qt from source. The link below includes download links for both dependencies.
- Refer to the [Qt docs](#) for requirements for building from source.
- Run `configure` from this new directory with the following parameters (using the *-fast* flag to speed up build process can be helpful too):

```
configure -platform win32-msvc2013 -no-gif -qt-libpng -qt-zlib -qt-libjpeg -no-openssl -  
↪skip qtwebengine -skip qtwebkit -D _BIND_TO_CURRENT_VCLIBS_VERSION=1 -debug-and-  
↪release -opensource -opengl desktop
```

- You will be asked a few questions on license before configuring.
- Now run `nmake` and compile all of Qt (this is a long build).

Mac OS X

Qt binaries are available on the Qt website or can be built from source code.

Linux

Qt 5 static libraries are **NOT** supported. If building from source, ensure that the shared library build is enabled.

If building from source, use the following:

```
./configure -qt-xcb -qt-zlib -nomake examples -nomake tests -skip qtwayland -skip_
↳ qtwebengine -skip qtwebkit -opengl desktop
```

Seg3D 2.0 to 2.3

Before building Seg3D, please make sure that **Qt 4.7-4.8** has been installed on your system.

Windows

A Visual Studio binary build is available. To our knowledge the Windows Visual Studio development libraries are only available in a 32-bit version. A 64-bit version can be built from the source code download, configuring it as described on the Qt webpage. A typical Qt build is as follows:

- Download the open source version of Seg3D2. [Qt 4.8](#) is recommended.
- Qt will need to be built from the command line using [Visual C++ command line tools](#). Troubleshooting tips can be found [here](#).
- Create a new directory called *C:\Qt4.8-64bit* (or name of your choice), change into that directory and copy all the source files into it.
- Now, run `configure.exe` from this new directory with the following parameters (using the *-fast* flag to speed up build process can be helpful too):

```
configure.exe -platform win32-msvc2013 -no-gif -qt-libpng -qt-zlib -no-libmng -qt-
↳ libtiff -qt-libjpeg -no-openssl
-no-phonon -no-phonon-backend -no-qt3support -D _BIND_TO_CURRENT_VCLIBS_VERSION=1
-debug-and-release -opensource
```

- You will be asked a few questions on license before configuring.
- Now run `nmake` and compile all of Qt (this is a long build).

Mac OS X

Qt binaries are available on the Qt website or can be built from source code.

Linux

Qt is available from most package managers. Look for Qt 4.7-4.8.

CMake

CMake versions 2.8 - 3.4 are supported.

Compiling Seg3D

Once you have obtained a compatible compiler and installed Qt on your system, you need to download and install CMake (<http://www.cmake.org>) to actually build the software. CMake is a platform independent configuring system that is used for generating Makefiles, Visual Studio project files, or Xcode project files.

Once CMake has been installed, run CMake from your build (*bin*) directory and give a path to the CMake *Superbuild* directory containing the master CMakeLists.txt file.

You may build Seg3D with [Ninja](#) by adding the tag “-G Ninja” to your cmake command. Note: Mac users will have to install Ninja with [Homebrew](#).

Seg3D 2.4 And Newer

CMake requires a path to the Qt 5 install directory. Typically, this directory will contain *bin* and *lib* directories. For example, for Qt installed on Mac OS X from the Qt 5 binaries on the Qt site:

```
cd bin
```

```
cmake -DQt5_PATH:PATH=/Users/test/Qt5.5.1/5.5/clang_64 ../Superbuild
```

If building on Linux, the path may be under /usr/lib or /usr/lib64.

The console version ccmake, or GUI version can also be used. Set the **Qt5_PATH** variable to the location of the Qt 5 installation.

After configuration is done, generate the make files or project files for your favorite development environment and build.

The build script can also be used:

```
./build.sh --qt5=/Users/test/Qt5.5.1/5.5/clang_64
```

The Seg3D application will be built in bin/Seg3D.

Seg3D 2.0 to 2.3

Running cmake on the command line:

```
cd bin
```

```
cmake ../Superbuild
```

Following the previous example, the Seg3D application will be built in bin/Seg3D.

The console version `ccmake`, or GUI version can also be used. You may be prompted to specify your location of the Qt installation. If you installed Qt in the default location, it should find Qt automatically.

After configuration is done, generate the make files or project files for your favorite development environment and build.

A bash build script ([build.sh](#)) is also available for Linux and Mac OS X to simplify the process. Usage information is available using the `-help` flag:

```
./build.sh --help
```

Optional Image Registration Tools

To build the optional *python-based image registrations tools*, the CMake option `BUILD_MOSAIC_TOOLS` must be turned on, either on the command line with the flag `-DBUILD_MOSAIC_TOOLS:BOOL=ON`, or in the console or GUI interfaces.

Seg3D Packages

Build the *package* target in Visual Studio or Xcode, or make `package` using GNU make on Mac OS X.

2.5.2 Seg3D Data

Seg3D sample data (also used in the tutorial) is available [here](#).

2.5.3 Seg3D Support

For questions and issues regarding building the software from source, please email our support list: seg3d@sci.utah.edu

2.6 Ir-tools

The ir-tools image registration command line tools ([ir-tools wiki](#), [more tool documentation](#)) are available as Python commands in Seg3D. Seg3D needs to be built from source with the CMake option `BUILD_MOSAIC_TOOLS` turned on (*Seg3D build instructions*).

2.6.1 Commands in Python

The following commands can be run in Seg3D's python console. The `layerid` parameter is not used, and should be set to '`<none>`'. The Seg3D GitHub repository has [sample scripts](#).

- **ir-fft:** `fftfilter(layerid=<none>, directory=<directory>, output_mosaic=<output_mosaic>, [tile_strategy=default], [min_peak=0.1], [peak_threshold=0.3], [images=[]], [shrink_factor=1], [num_threads=0], [pyramid_levels=1], [iterations_per_level=5], [pixel_spacing=1.0], [clahe_slope=1.0], [overlap_min=0.05], [overlap_max=1.0], [use_standard_mask=false], [try_refining=false], [run_on_one=false], [sandbox=-1])`

- **ir-assemble:** assemblefilter(layerid=<none>, input_mosaic=<input_mosaic>, output_image=<output_image>, directory=<directory>, [shrink_factor=1], [num_threads=0], [tile_width=max], [tile_height=max], [pixel_spacing=1.0], [clahe_slope=1.0], [use_standard_mask=false], [save_int16_image=false], [save_uint16_image=false], [save_variance=false], [remap_values=false], [defer_image_loading=false], [feathering=none], [mask=], [sandbox=-1])
- **ir-refine-grid:** refinegridfilter(layerid=<none>, input_mosaic=<input_mosaic>, output_mosaic=<output_mosaic>, directory=<directory>, [shrink_factor=1], [num_threads=0], [iterations=10], [cell_size=0], [mesh_rows=0], [mesh_cols=0], [pixel_spacing=1.0], [displacement_threshold=1.0], [use_standard_mask=false], [sandbox=-1])
- **ir-stos-brute:** slicetoslicebrutefilter(layerid=<none>, input_fixed=<input_fixed>, input_moving=<input_moving>, output_stos=<output_stos>, [shrink_factor=1], [pixel_spacing=1.0], [clahe_slope=1.0], [best_shift_x=0.0], [best_shift_y=0.0], [best_angle=0.0], [best_overlap=0.0], [use_standard_mask=false], [use_refinement=false], [use_cubic=false], [regularize=false], [flip_fixed=false], [flip_moving=false], [mask_fixed=<none>], [mask_moving=<none>], [image_dir_fixed=<none>], [image_dir_moving=<none>], [sandbox=-1])
- **ir-clahe:** clahefilter(layerid=<none>, input_image=<input_image>, output_image=<output_image>, [max_slope=0], [shrink_factor=1], [window_x=1], [window_y=1], [remap_min=max], [remap_max=-max], [median_radius=0], [bins=256], [sigma=max], [use_standard_mask=false], [mask=<none>], [sandbox=-1])
- **ir-refine-translate:** refinetranslatefilter(layerid=<none>, input_mosaic=<input_mosaic>, output_mosaic=<output_mosaic>, directory=<directory>, [shrink_factor=1], [num_threads=0], [prune_tile_size=32], [pixel_spacing=1.0], [intensity_tolerance=0], [max_offset_x=max], [max_offset_y=max], [black_mask_x=max], [black_mask_y=max], [use_standard_mask=false], [use_clahe=true], [sandbox=-1])
- **ir-stos:** slicetoslicefilter(layerid=<none>, input_fixed=<input_fixed>, input_moving=<input_moving>, output_stos=<output_stos>, [shrink_factor=1], [descriptor_version=4], [pixel_spacing=1.0], [clahe_slope=1.0], [use_standard_mask=false], [flip_fixed=false], [flip_moving=false], [mask_fixed=<none>], [mask_moving=<none>], [image_dir_fixed=<none>], [image_dir_moving=<none>], [sandbox=-1])
- **ir-stos-grid:** slicetoslicegridfilter(layerid=<none>, input_stos=<input_stos>, output_stos=<output_stos>, neighborhood=<neighborhood>, [shrink_factor=1], [iterations=1], [grid_spacing=0], [grid_rows=0], [grid_cols=0], [median_radius=1], [num_threads=0], [clahe_slope=1.0], [minimum_overlap=0.5], [displacement_threshold=1.0], [disable_fft=false], [slice_dir_fixed=<none>], [slice_dir_moving=<none>], [image_dir_fixed=<none>], [image_dir_moving=<none>], [sandbox=-1])
- **ir-stom:** slicetovolumefilter(layerid=<none>, input_files=<input_files>, image_dirs=<image_dirs>, output_prefixes=<output_prefixes>, [slice_dirs=[]], [shrink_factor=1], [clahe_window=64], [left_padding=0], [top_padding=0], [right_padding=0], [bottom_padding=0], [tile_width=max], [tile_height=max], [clahe_slope=1.0], [visualize_warp=false], [use_base_mask=false], [preadded=false], [remap_values=false], [save_int16_image=false], [save_uint16_image=false], [image_extension=.tif], [load_volume=false], [sandbox=-1])
- **ir-add-transforms:** addtransformsfilter(layerid=<none>, input_files=<input_files>, output_prefixes=<output_prefixes>, [slice_dirs=[]], [sandbox=-1])
- **ir-blob:** blobfeatureenhancementfilter(layerid=<none>, input_image=<input_image>, output_image=<output_image>, [shrink_factor=1], [num_threads=0], [radius=2], [threshold=3], [median_radius=0], [use_standard_mask=false], [sandbox=-1])

2.7 Specifications

2.7.1 Minimum recommended system configuration:

- CPU: Core Duo or higher, recommended i5 or i7
- Memory: 4Gb, recommended 8Gb or more
- Graphics Memory: minimum 128MB, recommended 256MB or more

2.7.2 Windows

Seg3D will install on Windows 7-10 (64-bit only).

2.7.3 Mac OS X

The Seg3D application will run on Mac OS X 10.8+.

The Seg3D 2 installer is not currently signed. The unsigned installer can be run by control-clicking (or right-clicking if 2 button mouse support is enabled) on the installer package and selecting open. The unsigned Seg3D application can also be run for the first time by control-clicking (or right-clicking if 2 button mouse support is enabled) on the installer package and selecting open.

2.7.4 Linux specifications

Build from source

Seg3D must be *compiled* from source on Linux platforms (OpenSuSE, Ubuntu etc.) and requires at least the following:

- GCC or clang compiler that supports C++11
- Git 1.8 or greater (system default packages are usually sufficient)
- CMake (tested with 2.8 - 3.4)
- NVIDIA card and drivers for Linux
- Graphics cards must support OpenGL 2.0 or greater (not available on older Intel embedded graphics cards).

Consult the distribution-specific section for additional package information and the developer documentation for build instructions.

Ubuntu

We are currently testing on 64-bit Ubuntu 14.04.2 LTS and 13.04. We usually build Seg3D using the GCC C++ compiler, so we recommend installing the package that provides that compiler. Ubuntu comes with a relatively minimal install, so several packages are required.

Ubuntu 14.x and 13.x Packages It is easy to get the necessary packages listed below via the Synaptic package manager or the command line instructions below:

```
sudo apt-get install cmake-qt-gui cmake-curses-gui build-essential \
libxft-dev libxi-dev libxmu-headers freeglut3-dev \
libtiff4-dev git
```

Qt SDK Version 4.8 is recommended. The libqt4-dev package provides Qt 4.8 on Ubuntu 14.x and 13.x.

If the libpng library is not installed by the Qt SDK packages, it will have to be installed.

OpenSuSE

We are currently testing on 64-bit Leap 42.1 (OpenSuSE package repository information).

OpenSuSE RPMs:

- gcc
- gcc-c++
- Make
- CMake
- git
- glu-devel
- libXmu-devel
- CMake-gui
- CMake

Qt SDK Version 4.8 is recommended.

If the libpng library is not installed by the Qt SDK packages, it will have to be installed. The libqt4 and libqt4-devel packages provide Qt 4.8.

2.8 Python Snippets

2.8.1 Using Tools

The following are sample Seg3D python code snippets that can be run in the python console:

```
# Threshold tool doesn't need to be open in GUI to run
# thresholding as a python function.
# Assumes a data layer with ID layer_0 is already open.
result = threshold(layerid='layer_0', lower_threshold=56360,
upper_threshold=24248)
# Open the Threshold tool.
id = opentool(tooltype='thresholdtool')
# Close the Threshold tool.
closetool(toolid=id)
```

```
# Run the Neighborhood Connected filter (does not have to be
# open in the GUI).
# Assumes a data layer with ID layer_0 is already open.
# Need at least one seed.
result = neighborhoodconnectedfilter(layerid='layer_0', seeds=
[[-0.24,20.3,57.2],[35.3,18.1,57.3],[18.7,25.4,57.3]])
print(result)
```

2.8.2 Seg3D State Variables

Seg3D can be manipulated through state variables using the get and set functions. Look for the variable names in the Controller Window under the State Variables tab.

```
# Switch the first view window (there are 6 view windows in total) to
the Sagittal slice plane.
# Returns bool value.
result = set(stateid='viewer0::view_mode', value='Sagittal')
Tool IDs can be obtained either from using the opentool method as shown above, or by
↳ looking them up in the Controller Window under the State Variables tab. Tool variables
↳ can be used to batch process slices in a given layer.
# Get the list of vertices making up a polyline in the Polyline or
Speedline tools.
current_vertices=get(stateid='speedlinetool_0::vertices')
# Use polyline list
polyline(target='layer_2', slice_type=0,
slice_number=93,vertices=current_vertices,erase=False)
polyline(target='layer_2', slice_type=0,
slice_number=94,vertices=current_vertices,erase=False)
```

2.8.3 Running scripts

Python scripts can be run in the console using the open and exec commands, for example:

```
exec(open('/home/user/mypythonscript.py').read())
```

2.8.4 Script example

A more complex example involves waiting for a filter to finish before exporting the results, in this case, mask layers. It is necessary to check if data is available in the layer using the ID returned by the filter (a layer has 4 states: creating, processing, in use and available). In this case, a condition variable blocks until the predicate (lambda function that checks the layer state) evaluates to True.

```
import os
import threading
class MyThread(threading.Thread):
    def __init__(self, layerID, timeout=2.0, max_iterations=1000):
        self.layerID = layerID
        self.outputDir = outputDir
        self.TIMEOUT = timeout
        self.MAX_ITERATIONS = max_iterations
        self.condition = threading.Condition()
        threading.Thread.__init__(self)
    def run(self):
        stateIDData = self.layerID + "::data"
        layerStatus = get(stateid=stateIDData)
        print(self.layerID, " ", layerStatus)
        with self.condition:
            self.condition.wait_for(lambda: "available" == get
(stateid=stateIDData), timeout=self.TIMEOUT)
```

(continues on next page)

(continued from previous page)

```

    print("Layer {} done processing".format(self.layerID))
def transformExportNRRD(filenamees, outputDir):
    if not os.path.exists(outputDir):
        raise ValueError("Path %s does not exist." % outputDir)
    layers = []
    for f in files:
        if not os.path.exists(f):
            raise ValueError("Path {} does not exist.".format(f))
        importedFile = importlayer(filename=f, importer='[Matlab
Importer]', mode='label_mask')
        if len(importedFile) < 1:
            raise Exception("importlayer failed")
        layers.append(importedFile[0])
transformedLayers = transform(layerids=layers, origin=[-128.5,
-109.5, -108.5], spacing=[1, 1, 1])
print(transformedLayers)
threads = []
# make sure data is available in layers
for l in transformedLayers:
    thread = MyThread(l, 0.5, 4)
    thread.start()
    threads.append(thread)
for t in threads:
    t.join()
exportsegmentation(layers=transformedLayers, file_path=outputDir,
exporter='[NRRD Exporter]', extension='.nrrd', mode='single_mask')

```

2.9 Python Packages and Matlab

2.9.1 Installing packages

If there is an installation of the packages with other python builds on the machine, the system path can be used to use those packages. There are many ways to install packages in Python. pip is an easy way that is usually included with Python.

Installing pip

If pip is not installed for the Seg3D installation of python, follow these [directions](#). Be sure to use the python included in the Seg3D application. With OS x versions, it is in Seg3D2.app/Contents/Frameworks/Python.framework/Versions/Current/bin.

Installing numpy, scipy, and other major packages

- Make sure that pip is installed
- in the terminal:

```
cd Seg3D2.app/Contents/Frameworks/Python.framework/Versions/Current/bin
```

```
./python3 -m pip install numpy
```

Make sure that the package is found in Seg3D2.app/Contents/Frameworks/Python.framework/Versions/Current/lib/python3.4/site-packages.

Installing Matlab engine for Python in Seg3D

In the terminal:

```
cd "matlab_root"/extern/engines/python
```

```
Seg3D2.app/Contents/Frameworks/Python.framework/Versions/Current/bin/python3.4 setup.py  
↪ build --build-base="builddir" install --prefix="installdir"
```

Make sure that the package (matlab and matlabengineforpython...) is found in Seg3D2.app/Contents/Frameworks/Python.framework/Versions/Current/lib/python3.4/site-packages

Full instructions are located on the [mathworks site](#).

2.9.2 Matlab engine in Seg3D (through Python)

In Seg3D, Matlab code and functions can be run using the Matlab engine for Python in the python console or Python interface. To do so, make sure that the Matlab engine is installed (previous section).

To run the Matlab engine in the Python console in Seg3D, the **DYLD_LIBRARY_PATH** environment variable needs to be set to find python libraries and the matlab engine:

```
export DYLD_LIBRARY_PATH="path"/Seg3D2.app/Contents/Frameworks/Python.framework/Versions/  
↪ Current/lib: "matlab_root"/extern/engines/python/dist/matlab/
```

This can be added to the .bashrc file or you could try the /etc/launchd.conf file.

Full documentation for the Matlab engine can be found on the [mathworks site](#).

2.10 Bibliography

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

BIBLIOGRAPHY

- [Can86] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986. doi:[10.1109/TPAMI.1986.4767851](https://doi.org/10.1109/TPAMI.1986.4767851).
- [PM90] Pietro Perona and Jalhendra Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990. doi:[10.1109/34.56205](https://doi.org/10.1109/34.56205).
- [TOBrien02] Greg Turk and James F. O'Brien. Modelling with implicit surfaces that interpolate. *ACM Trans. Graph.*, 21(4):855–873, October 2002. doi:[10.1145/571647.571650](https://doi.org/10.1145/571647.571650).