iSeg v3.5
Manual

May 20, 2011
Contents

1 Introduction 5

2 Overview 5

3 Required Files 11

4 Example Projects 12

5 Technical Description 12
   5.1 Components ................................. 12
   5.2 Menu .................................. 14
      5.2.1 File .................................. 14
      5.2.2 Image ................................ 21
      5.2.3 Edit .................................. 27
      5.2.4 View .................................. 30
      5.2.5 Tools .................................. 30
      5.2.6 Atlas .................................. 32
      5.2.7 Help .................................. 32
   5.3 Viewers .................................. 33
      5.3.1 2D Viewers ............................. 33
      5.3.2 Cross-Section Viewers ................. 35
      5.3.3 3D Viewers ............................. 36
   5.4 Tissues .................................. 36
   5.5 Tabs ..................................... 38
      5.5.1 Thresh ................................. 39
      5.5.2 Growing ............................... 42
      5.5.3 Contour ............................... 43
      5.5.4 IFT .................................. 44
      5.5.5 Fuzzy .................................. 45
      5.5.6 Watershed ............................. 47
      5.5.7 OLC .................................. 48
      5.5.8 Interpol ............................... 53
      5.5.9 Smooth ................................. 54
      5.5.10 Morpho ............................... 57
      5.5.11 Edge ................................. 57
      5.5.12 Feature ............................... 59
      5.5.13 Measurement ......................... 60
      5.5.14 Vessel ............................... 60
      5.5.15 Picker ............................... 61
1 Introduction

iSeg is a general purpose segmentation tool that has been created in the context of extending SEMCAD X to a full hyperthermia treatment planning tool. The requirements when creating iSeg were: The segmentation tool should be able to generate highly detailed anatomical models within an acceptable time-frame. The segmentation tool should be able to work with all common types of medical image data (CT, MRI). It should interface well with SEMCAD X, be easy to use, robust and flexible to extend. Segmentation should be executable in clinical practice by technical staff with only minor supervision by medical personnel.

It was decided not to use any segmentation techniques that require prior knowledge (such as organ shapes, relative position of tissues), as the tool should be applicable to a large section of the body and many different tissues. Since different tissues and different image modalities require different levels of user interaction for reliable segmentation (bones can be segmented automatically in CT images, while some tissues can hardly be distinguished even in MRI), a tool box unifying various segmentation techniques and allowing a flexible combination thereof has been designed.

The general approach is to load image data, to apply various preprocessing operations to improve the image quality (smoothing...), to execute segmentation functions to identify regions which are then assigned to tissues, to use postprocessing to correct the segmentation and improve the tissue distribution and finally to export surfaces, outlines of label-fields in formats suitable for further use (e.g., by SEMCAD X).

2 Overview

The user works with three images: a source image, a target image and a tissue image (s. Figure 1 for a schematic depiction of the concept of the segmentation tool). Operations (preprocessing, segmentation) can be applied to the source image, resulting in new target images (which can in turn be shifted to the source image and used as new source images for the next operation if required, e.g., after a preprocessing operation). Minor modifications (postprocessing) can be performed directly on the target image. Segmentation operations result in homogeneous regions (same grey value) in the target image. Homogeneous target image regions can be assigned to a tissue type and will be added to the tissue image. Alternatively, target image regions can also be subtracted from the tissue image. Addition and subtraction can be performed slicewise or for entire homogeneous 3D regions. It is possible
to add only an individual connected component or all the components with the same grey value. Individual tissues can be locked to avoid erasing an already identified tissue with a new tissue. Slices from the source, target and tissue image can be named and pushed on an image stack to capture their state at a given moment and can be freely copied from there to another slice of the source, target or tissue image.

Figure 1: Schematic representation of the iSeg structure: Operations can be applied on one or more source images, taking into account user-defined parameters. The result images can in turn be used as source images, e.g. after a preprocessing step or to apply postprocessing operations. Images can be temporarily saved on a stack. Regions from the result image are assigned to specific tissues. Contours or surfaces can then be extracted from the tissue distribution, refined and exported to the simulation platform.

The tool can load .dicom, .raw, .avw, .mhd and .bmp images and subsections thereof. Contrast and brightness can be adjusted both in the visualization and the actual image. The segmented and identified tissue distribution, including color and name information, can be exported as slice-wise contours (which can be simplified using the Douglas-Peucker algorithm [14, 6]), as triangulated surfaces [27] (routines for surface simplification are available to allow the generation of reasonably sized, unsliced models), as .vtk data and as AMIRA labelfields. The user can set the slice thickness and pixel size, but this information will usually be extracted from the input .dicom files.

While the operations are generally performed on single slices, it is possible for most operations to be applied to volumes as well. As some slices may
cause specific techniques to fail, e.g. when a tissue does not have a sharp contour on a specific slice, the active slices on which 3D operations work can optionally be reduced to a range of user defined slices. User-configurable **Undo** and **Redo** operations are supported.

The source and target images are displayed in separate viewers (s. Figure 2) that simultaneously act as interfaces for the user’s point, click and draw operations. The tissue information can be overlaid as color information in both of these viewers, and it is possible to overlay the borders between homogeneous regions in the target image on the source image to allow for exact localization. Zooming and scrolling are supported by the viewers. Markers and limit lines can be added and displayed. The user can open additional viewer windows to display cross section images. However, these viewers do not act as interfaces, and if the user wants to perform operations on cross sections (e.g., to correct for discontinuities between slices), he first has to rotate the image volumes accordingly using available axis swapping operations.

![Figure 2: The frontend of iSeg](image)

**Figure 2:** The frontend of iSeg: Operations are generally applied to the source picture (left) and result in the identification of homogeneous regions (target picture, right), which can then be assigned to specific tissues (overlaid in color). Methods and parameters can be selected/set via tabs, (bottom left).

For preprocessing, a series of filters is provided to reduce noise and enhance boundaries. Smoothing can be achieved using simple **Averaging**, **Gaussian** averaging, **Median** and **Sigma** filters. Boundary enhancement
can be achieved using Gaussian Sharpening and Anisotropic Diffusion, which reduces noise below a user specified gradient level and enhances boundaries above this threshold. Furthermore, simple mathematical operations can be applied to the images (e.g., subtraction of images without contrast agent from images with contrast agents to highlight tumor locations; multiplication to apply masks...).

Various segmentation techniques are available in the toolbox and can be flexibly intercombined (s. Figure 3): Thresholding can be performed manually or with automatically determined thresholds (using k-means clustering, expectation maximization (both support multi-dimensional, multi-modal image information) [4, 12] or histogram analysis). Level-set based segmentation [26, 5] allows the user to click on a point, initializing a curve that can be enlarged by pulling the mouse and that sticks to boundaries. The current implementation uses gradient information, grey level information and information about the distance to the nearest edge (canny edge detector). A fast marching variant of the level-set method is available [25, 16], and a narrowband version is being developed [1]. Fuzzy connectedness based segmentation, using homogeneity and edge based criteria, is provided [18, 28] and has a similar interaction interface as the level-set method.

Several region growing methods are provided [19]. The user can manually specify seeds or use automatic, grey level based seed placement. Competitive growing from seeds attributed to different tissues, based on the image foresting transformation [9], and hysteretic growing are supported. To prevent leakage, the manual placement of limits is possible.

The interactive watershed transformation described in [11] (s. [23, 29] for more information) has been implemented. It allows the user to place various tissue markers, after which the algorithm automatically tries to find the optimal limits between competing tissues. Interactive corrections are possible, as the user can add additional markers prompting immediate redetermination of the tissue boundaries.

Various contouring methods are available, ranging from fully manual contouring to live-wire based contouring [5, 8, 7] (based on image gradient, laplacian zero-crossing, direction continuity and edge orientation information). Cooling [3] is supported by the live-wire method - parts of the contour that have not changed over a sufficiently long time period are automatically fixed.

Finally, the least automatic and most interactive segmentation technique available is a brush of variable radius that can erase, draw or shift tissue boundaries. This is suitable for high noise image data.

For body regions in which little inter-slice change of the tissue distribution is present, topologically flexible, shape based interpolation [21, 13] can
Figure 3: Some examples of segmentation techniques that have been implemented in iSeg: a) interactive watershed transformation (the merging of individual basins is interactively guided by user-set marks), b) fuzzy connectedness based segmentation identifies the likelihood for each image point of it belonging to the same tissue as the seed point (the likelihood distribution is shown), c) contour identification with the live-wire technique, d) level-set based segmentation, e) a competitive growing based segmentation that does not require precise interaction by the user, f) simple thresholding, suitable for bones, contrast enhanced vessels, internal cavities, etc.

be used. The user segments two distant slices, and the interpolation procedure automatically estimates the tissue distribution on intermediate slices. Outline correction routines or automatic adaptation (based on [24]) can then be used to correct differences between the estimated tissue distribution and the source image.

A series of postprocessing methods is provided to perform tasks such as hole closing, gap filling, outline correcting (various methods) and skin adding. Simple morphological operations can be performed. Various image analysis functions are provided, including: the analysis of grey value distribution in image subsections (mean, standard deviation, max/min), histograms, volume, distance, angle and four-point-angle measurements. Various helpful routines are available for programming, e.g., distance transformations (fast variant based on the ‘dead-reckoning’ algorithm [10] and other approaches), contour extraction, connected component analysis and edge detection.

Vessel segmentation is provided (s. Figure 4). A novel segmentation technique was developed that requires the user to specify the start and end
points of the vessel tree and uses a variant of the live-wire method to identify the vessel tree (in contrast enhanced CT images). This level of user inter-
action is deemed acceptable and allows the identification of even complex vessel trees. Furthermore, with this method the user can naturally interact by adding additional points when the algorithm fails to correctly identify a vessel track.

Figure 4: Livewire based vessel segmentation technique [20]: The user iden-
tifies the start- and end-points of vessels (left), which subsequently leads to the identification of the whole vessel tree (here: coronary arteries).

iSeg has been extensively used for the Virtual Family project [15] (s. Figure 5) and has been evaluated in a clinical environment by the Hyperther-
mia Unit of the Daniel Den Hoed Cancer Center in Rotterdam. The main conclusions are: Different methods should be used for different tissues (auto-
matic thresholding to identify bones...). Some tissues require interaction (but perhaps not on every slice when interpolation is applicable). Segmentation, except for the simplest tissues, is mostly performed in 2D due to speed con-
cerns and the higher likelihood of volumes to include regions where leakage can occur. Hybrid methods combining contour and area based information behave more robustly. Robustness is increased as well by using competitive region growing methods. It is crucial to provide each segmentation technique with its optimal interface (mouse clicks, dragging, line drawing, sliders...) to enable intuitive and fast usage. A predefined scheme should be developed that specifies the order of steps to be followed by a user to allow segmentation to be performed on routine basis by technical staff.

iSeg is programmed in a very modular way and can easily be extended. It is based on the ‘model-view-controller’ design pattern [22] and the notifi-
cation concept. The tool is written in C/C++, and the front-end uses QT (http://www.qtssoftware.com/), allowing for compilation on most platforms. 3D visualization is performed using the VTK package (http://www.vtk.org/)
Figure 5: The **Virtual Family** that has been segmented using **iSeg**. The man (age: 34), woman (26), girl (11) and boy (6) have sizes and weights that are typical for their age groups. In addition, a dog model has been created. They have all been segmented in high detail based on MRI image data (s. [15]). Additional surface simplification has been performed with the Amira software.

and the import of RTstruct DICOM images (presegmented data) is enabled based on the GDCM2 library.

### 3 Required Files

A complete installation of **iSeg** requires the presence of the following files in the following folder structure:

The main folder contains the **iSeg** executable as well as a series of dll’s (from QT, VTK, GDCM and QTech - s. 7). In addition it is the default location for the valid license file which by default is called **license.dat**. If the license is not found at this location, a dialog prompts the user to specify the license location and name or the license server. In the main folder two sub-folder should exist: **images** and **tmp**. In addition a **atlas** subfolder can exist and hold reference segmentations to help navigate the image data (s. 5.2.6). The images folder contains all the icons required by **iSeg**. The **tmp** folder contains at least the two files: **latestproj.txt** and **settings.bin**.
Other files can be stored in the `tmp` file (e.g., when axis flipping is performed - s. 5.2.2, or when a default tissue list is defined), but these files can be erased without problem. `latestproj.txt` contains the path to the latest projects which have been opened or created in `iSeg`. `settings.bin` is used to store the user settings and latest state of `iSeg` and should never be erased.

When a new license file is required, it should replace the current license file or be placed as `license.dat` file in the main folder. A warning is displayed upon opening `iSeg` if the current license is not valid anymore or if it will become invalid in less than 15 days.

### 4 Example Projects

`iSeg` has been used in various projects. Some of them will be mentioned here to give an idea of application areas:

- The virtual family [15] (s. Figure 5) is a set of meanwhile more than ten highly detailed models of humans (man, woman, children, obese, pregnant, asian...) with close to a hundred distinguished tissues that are commonly used in the context of EM simulations. These freely available models can easily be used in `SEMCAD X`. They have been segmented based on MRI scans using `iSeg`. In addition, rat, dog and pig models have been generated as well and are available for simulation purposes.

- Multiple clinical groups have used `iSeg` to generate patient specific models for treatment planning. `iSeg` has been most actively used in the context of hyperthermia treatment planning by the ‘Hyperthermia Unit’ of the Erasmus MC in Rotterdam, NL (e.g., [17, 2]).

- Other groups have used `iSeg` to generate realistic models of body regions for simulation purposes.

### 5 Technical Description

This section gives a technical description of the different components of `iSeg`.

#### 5.1 Components

When `iSeg` is opened it features a menu bar on the top (s. 5.2). Below two 2D viewers (s. 5.3.1) are present: The left one shows the **Source** image
on which operations are applied and the right one shows the Target image where the results of operations is typically displayed. On top of these images the assigned Tissue distribution can be displayed.

Between these image viewers are buttons that allow to copy a Source image to the Target side, to copy a Target image to the Source side, to exchange a Target image with a Source image or to exchange all Target images with the corresponding Source images (e.g., after a noise reduction operation has been performed, the result images can be moved to the Source side and used as a base for subsequent segmentation operations).

The scaling (zoom) of the image viewer content (and the content of other viewers accessible through the Image menu) can be specified in a field of the right of the Target image. Additional buttons allow to zoom in or out and to reset the zoom factor.

A slider and a spinbox below the viewers can be used to browse through the different slices. Next to it, a field displays and permits the modification of the slice thickness. When medical images in the DICOM format are loaded, the slice-thickness is automatically calculated if corresponding information is present in the DICOM files.

Below the viewer related area, on the left, various buttons allow the selection of segmentation and image processing tools. The parameters and options related to the selected tool are shown next to it and are discussed in detail in section 5.5.

A window allows writing notes that are saved in the project file and will be available again upon opening. The note window supports formatting (colour, font type...).

A stack is provided next to the note window which can be used to temporarily and permanently save Source, Target and Tissue images and retrieve them (s. 5.6). This functionality can furthermore be used to transfer images from one project to another.

Various elements on the right permit the managing of tissues and the modification of the tissue distribution (e.g., addition or subtraction of regions...) (s. 5.4).

Selected elements of the interface (tool tabs, stack, notes...) can be shown or hidden via the View menu. Many elements can furthermore be resized or dragged thus allowing the user to extensively personalize the user interface. These elements can be docked to the sides or bottom of the application or positioned as separate floating objects anywhere on the screen.

Different viewers can be activated from the Image menu (3D viewers, slice viewers for alternative slicing directions orthogonal to the main image view...). These viewers are embedded in separate windows and can be closed or rescaled without affecting the main window.
Additional dialogs can appear when the user chooses to activate selected functions which require additional parameters of file locations.

Figure 6: The main window and the various components of iSeg.

5.2 Menu

The following sections will discuss the various functions that can be found in the menu.

5.2.1 File

The File menu (s. Figure 7) provides the following functions:

- **New...**: Creates a new project. The user can specify the image dimensions and number of slices (s. Figure 8).

- **Open**: Creates a new project and loads image data. Currently, the following formats are supported: raw, dicom, mhd, bmp (only grey-scale bmp, no color bmp!), vti and avw (s. Figure 9). For raw data, the user has
to provide the extent. It is possible to load only selected slices or a subvolume of the raw data. In the latter case, the user has to specify the dimension and offset of the subsection and the start slice (if the whole data should be loaded, the start slice field should be set to zero). It is possibly to use both 8- and 16-
bit raw data. The bmp ordering is performed based on the file name which is expected to end with a number. It is possible to load only a subsection of the bmp images. When dicom files are loaded, the information contained in the files is used to order the images and to determine the offset vector, pixel size and slice thickness. If the selected dicom data contain multiple image series, the user is provided with the possibility of selecting which one he wants to open. With dicom, it is also possible to load only a subsection of the images. When loading dicom images, the user can specify if they are CT images. If they are, it is possible to rescale the grey values of the Hounsfield unit based image such that bones or muscles are ideally displayed in the 0-255 range. Optionally, values that end up being below zero can be set to zero and values above 255 to 255 if the crop option is selected.

Figure 9: The Load menu. The user can load raw, dicom, mhd, bmp, vti and avw files.
**Reopen**: Reopen is similar to Open, but it only reloads the images into the Source images (no new project is created). This is helpful, when the Source images have been modified (e.g., smoothed) but are required in their original form again. Care has to be taken to only reload images that have the same number of pixels as the original images. As the image dimension (number of pixels) is already known, it is not required to specify the subsection extent, when a subsection is loaded (s. Figure 10).

![Reopen menu](image)

**Figure 10**: The Reopen menu.

**Import RTstruct...**: This function allows loading segmentation information (tissue distributions) that have been segmented using another program and stored in the RTstruct dicom format. The GDCM2 library is used to perform this task. The function retrieves the names of all tissues mentioned in the RTstruct file. When no corresponding tissue exist in the current
**iSeg** project, it is created and assigned a colour based on information from the RTstruct file. Subsequently, the segmented regions from the RTstruct file are added to the segmentation in the **iSeg** project. As this is performed based on the absolute 3D coordinates, it is important to have correct values set in the offset vector (Image→Offset).

**Export Image(s)...**: This function allows to export one or all slices of the **Source** image, the **Target** image or the tissue distribution. The **bmp, raw, mat** (MATLAB) or compressed binary **vtk** format (point data) are available (s. Figure 11). This permits exporting the image data and segmentation into formats readable by other programs.

![Export Images](image)

*Figure 11: The Export Images menu. The user can save one or all slices of the Source image, the Target image or the tissue distribution.*

**Export Contour...**: This function exports the segmented tissues in formats readable by **SEMCAD X** (s. Figure 12). It is possible to export either outlines (essentially a 2.5 dimensional format where contours of tissues are specified slicewise) or triangulated surfaces (which can result in very large files). The tissues that have to be exported can be selected individually (of course, multiple tissues can be selected). The contour export guarantees (contrary to the triangle export) that no small spaces or holes are introduced at tissue interfaces. When triangulated surfaces are exported, it is possible to reduce the number of triangles by choosing the collapse option and specifying the ratio (a small ratio is equivalent to a large reduction). The employed algorithm performs the reduction by performing successive triangle collapse in a way that is optimized to retain the essential tissue features. This can be a very time consuming process. The number of points on the outlines can be reduce as well using the Douglas-Peucker algorithm. For this, the user has to specify the maximal allowed deviation of the simplified outline from the initial outline. A high value leads to smoothing, while a small value retains all relevant features. Using a value smaller than one typically results in significant memory requirement reduction without noticeable distortions. It is possible to specify a minimal area size below which contours will not be ex-
ported. This can be used to exclude species and small holes. The **Extrusion** option allows to make the first and the last slice thicker. This can be useful when the available image data is only a slab out of the entire person/animal and the segmented model will be used for simulations extending beyond the domain for which image data has been processed. When exporting outlines it is possible to add additional slices between the segmented slices to reduce staircasing due to large distances between the image slices. Topologically flexible interpolation is then used to estimate the tissue distribution between the segmented slices such that a smooth transition is obtained. Connectivity of tissues is guaranteed between slices as long as a partial overlap exists. It is possible to smooth triangle surfaces by checking the corresponding option. Triangle surfaces can be exported in the formats **vtp** (vtk), **stl** as well as **dat** (**iSeg** / **SEMCAD X** specific format). Various widely available programs (such as Paraview, Visit...) can read the vtk files. Which format is used is determined by the extension of the selected file name.

![Export Contour menu](image)

Figure 12: The **Export Contour** menu. The user can export the segmented tissues in formats readable by **SEMCAD X**.

**Export Tissue Distr.**: This allows to export the tissue distribution in
various formats readable by other programs: **vtk** point data (compressed and uncompressed, binary and ascii), **mat** (MATLAB) as well as **Amira labelfields**. Various widely available programs (such as Paraview, Visit...) can read the vtk files. In addition, an **xml** file can be exported that contains information about the various tissues: name, colour, label id, extent of the tissue bounding box (in pixels) and absolute coordinates of the bounding box (s. Figure 13).

![Export Tissue Distr. menu. The user can export the tissue distribution in various formats.](image)

**Save Project As...**  : Saves the current project and all its settings (stack, notes, marks...) at a user specified location. The level of compression (affects file size and saving/loading speed) is based on the value specified under **Edit → Settings**.
Save Project: Saves the current project and all its settings (stack, notes, marks...) at its current location overwriting the previous project file. The level of compression (affects file size and saving/loading speed) is based on the value specified under Edit→Settings.

Save Active Slices...: Same as Save Project As but only storing the active slices (specified in Edit→Active Slices).

Open Project: Opens an existing project.

Save Tissuelist...: Saves the current tissues (colours and names) in a human readable ascii file. This file can be edited manually. It can be used when multiple projects require the same tissue list as it can be imported with the Load Tissues.

Open Tissuelist...: Imports a tissue file (colours and names) in the format obtained by using the Save Tissues function. New tissues with the corresponding names and colours are created and appended to the list of existing tissues in the project.

Save Tissuelist as Default: Replaces the default tissue list available when starting iSeg by the current list of tissues (names and colours). This updates the file called def_tissues.txt in the tmp subfolder of the iSeg installation which specifies the default tissue list.

Project names: The last four opened/saved projects are kept in the menu and can be opened from here provided that the project files have not been moved. Their paths are stored in the latestproj.txt file in the tmp folder.

Exit: Leaves iSeg prompting the user to save the open project. User settings (visibility settings, last used tissue list...) are stored in the settings.bin file in the tmp folder and applied again the next time iSeg is started.

5.2.2 Image

The Image menu (s. Figure 14) provides the following functions:

Pixelsize...: This opens a dialog in which the user can specify the x and y dimension of a pixel (dx,dy) or alternatively of the x and y extent of the image (lx,ly) (s. Figure 15). This affects many things ranging from the viewers to the exported tissue geometry and more. When dicom images are loaded into iSeg which contain this information it is filled in automatically.

Offset...: This permits to specify the absolute coordinate of the lower left pixel of the first slice (s. Figure 16). The offset is important, when multiple scans of different sections have to be segmented separately and will be merged later or loaded separately into a common project. Their absolute offset vectors permit to correctly position them relative to each other. When
Figure 14: The **Image** menu.

Figure 15: The **Pixelsize** dialog of the **Image** menu. The user can specify the x and y dimension of a pixel \((dx,dy)\) or alternatively of the x and y extent of the image \((lx,ly)\).

dicom images are loaded into **iSeg** which contain this information it is filled in automatically.

**Pad...** : This function allows the user to add additional padding around the image. The user has to specify the number of pixels by which the image...
Figure 16: The **Offset** dialog of the **Image** menu. The user can specify the absolute coordinate of the lower left pixel of the first slice.

will be extended separately for all directions (s. Figure 17).

Figure 17: The **Pad** dialog of the **Image** menu. This dialog allows the user to add additional padding around the image.

**Crop...**: This function allows the user to crop the image. The user has to specify the number of pixels by which the image will be reduced separately for all directions (s. Figure 18).

Figure 18: The **Crop** dialog of the **Image** menu. This dialog allows the user to crop the image.

**Histogram...**: This opens up a dialog in which the histogram of the **Source** or the **Target** image can be displayed (s. Figure 19). It is possible to restrict the area of which the histogram is displayed to a rectangular subsection of the image of which the extend and the bottom left pixel have
to be specified. The histogram shows the frequency of grey values in the range 0-256.

![Histogram dialog](image.png)

Figure 19: The **Histogram** dialog of the **Image** menu. This displays the histogram of the **Source** or the **Target** image.

**Contr./Bright. **... : This opens a dialog (s. Figure 20 in which the user can modify the grey values of the **Target** image on either a single slice or all the active slices (**3D** option). **Get Range** calculates and displays the maximal and the minimal grey value of the slice/slices. **Scale** does scale the grey values in a linear manner such that what was previously in the user specified (or **Get Range** determined) range is now in the range 0-255. Pixels with values that were below the user specified range will end up having a negative grey value while pixels with values previously above the user specified range will have a value above 255. If **Crop** is pressed instead, pixels with values below the the user specified range will be set to zero, while values above it will be set to 255. The ⊖ and ⊖ sliders, stand for brightness and contrast respectively and adjust the brightness and contrast of the **Target** image(s). This function is different from the contrast and...
brightness slide on top of the image viewers in that it really modifies the grey values and not only affects their displaying.

![Image](image.png)

Figure 20: The Contr-Bright dialog of the Image menu. The user can modify the grey values of the Target image on either a single slice or all the active slices.

**Image Math.** ... : This opens a dialog (s. Figure 21) that offers functionality to perform mathematical operations on one or multiple images (3D option). The mathematical operations either involve the Source and Target image(s) or Target image(s) and a scalar (Image or Scalar mode). If two images are involved, the operation is executed in a pointwise manner. The result is written to the Target image. Addition, subtraction, multiplication and negation (actually, subtraction from a scalar value which is 255 in the Image mode). Image multiplication can be applied for example to erase everything in an image that is outside of a mask area (e.g., by using thresholding to identify the background in a CT image and multiplication to set the background to zero - this will render automatic image segmentation by the k-means method much more effective).

**Unwrap:** Can be used to unwrap images that have wrapped a larger greyscale range into a smaller range or images that show phase (and thus have jumps from 360° to 0°).

**x Sliced:** This opens a viewer displaying cross-sections orthogonal to the x direction (s. Figure 22). Either the Source or the Target image can be displayed and the tissue distribution can be overlayed in colour. The viewer window can be resized and it stays on top while open to allow displaying it while performing operations in the main iSeg window. A slider allows scrolling through the different slices. The Show zpos option permits to
Figure 21: The **ImageMath** dialog of the **Image** menu. This offers functionality to perform mathematical operations on one or multiple images.

Indicate by a green line the location of the slice currently displayed in the main window viewers. The **Show xypos** option permits highlighting the currently displayed slice in the y-slice-viewer if it has been opened. The image magnification is affected by the zoom factor specified in the main **iSeg** window. Closing the slice-view window does not result in **iSeg** being terminated. Two cross-section viewers (orthogonal to x and y direction) can be opened simultaneously.

**y Sliced**: This is similar to **x Sliced** but orthogonal to the y direction.

**3D Surface View**: This opens a viewer that can display a 3D surface view of the tissue distribution. The colour and transparency settings are defined by the tissue settings and can be modified by changing the opacity value or the RGB values in the **Mod. Tissue** dialog (s. 5.4). In addition, the overall transparency can be changed with a slider in the 3D surface viewer. The 3D viewer window can be resized.

**3D Isosurface View**: This opens a viewer that displays a 3D isosurface view of the source image distribution. The grey level at which the isosurface is extracted can be selected with a slider and another slider influences the transparency of the rendered surface. The 3D viewer window can be resized.

**3D Volume View Source**: This opens a viewer that can display a 3D view of the **Source** images (s. Figure 23). A high grey value region is rendered as a bright opaque volume, while a low grey value results in a transparent and dark region. Furthermore, it is possible to display up to two cross section planes. These planes can be oriented and moved arbitrarily, by dragging their orientation arrow and frame. The displaying of the individual planes and the volume can be deactivated. The volume rendering can be enhanced by shading, which increases the plasticity but can reduce rendering speed significantly. Volume rendering can be performed either using ray-tracing (faster, activated when **Ray Trace / Texture** is checked) or texture mapping (slower). The 3D viewer window can be resized.
Figure 22: The x Sliced dialog of the Image menu. A viewer displaying cross-sections orthogonal to the ‘x’ direction.

**3D volume view tissue**: This is similar to 3D volume view source but for the tissue distribution. The transparency and colour of individual tissues can be modified by changing the opacity value or the RGB values in the Mod. Tissue dialog (s. 5.4).

**Swap xy**: This function allows swapping the x and y axis of the image. Obviously, the data is mirrored in the process (heart moves from left side to right side).

**Swap xz**: This function allows swapping the x and the z direction. In the process a potentially large file is created in the tmp folder. The
swapping can be helpful, when segmentation can more easily be performed on slices orthogonal to original image slices. It is valuable as well at the end of the segmentation process to remove fuzzy borders due to slightly different segmentation boundaries on neighboring slices.

**Swap yz:** This is similar to **Swap xz** but involves the y and the z axis.

### 5.2.3 Edit

The **Edit** menu (s. Figure 24) provides the following functions:

![Edit menu](image)
**Undo**: This function can be used to undo one or multiple operations previously performed. The **Undo** functionality can be configured using the **Configure Undo** function. **Undo** can be reversed using the **Redo** function.

**Redo**: This function can be used to undo one or multiple **Undo** step as long as no modifying operation has been performed since.

**Configure Undo...**: This opens a dialog that permits the configuration of the undo/redo functionality to control the maximal memory available to it (s. Figure 25). The user can specify, whether operations that affect many slices (3D operations) and therefore require large amounts of memory to backup should be undoable, the equivalent of how many images should maximally be stored and how many operations should be undoable maximally. Operations that require more memory to backup the previous state than is provided by the user in the undo configurations result in the undo history being erased.

![Configure Undo dialog](image)

Figure 25: The **Configure Undo** dialog of the **Edit** menu. This allows the configuration of the undo/redo functionality to control the maximal memory available to it

**Active Slices...**: This function allows to restrict the slices on which 3D operations take effect (s. Figure 26). More important than the speed improvements this can provide is the fact that it restricts algorithms such as region growing algorithms to a selected range of slices. Therefore, an algorithm that might perform badly when applied to all slices because it ‘leaks’ on some slices might prove to be useful when the active slices are restricted such that the leak producing slices are excluded. A typical example is the hysteretic thresholding method.

**Settings...**: This opens the the settings dialog where general settings of **iSeg** can be adjusted. Currently only the compression level can be set, but future versions of iSeg will offer more settings options under the **Settings...** dialog. 0 means no compression while 9 is the highest compression level. High compression can considerably reduce the size of the project files, but results in longer saving and loading times.
5.2.4 View

The View menu gives the user the possibility of personalizing the user interface (s. Figure 27). Using the Toolbars and Methods submenus the user can choose which individual components of the interface are shown or hidden (s. Figure 28).

The Simplified function in the View→Methods menu allows to swap between a state where all parameters are displayed and a state where only the most relevant parameters are shown to reduce the complexity of the user interface.

5.2.5 Tools

The Tools menu (s. Figure 29) provides a collection of miscellaneous useful functions:

Group Tissues...: This function can be used to merge different tissues into a common tissue (e.g., if multiple bones have been segmented as separate tissues and should be merged into a common ‘Bone’ tissue) or to reassign tissues. For this a text file has to be provided that contains lines with two integer numbers each. Regions currently assigned to the tissue with the first number will be reassigned to the tissue with the second number.

Target→Tissue: This function is of special interest to convert segmented image data (e.g., data available in raw format which has been loaded into iSeg). It assumes that the grey level value corresponds to the tissue number and assigns the corresponding pixels to that tissue. If grey values
Figure 28: The **Toolbars** and **Methods** submenus.

Figure 29: The **Tools** menu.

exceed the number of existing tissues, new tissues are created up to a maximum of 250 tissues.

**Target → Tissue grouped...** : This function corresponds to the execution of the **Group Tissues** function followed by the **Target → Tissue** function.

**Tissue → Target**: This function performs the inverse of the **Target → Tissue** function.

**Inverse Slice Order**: This function can be used to inverse the order the slices.

**Clean Up**: This function can be used to remove islands, speckles and holes to clean up a project. It can be memory intensive. A rate and a pixelsize
have to be specified. All structures smaller than the pixelsize and smaller than the total size of the volume attributed to the same tissue divided by the rate are removed and replaced by neighboring tissues or background.

**Smooth Steps**: This function can be used to clean up a segmentation that shows irregular segmentation between neighboring slices (due to the sliced nature of the images and the often 2D segmentation procedure). It can be used to smooth steps and fill gaps along the z direction.

### 5.2.6 Atlas

It is possible to create and load reference image data with associated tissue distributions. This is called an ‘Atlas’. Atlases can be used as orientation during the segmentation process. Atlases of the leg, knee, pelvis, thorax and head region are distributed along iSeg. Any atlas in the atlas subfolder of the iSeg installation is available under the Atlas menu. If the content of the atlas folder changes, **Update Menu** has to be activated to update the Atlas menu. Selecting an atlas results in the opening of an atlas viewer. The atlas viewer shows an overlay of image data and segmentation data. The contrast and brightness of the image data can be modified with two sliders above the image. Scrolling through individual slices is provided by a scroll bar below the image. It is possible to select the cutting orientation (x, y or z) of the visualization plane. The transparency of the tissue information can be varied with a slider to show more or less of the image data. The usual zooming functionality is provided and a string at the bottom indicates the tissue currently under the cursor.

**Create Atlas...**: Based on the current image data, tissue list and tissue distribution a new atlas is generated at a user defined file location. Atlases stored in the atlas subfolder of the iseg directory are visible in the Atlas menu.

**Update Menu**: All the Atlases (files ending on .atl in the atlas subfolder of the iSeg installation are made available in the Atlas menu under their filename.

### 5.2.7 Help

The **Help** menu (s. Figure 30) provides the following functions:

![Help Menu](image)

Figure 30: The Help menu.
About: This function displays the version number of the used iSeg executable.

5.3 Viewers

5.3.1 2D Viewers

The Source and Target viewers are 2D viewers (s. Figure 31). The segmented tissue distribution can be overlayed as color information (Show Tissue option). The transparency of the color information can be specified tissuewise (Opacity value set in the Mod. Tissue dialog). The brightness and contrast of the image displayed in the viewer can be modified using the $\mathbb{R}$ and $\mathbb{Q}$ sliders respectively. This affects only the displaying and not the underlying image data (contrary to the contrast and brightness slides in the Image → Contr./Bright. dialog). Zooming of the image displayed in the viewers can be performed. It is possible to display interfaces between different grey values of the Target image as contours on top of the Source image (Show Outlines option). This is useful for interactive segmentation techniques or postprocessing steps, where one wants to ascertain that the segmentation boundaries correspond with what the user recognizes as boundaries of the underlying image data (e.g., when the brush is used to correct a border). The colour of the outline is based on the colour of the tissue currently selected in the tissue list. When cross-section viewers have been opened (using Image → x Sliced or y Sliced, s. 5.2.2), it is possible to highlight the lines where the currently displayed cross-section slices intersect the displayed image using the Show Crosshair option.

Right clicking on the 2D viewers opens up the context menu (s. Figure 32). It contains the following functions:

Add Mark: This allows to add a marker of the tissue type currently selected in the tissue list. Some segmentation techniques such as the watershed transformation are based on these markers. Markers are displayed in the 2D viewers as $\mathbb{R}$ on top of their location.

Add Label: This allows to add a label of the tissue type currently selected in the tissue list. A name must be assigned. The vessel segmentation and the measurement functions are based on these labels. Furthermore the labels can be used to place small notes on specific regions of the image. Labels are displayed in the 2D viewers as $\mathbb{R}$ on top of their location followed by the label-name.

Remove Mark: The Remove Mark function removes a mark or label below the location (within 5 pixels) where the mouse click occurred.

Clear Marks: This function clears all marks and labels on the current
Figure 31: The **Source** and **Target** 2D viewers. The user can adjust the displayed brightness and contrast using the respective sliders and display interfaces between different grey values of the **Target** image as contours on top of the **Source** image with the **Show Outlines** option.

**Select Tissue:** The tissue currently under the cursor is selected as active tissue in the tissue list.

Figure 32: The **context** menu of the **Source** 2D viewer.

The following functions are only available in the **Target** image context menu (s. Figure 33) and are essentially short cuts to some of the functionality provided by the buttons (+, -,...) on the bottom right (s.).

**Add Tissue:** The regions on the current slice having an identical grey value in the **Target** image as the pixel where the context menu has been opened are added to the tissue currently selected in the tissue list.
**Subtract Tissue**: The regions on the current slice having an identical grey value in the Target image as the pixel where the context menu has been opened are removed from the tissue currently selected in the tissue list.

**Add Tissue 3D**: The regions on all slices having an identical grey value in the Target image as the pixel where the context menu has been opened are added to the tissue currently selected in the tissue list.

**Add Tissue Conn**: The regions on the current slice having an identical grey value in the Target image as the pixel where the context menu has been opened are added to the tissue currently selected in the tissue list, provided they are part of a connected region containing the selected pixel. This can be used, e.g., when a ‘Rib’ tissue should be generated on a slide where all bones have been segmented in the Target image.

**Add Tissue Larger**: The regions on the current slice having a grey value in the Target image that is identical or higher (whiter) than the grey value of the pixel where the context menu has been opened are added to the tissue currently selected in the tissue list. This is a short cut to first performing thresholding and then adding performing Add Tissue.

![Figure 33: The Context menu of the Target 2D viewer.](image)

### 5.3.2 Cross-Section Viewers

The cross-section viewers can be opened by selecting the x Sliced or y Sliced function in the Image menu. This opens a viewer displaying cross-sections orthogonal to the x or y direction. Either the Source or the Target image can be displayed and the tissue distribution can be overlayed in colour.
The viewer window can be resized and it stays on top while open to allow displaying it while performing operations in the main iSeg window. A slider allows scrolling through the different slices. The Show zpos option permits to indicate by a green line the location of the slice currently displayed in the main window viewers. The Show xypos option permits highlighting the currently displayed slice in the y-slice-viewer if it has been opened. The image magnification is affected by the zoom factor specified in the main iSeg window. Closing the slice-view window does not result in iSeg being terminated. Two cross-section viewers (orthogonal to x and y direction) can be opened simultaneously.

### 5.3.3 3D Viewers

3D volume viewers can be opened by selecting the **3D volume view source** or **3D volume view tissue** function in the Image menu. This opens a viewer that can display a 3D view of the Source images or of the tissue distribution. In the Source viewers, a high grey value region is rendered as a bright opaque volume, while a low grey value results in a transparent and dark region. In the tissue viewer, the tissues are rendered with a colour and opacity that can be modified using the Mod. Tissue dialogue (s. 5.4). It is possible to display up to two cross section planes in addition to the volume rendering. These planes can be oriented and moved arbitrarily, by dragging their orientation arrow and frame. The displaying of the individual planes and the volume can be deactivated. The volume rendering can be enhanced by shading, which increases the plasticity but can reduce rendering speed significantly. Volume rendering can be performed either using ray-tracing (faster, activated when Ray Trace / Texture is checked) or texture mapping (slower). The 3D volume viewer window can be resized.

Besides the 3D volume viewers there exist as well 3D surface viewers. 3D surface viewers can be opened by selecting the **3D surface view** or **3D isosurface view** function in the Image menu. This opens a viewer that can display a 3D surface view of the segmented tissue or an iso source of the Source images. In the tissue surface viewer the colour and transparency settings are defined by the tissue settings and can be modified by changing the opacity value or the RGB values in the Mod. Tissue dialog (s. 5.4). In addition, the overall transparency can be changed with a slider in the 3D surface viewer. In the isosurface viewer the grey level at which the isosurface is extracted can be selected with a slider and another slider influences the transparency of the rendered surface. The 3D volume viewer window can be resized.
5.4 Tissues

The segmentation process aims at assigning regions to specific tissues. The list of tissues currently available is shown on the right of the user-interface. The last used tissue list will be reloaded when iSeg is opened and is available for a new project. When a project is stored, the tissue list is part of the project file and will be available when the project is loaded again. The current tissue list (incl. colours...) can be exported into a readable ASCII file using the **Save Tissues** function in the **File** menu. A file list can be imported using the **Load Tissues** function in the **File** menu. This creates tissues with colour and opacity settings as defined by the specified file. Already existing tissues will not be erased. The default list of tissues available when iSeg is started can be set using the **Set Tissuelist as Default** function in the **File** menu. This updates the file called **def_tissues.txt** in the **tmp** subfolder of the iSeg installation which specifies the default tissue list.

To protect specific tissues from being overwritten or erased, it is possible to lock selected or all tissues by checking the box next to the lock symbol. All tissues can be locked simultaneously by activating the **All** button next to it. The **New Tissue** button below it can be used to generate a new tissue. For this a name and the values of the red, green and blue component of the colour have to be specified. Furthermore, an opacity setting is available. This setting has an impact on the various viewers. It controls the transparency of solids in the 3D tissue viewer (s. 5.3.3) and of the overlayed tissue information in the 2D **Source** and **Target** image viewers. The name, colour and opacity of the selected tissue can be modified using the **Mod. Tissue** button. The colour of currently selected tissue is used to display region boundaries of the **Target** image on top of the **Source** image when the **Show Outlines** option of the **Source** image viewer is activated. The selected tissue can be removed from the tissue list (resulting in the erasing of all regions assigned to that tissue) using the **Remove Tissue** button.

The **Get Tissue** function copies the regions assigned to the currently selected tissue into the **Target** image, where they are available for additional processing. The **Clear Tissue** button erases the region(s) assigned to the selected tissue from the tissue distribution model (without removing the tissue from the tissue list). **All** next to **Remove Tissue**, **Get Tissue** and **Clear Tissue** does the same for all tissues. The **3D** option on on top of the **Get Tissue** function controls whether the **Get Tissue**, **Clear Tissue** and **Clear Tissues** functions are applied to a single slice or to all slices.

Homogeneous regions (constant grey level) in the **Target** image can be added or removed from the tissue distribution using the buttons at the bot-
tom right of the user interface. The + button can be used to add a region to
the currently selected tissue by subsequently clicking on the Target image.
++ performs a similar action, but allows the addition of multiple regions
using multiple selections in the Target image. The adding mode is deacti-
vated again by pressing ++ a second time. The - button can be used to
subtract a region from the currently selected tissue by subsequently clicking
on the Target image. – performs a similar action, but allows the subtrac-
tion of multiple regions using multiple selections in the Target image. The
subtraction mode is deactivated again by pressing – a second time. The
3D, Conn. and Override options can be used to modify the behavior of
the +, ++, - and – button. When checked, the 3D option defines
that the addition or subtraction is not limited to the currently displayed
slice. All regions of the Target images on all slices will be added/subtracted
from the tissue distribution if they have the same grey value as the selected
point. The Conn. option restricts the adding/subtracting to the region
connected to the selected point. Other regions with the same grey value in
the Target image (e.g., speckles or regions that have been segmented as well
by the selected segmentation method due to similar image features but that
do not belong to the selected tissue) will not be added to the tissue. Conn.
works in both 2D and 3D (as specified by the 3D option). Override can
be used to steer, whether regions already assigned to another tissue should
be reassigned. However, locked tissues will not be overwritten, even when
Override is active. Some of this adding and subtracting functionality can
be access as well through the context menu obtained by clicking the right
mouse button on the Target image viewer (s. 5.3.1).

5.5 Tabs

There are multiple tabs containing segmentation, measurement, pre- and
postprocessing functions (s. Figure 37). These functions have been grouped
based on common functionality, common interfaces or shard algorithms. A
specific tab can be selected by pressing the corresponding button on the
bottom left of the user interface. Depending on the currently selected tab
and method, the interaction modes with the image viewers change (e.g.,
mouse clicks vs. dragging...). Each tab displays the parameters and op-
tions related to the specific functions that can be activated from the current
tab or based on the mode determined by the currently selected tab. The
View→Methods menu allows to customize the visibility of individual tabs
(these settings are stored when iSeg is closed and restored the next time it is
started). The Simplified function in the View→Methods menu allows to
swap between a state where all parameters are displayed and a state where
Figure 34: The list of available tissues and the buttons that allow editing of the tissues and tissue regions.

only the most relevant parameters are shown to reduce the complexity of the user interface.
The following sections will provide more detailed information about the different tabs and segmentation techniques.

### 5.5.1 Thresh

The **Thresh** tab contains thresholding techniques. These techniques are well suited, when there are tissues with a clearly distinguished grey level range or that there is a clear anatomical distance to other tissues with simi-
lar grey values. There can be multiple thresholds to identify multiple regions (e.g., air, muscle, fat and bones in CT images) in one step. It is possible to specify the grey level boundary values manually (s. Figure 38) or automatically using one of three different automatic techniques.

Figure 38: The **Manual** method of the **Thresh** tab.

It is possible to save manually determined thresholds in a file and to load them again using the **Save** and **Load** button. This can be specially valuable for image modalities, where specific tissues are known to have a constant value range (e.g., a Hounsfield unit range in CT images).

The **Histo** technique (s. Figure 39) determines the number of separable grey level classes and the ideal thresholds by analyzing the histogram of the image (The histogram is made with 256 bins and is slightly smoothed using Gaussian smoothing). It tries to find ‘valleys’ in the histogram indicating that there are separable grey level classes. It considers having found a significant valley when a minima in the histogram is lower than the **Ratio** setting multiplied with the height of the last maxima. In addition, the **Min. Pixels** setting can be used to put a limit on the minimal size of local maxima to avoid introducing to many thresholds in grey scale intervals where only very few pixels exist. The number of identified classes will be reduced, when **Ratio** is reduced or **Min. Pixels** is increased. It is possible to restrict the region contributing to the histogram to a rectangular subregion using the **Subsection** option and specifying the corner coordinate and subsection extent. This allows to perform the classification based only on the region of interest, neglecting for example noise in the background.

The **k-means** and **EM** (Expectation Maximization) method (s. Figure 40) require a predefined number of tissue classes and use clustering algorithms to find the ideal thresholds (s. [4, 12]). Two parameters have to be provided when using these methods: **#Iterations** specifies the maximum number of iterations performed by the clustering algorithm. **Convergence** is a parameter that is used to determine, when the clustering has converged. This is considered to have occurred when less than the specified number of
points have been reclassified in the last iteration. The k-means and EM techniques can work with multidimensional data.

Apply to all slices allows to apply the identified thresholds to all slices. The automatic detection of the thresholds, however, is always performed based on the currently selected slice.

5.5.2 Growing

Growing is similar to thresholding, but will only segment the regions connected to the seed point(s) (basic region growing algorithm). An Upper and a Lower grey value threshold must be specified that limit the grey values of regions that can possibly be assigned to the segmented tissue. Either a seed point is provided by clicking on the Source image (depending on the mouse location, different regions will be segmented) or seed points are identified automatically. For this, AutoSeed must be checked and the subrange of the selected grey value range must be specified (second Lower and Upper slider; the value of zero corresponds here to the grey value selected by the Lower slider of the main thresholds and the value hundred to the grey value selected by the Upper slider of the main thresholds). Seeds are then automatically generated at all image points with a grey value in the subrange.
and all regions connected to these seeds with a grey value in the main range are identified. This is equivalent to hysteretic growing.

The segmentation is applied/updated when a slider is moved, a mouse click is performed (if AutoSeed is not selected) or the Execute button is pressed (e.g., to apply a setting optimized on one slice to another slice).

In case the segmentation results in an unwanted connection between the intended region and another region of similar grey value, it is possible to manually draw limits using the Draw Limit functions. After pressing the button, the limit can be freely drawn on the Source image. Multiple limits can be drawn (even separately on different slices). A wrongly drawn or unwanted limit can be removed using the Clear Limit button followed by a click on the specific limit.

Growing without automatic seed placement can be performed in 3D by checking the 3D option. Here the possibility of placing multiple limits on different slices is valuable. If seeded growing is used, the Apply to all slices option allows to apply the seeded growing with identical parameters to all slices.

Thresholds (and the sub-thresholds when using AutoSeed) can be saved and reloaded using the Save and Load buttons.

Figure 41: The Growing tab.

5.5.3 Contour

The Contour tab (s. Figure 42) offers various contouring methods with a varying degree of algorithmic support.

The Free contouring variant requires the user to fully manually contour a region. The contour will start at the point where the left mouse button is pressed down and it will follow the mouse pointer movement until the left mouse button is released. The contour is then closed by a straight line.

The Straight contouring allows the user to draw a polyline. Additional points are added by clicking the left mouse button. Successive removing of
unwanted points is obtained by clicking the middle mouse button. A double left click closes the contour while a double middle click aborts the line drawing process.

**Auto Trace** uses an intelligent scissor (or live-wire, s. [5, 8, 7]) algorithm to automatically identify the ideal contour path. This algorithm uses information about the strength and orientation of the (smoothed) image gradient, the zero-crossing of the laplacian (for fine tuning) together with some weighting to favour straighter lines to determine the most likely contour path. The contouring is started by clicking the left mouse button. Each subsequent left button click fixes another point and the suggested contour line in-between. Successive removing of unwanted points is obtained by clicking the middle mouse button. A double left click closes the contour while a double middle click aborts the line drawing process. The **Freezing Delay** option allows the user to specify the number of seconds after which a line segment is frozen even without mouse click if it has not changed.

The **Close Contour** option allows to switch on and off the dynamic drawing of the straight line that would currently be used to closes the curve.

![Contour tab](image)

Figure 42: The **Contour** tab.

### 5.5.4 IFT

The **IFT** tab (s. Figure 43) offers access to an ‘Image Foresting Transformation’ (s. [9]) based segmentation technique. This method lets the user draw lines freely into different tissues. These lines are drawn with the colour of the currently selected tissue. Multiple lines of different colours can be drawn and they are subsequently used as seeds to grow regions based on a local homogeneity criterium. Through competitive growing the best boundaries between regions grown from lines with different colours are identified.

The colours are needed to specify the multiple objects. However, a colour does not mean, that the identified region will automatically be assigned to the corresponding tissue. To assign a region to a tissue the normal addition
procedure must be performed (s. 5.3.1 and 5.4). The competitive aspect increases the robustness. The ‘Image Foresting Transformation’ based method has various advantages:

- Through the competitive growing boundaries can be identified between tissue even if the boundary itself is vague.

- Multiple regions can be segmented in a single segmentation process with each additional region benefiting from the already marked regions (ideally, only one additional line is needed to mark a new tissue).

- It is not necessary to point or click at some precise location (contrary, e.g., to the live-wire method which needs clicks on points that have to lie exactly on the boundary). Instead, the lines can be drawn freely into the target regions. (Drawing long lines through large parts of the regions will improve the performance of the algorithm.)

- The method is very interactive. Every additionally drawn line will update the segmentation automatically. If some region-part has been wrongly assigned, it is enough to draw a line with the correct colour into the wrongly assigned part to correct it.

**Clear Lines** can be used to erase all drawn lines on the current slice (not on other slices). **Remove Line** followed by a click on a line deletes this line and automatically updates the segmentation. If **Remove Line** was pressed accidentally, a second press will deactivate the function.

A slider is available, that can be used to modify the homogeneity criterium used to identify regions that should be connected. If the slider is at a high value (right), regions are likely to grow as far as their competition with other growing regions permits. If it is at a low value (left), regions might stop growing earlier if they meet a clear border (the definition of ‘clear’ depending on the position of the slider). The segmentation is automatically updated, when the slider is moved.

### 5.5.5 Fuzzy

The **Fuzzy** tab actually gives access to two different segmentation techniques that have a very different background but a similar user- and interaction interface: fuzzy connectedness [18, 28] (**Fuzzy Connect.**) and a fast marching implementation of a level-set technique [25, 16] (**Fast Marching**).

Both techniques require a start point to be specified.
• **Fuzzy Connecteness**: Fuzzy connectedness computes for each image point the likelihood of it belonging to the same region as the start point. It does this by looking at each possible connecting path between the two points and assigning the image point probability identical to the probability of this path being entirely in the same tissue as the start point. The variant from [18] is implemented. It takes three parameters: \( m_1 \) should be the average grey value of the region to be segmented, \( s_1 \) is a measure for how much the grey values are expected to deviate from \( m_1 \) in region to be segmented (something like a standard deviation) and \( s_2 \) is a criterion from local homogeneity (sudden changes larger than \( s_2 \) are considered to indicate boundaries). All points with an assigned likelihood above a threshold-probability are then considered to be part of the segmented region.

• **Fast Marching**: Level-set techniques as employed in iSeg describe the evolution of a line (boundary) on a 2D image in time by instead calculating the evolution of a field defined on the image and looking at the line where the field crosses zero. This allows flexible and accurate handling of merging and splitting contours, sharp edges. When the line is continuously expanding (never shrinking) it is possible to use fast-
marching variants where the arrival time of the zero crossing is calculated instead of the field evolution over time. This considerably speeds up calculations. iSeg implements a generalized level-set method and a fast marching method with different image based and shape (smoothness of contour) based forces. Only a fast marching method is currently accessible through the GUI. It assumes an expanding (balloon) force and an expansion delaying force which depends on the local image gradient. Slight Gaussian smoothing (with parameter $\text{Sigma}$) is used to reduce the impact of noise. The parameter $\text{Thresh}$ can be used to tune how strong a gradient must be to significantly affect the curve expansion. A spinbox specifies the upper range of the $\text{Thresh}$ slider. All points inside the curve are considered to be part of the segmented region.

![Figure 45: The Fast Marching method of the Fuzzy tab.](image)

Both the fuzzy connectedness and the fast marching technique allow the user to dynamically adjust the size of the identified region. This corresponds to setting the probability threshold in the fuzzy connectedness method and the arrival time in the fast marching method. Two possibilities exist to set the region extent: **Drag** allows the user to drag the mouse (after clicking on the start point and keeping the mouse button pressed down) specifying the extent of the region on the image. **Slider** permits to use a slider to increase or decrease the region size.

### 5.5.6 Watershed

The **Watershed** tab (s. Figure 46) gives access to an interactive watershed transformation segmentation method [11]. For this the gradient of the slightly smoothed image is calculated. High values are interpreted as mountains and low values as valleys. Subsequently, one simulates the flooding with water and whenever water from two adjacent basins meets a dam is built. This results in the division of the image into tens of thousands of basins and
a merging order for these basins. The user can now set markers (using the context menu obtained by clicking the right mouse button on the Source or Target image) which together with the merging order steer how the basins are connected to homogeneous regions. New markers obtain the colour of the currently selected tissue. The colours are needed to specify the multiple objects. However, a colour does not mean, that the identified region will automatically be assigned to the corresponding tissue. To assign a region to a tissue the normal addition procedure must be performed (s. 5.3.1 and 5.4). Sometimes multiple marks are required to identify a region. Even weak boundaries can be identified by putting markers belonging to different tissues on both sides of the boundary. If some region-part has been wrongly assigned, it is enough to set a mark with the correct colour in the wrongly assigned part to correct it.

Just like the IFT method (s. 5.5.4), the Watershed method has various advantages:

- The competitive aspect of the method allows to identify boundaries between tissue even if the boundary itself is vague.

- Multiple regions can be segmented in a single segmentation process with each additional region benefiting from the already marked regions (ideally, only one additional mark is needed to create a new tissue).

- It is not necessary to point or click at some precise location (contrary, e.g., to the live-wire method which needs clicks on points that have to lie exactly on the boundary). Instead, the marks can be set freely into the target regions.

- The method is very interactive. Every additional mark will update the segmentation automatically. If some region-part has been wrongly assigned, it is enough to add a mark with the correct colour in the wrongly assigned part to correct it.

Marks can be removed using the context menu (s. 5.3.1).

The flooding height can be specified using the h slider. The spinbox next to the slider allows to set the maximal value of the slider. A high h value will result in the entire image being divided into regions based on the marks, while a low value will stop the flooding when strong gradients (clearly corresponding to boundaries) are encountered. Thereby background and internal air or bones can be excluded automatically.

The Execute button allows to perform watershed based segmentation after changing to a new slice without having to add a mark.
5.5.7 OLC

The OLC tab contains OutLine Correction routines that can be used to modify the result of a segmentation operation and to correct frequently occurring issues. It contains methods to remove speckles, close holes and gaps, add skin... It is possible to correct regions in the Target image or to perform corrections directly on the tissue distribution (TargetPict or Tissue option). If the TargetPict option is selected, the Select Object button allows the user to select the grey level of the region that should be modified. This has to be selected first to prevent unwanted behavior. If the Tissue option is active, the modifications will affect the tissue currently selected in the tissue list. For some techniques the Apply to all slices option can be checked to specify that the operation should be applied to all slices.

The following correction modes are available:

- **Outline Corr**: The user can draw an alternative boundary segment for a region. This segment will be connected to the region using the shortest possible lines and will replace the boundary segment between the connection points. This can be used to extend or reduce a region locally. It cannot be used to join multiple disconnected regions (s. Figure 47).

- **Brush**: This is the most manual correction and segmentation tool available. It provides the user with a brush of selectable Radius (in pixels). When the Draw mode is active, this brush will draw the currently selected grey value (TargetPict) or tissue assignment (Tissue). When the Erase mode is selected, the brush can be used to erase a region leaving black (resp. no tissue assigned) behind. The Modify mode depends on where the mouse is pressed down. If it is pressed down in the selected region/tissue, it acts as a drawing brush. When
Figure 47: The **Outline Corr** method of the **OLC** tab.

It is pressed down outside, it will overwrite the selected region/tissue with the grey value or tissue assignment of the point when the mouse has been pressed down (s. Figure 48).

Figure 48: The **Brush** method of the **OLC** tab.

- **Fill Holes**: This function can be used to close all holes in the selected region/tissue that have a size smaller than the number of pixels specified by the **Hole Size** option. It will be executed when the **Fill Holes** button is pressed (s. Figure 49).

Figure 49: The **Fill Holes** method of the **OLC** tab.
• **Remove Islands**: This function can be used to remove all islands (species, outliers) with the selected grey value or tissue assignment that have a size smaller than the number of pixels specified by the Island Size option. It will be executed when the **Remove Islands** button is pressed (s. Figure 50).

![Figure 50: The Remove Islands method of the OLC tab.](image)

• **Fill Gaps**: This function allows to close gaps between multiple disconnected regions having the same grey value or belonging to the same tissue. The user can specify the largest size of gaps that should still be closed with the Gap Size option. It will be executed when the **Fill Gaps** button is pressed. The **Fill Gaps** function will only close gaps between disconnected regions but does not affect gaps due to a thin filament reaching into a connected region. This allows to conserve wanted fine outline features while closing unwanted gaps (s. Figure 51).

![Figure 51: The Fill Gaps method of the OLC tab.](image)

• **Add Skin**: This function can be used to add a skin to a finished segmentation. The skin thickness (in pixels) can be specified with the Thickness option. The skin will be assigned the currently selected grey value or tissue and will be place around all regions having a nonzero grey value or a tissue assignment. The skin can be added outside
the segmented regions - extending the model volume - or inside the segmented regions (Outside/Inside option). When it is added inside in the Tissue mode, it will only overwrite tissues that are not locked. The function will be executed when the Add Skin button is pressed. When the Apply to all slices option is checked, it does actually not only perform the skin adding on all slices - rather, the skin is added in a 3D manner around the whole volume covering as well the surfaces in ‘z’ direction (s. Figure 52). When Apply to all slices is selected, the skin thickness can be specified in mm instead of pixels. If the pixelsize and thickness is inhomogeneous, the thickness in pixels can then be different along the x,y and z orientation.

Figure 52: The Add Skin method of the OLC tab.

- Fill All: This function can be used to make sure that there are no unassigned regions inside the segmented model. As body cavities are typically assigned to specific regions (e.g., trachea-lumen) and not left unassigned, this is a valuable function that makes certain, that no holes remain in the model that would for example be segmented as ‘Background’ in SEMCAD X. The function will be executed when the Fill All button is pressed (s. Figure 53).

Figure 53: The Fill All method of the OLC tab.
- **Adapt**: This function is still in experimental stage and is currently not performing in a stable manner. It extracts the outlines of the currently selected regions, fixes selected points along these outlines, uses the livewire technique to connect these points (s. 5.5.3) and fills the newly drawn contour. The function will be executed when the Adapt button is pressed. It can be used after having performed an interpolation step (s. 5.5.8) to adapt the interpolated regions to the Source images (s. Figure 54).

![Figure 54: The Adapt method of the OLC tab.](image)

**Fill Holes** and **Remove Islands** are particularly useful to reduce noise in the segmentation, simplify it and reduce the number of disconnected regions.

### 5.5.8 Interpol

The **Interpol** tab (s. Figure 55) offers image interpolation and extrapolation. Topologically flexible interpolation is performed [21, 13]. The topological flexibility means that interpolation can automatically generate a branching if it detects two subregions that should joint into a common region. This is achieved by using a distance transformation on the original slices and inter-/extrapolating the distance values pointwise to determine the assignment of the point on any other slice (using the zero-crossings of the inter-/extrapolated distance transformations). Multiple objects (with different grey values) can be interpolated jointly without introducing gaps or overlap (e.g., following segmentation with multiple thresholds or IFT...). Interpolation and extrapolation are particularly helpful when a tissue doesn’t undergo major changes on a few consecutive slices (e.g. in limbs or for vessels). The inter-/extrapolation can be used both on the Target image or the Tissue distribution (TargetPict or Tissue option). When Tissue is selected, the inter-/extrapolation is used to estimate the distribution of the currently selected tissue. The result will, however, not be directly added to
the tissue distribution. Instead, it will be displayed in the Target window for further editing (e.g., to further correct the outlines using methods from the OLC tab, s. 5.5.7). When TargetPict is selected, the inter-/extrapolation is based on the Target pictures. Interpolation is performed based on the grey value for all regions jointly and on all slice between the Start Slice and the current slice. Extrapolation is performed assuming all non-zero points to belong to the same region (as extrapolation of multiple adjacent grey regions can result in conflicting assignments).

The condition for identifying two regions on different slices as belonging together, resulting in an inter-/extrapolation between them is that they partially overlap.

Both interpolation and extrapolation are based on two slices. **Start Slice** must be pressed to select the first one and **Execute** must be pressed on the second slice. Interpolation will the automatically interpolate the intermediate slice(s) while extrapolation will determine the image or tissue distribution on the slices specified by **Target Slice**.

![Figure 55: The Interpolation method of the Interpol tab.](image)

### 5.5.9 Smooth

The **Smooth** tab offers various smoothing functions. This functions can be used prior to segmentation to increase the homogeneity of regions that should be recognized, thereby increasing the effectiveness of many segmentation methods. Different smoothing functions are available. Some of them smooth everything, while others are designed to retain significant edge features. The filters are applied when the **Execute** button is pressed or sliders are moved. The **Apply to all slices** option allows to apply a filter with the specified parameters to all active slices.

The smoothed image is based on the **Source** image(s) and written into the **Target** image(s). To use them as a basis for segmentation, the or button must be pressed to shift them back into the **Source** image (s. 5.1).
The following smoothing filters are available:

- **Gaussian**: This applies a Gaussian smoothing filter (convolution with a Gaussian kernel). The width of the Gaussian is specified using the `Sigma` parameter (s. Figure 56).

  ![Figure 56: The Gaussian of the Smooth tab.](image)

- **Average**: This assigns each point the average grey value of the region of size $n \times n$ around it. $n$ can be specified by the user (s. Figure 57).

  ![Figure 57: The Average of the Smooth tab.](image)

- **Median**: This assigns each point the median of all points in its neighbourhood (incl. own pixel). The median is often less sensitive to noise than the average (s. Figure 58).

- **Sigma Filter**: This filter is identical to the Average filter but only considers neighbourhood pixels with a grey value deviating by a maximum of `Sigma` from the central pixel value. This allows to reduce the smoothing of edges (s. Figure 59).

- **Aniso. Diff.**: This is the most sophisticated smoothing filter (and it can be slow especially when acting on all slices). It applies diffusion smoothing, but only to regions where the gradient is below a certain
value. Above that value, the result is rather an increase of the edge sharpness. This threshold can be modified using the k slider (the upper value of the k slider can be adjusted in a spinbox next to it). dt is the time-step of the smoothing process. A low value will increase the robustness of the smoothing procedure, but will increase the number of required iterations (specified under Iterations) and hence the execution time. A restraining force can be specified (Restrain) to have a continued impact of the initial image on the smoothing process. This will reduce and delay the smoothing effect, but will prevent the smoothing process from drifting too far away from the original image. If more diffusion smoothing is required, Cont. Diffusion can be pressed to add more iterations.

5.5.10 Morpho

The Morpho tab (s. Figure 61) offers access to simple morphological functions. They are based on expanding or shrinking (Dilate/Erode) regions by a given number of pixel layers (n). Growing followed by shrinking results in the closing of small (< 2n) gaps, bays and holes. This operation is called
Close. First shrinking before growing is called **Open** and results in the deletion of small islands and thin links between structures. The **Close** function is similar to the execution of the **Close Gaps** and the **Fill Holes** function in the **OLC** tab (s. 5.5.7), but at the same time it does not preserve thin bays that might be wanted features. The functions are applied once **Execute** is pressed and they act on the **Target** image.

The expanding and shrinking are performed by adding or removing pixels wherever two neighboring pixels one belongs to the region and one does not. Two different definitions of neighbor are available: When **4-connectivity** is selected, only the four touching pixels (left, right, front, back) are considered neighbors. When **8-connectivity** is selected, the four pixels diagonally bordering are considered neighbors as well.

**5.5.11 Edge**

This tab (s. Figure 62) gives access to various edge extraction routines. These are mostly useful as part of other segmentation techniques. However, occasionally the user might want to detect edges or use this tab to find optimal parameters for other techniques that employ edge detection.

The following edge detection/highlighting techniques are available:
- **Sobel**: A sobel filter is applied to highlight edges. The sobel filter convolutes the image with sub of two filters, one of which highlights edges in \( x \)-direction and one of which highlights edges in \( y \)-direction. Both of them are essentially gradient filter. The result is a grey-scale distribution in the **Target** image with high values corresponding to likely edges.

- **Laplacian**: The laplacian (measure for image curvature) of the image is calculated using convolution with a suitable kernel. The result is a grey-scale distribution in the **Target** image with high values corresponding to likely edges.

- **Interquart.**: The difference between the maximum and the minimum value in the neighborhood of each point is determined. A high value indicates a likely edge.

- **Moment**: The second moment of the pixel vicinity is calculated. This is the average square of the difference between the vicinity values and the local average.

- **Gauss**: This filter calculates the difference between the image and a Gaussian smoothed image (with a width defined by **Sigma**). This will result in non-zero values at edges while homogeneous regions will have values close to zero.

- **Canny**: The Canny edge detector is based on a sobel filter applied to a smoothed image (Gaussian smoothing with width **Sigma**). In addition, the likely edge orientation is determined by looking at the \( x \)- and \( y \)-direction Sobel filters. Only the highest local value in the direction perpendicular to the local edge orientation is considered while all non-maximum points are set to zero. This results in thin and well localized edges. Subsequently, all values above an upper threshold (**Thresh high**) are identified as edge points and all points with a value between **Thresh low** and **Thresh high** are considered as edges if they are connected by points with a value above **Thresh low** to a point with a value above **Thresh high** (hysteretic thresholding). This identifies clear edges and considers all potential edge candidates as edges if they are connected to a previously identified edge. The result is a binary image with clearly localized edges.

- **Lapl. Zero**: The laplacian of the smoothed image (Gaussian smoothing with width **Sigma**) is determined and its zero crossing locations are treated as potential edges. If the local gradient has a value above
Thresh, the point is identified as edge. The result is a binary image with clearly localized edges.

Execute must be pressed to apply the edge detection filter. The ‘Canny’ edge detector is the most advanced algorithm and should typically be used.

Figure 62: The Edge tab.

5.5.12 Feature

The Feature tab (s. Figure 63) can be used to obtain information about the grey value and tissue distribution. A rectangular area is marked by pressing down the left mouse button and moving the mouse. The following information about the Source and Target image distribution is then dynamically updated and shown in two columns.

Average shows the average grey value of the marked rectangle and Std Dev. shows the standard deviation of the grey value in the rectangle. Minimum and Maximum show the extreme grey values in the rectangle. Grey val. shows the grey values of the point under the cursor in the Source and Target images. Coord. shows its coordinate both in pixels and in mm (in parenthesis). In addition the tissue assignment (name) and corresponding index of the point below the cursor is displayed (Tissue).

The feature tab is particularly helpful when parameters for different segmentation methods (e.g., fuzzy connectedness - s. 5.5.5) have to be estimated.

5.5.13 Measurement

This tab (s. Figure 64) offers the possibility to perform different types of distance, angle or volume measurements:

- Vect.: The vector between the start- and end-point is displayed.
- Dist.: The distance between the start- and end-point is displayed.
Figure 63: The **Feature** tab.

- **Angle**: The angle between the line from the first to the second point and the line between the second and the third point is displayed.

- **4pt-Angle**: The dihedral angle of four consecutive points is displayed.

- **Volume**: The total volume of all regions (3D) having the same Target image grey value as the point which has been clicked is displayed together with the name and the volume of the tissue on which the user has clicked.

All measurements consider the pixel size and slice thickness. They are in $\text{mm}$ (resp. $\text{mm}^3$), not pixels. Two different possibilities exist to select the points: If **Clicks** has been selected, the points are defined by clicking on the **Source** or **Target** image and the order of the clicks determines the point order. If **Label** is selected, the names of user defined labels (s. 5.3.1) can be selected in drop-down lists and the order of the points is the order of the lists. The second possibility is only available, if labels have been set by the user.

Figure 64: The **Measurement** tab.

60
5.5.14 Vessel

This tab (s. Figure 65) offers access to a vessel segmentation technique. The method is still experimental and not yet robust. It is designed to segment contrast enhanced CT images (the vessels appear bright). The segmentation technique is based on the live-wire method (s. 5.5.3). The ‘best’ path is selected, where the length, the straightness and, most importantly, the grey value is considered. Values between 1150 and 1250 are the preferred values. However, values in the range of 1000 to 1150 and in the range of 1250-1300 are considered as well - but with penalty weights. The user defines a start point (Start Pt - the points must be specified as labels, s. 5.3.1), the number of end points (Nr. of Pts.) and all end points (by specifying the End Pt. for each Pt. Nr.). Subsequently, after pressing Execute, a vessel tree is constructed. Only lines along the vessels are constructed (no vessel radii or vessel shapes). In a future version, the radii will be extracted and it will be possible to help the algorithm converge (in case of problems) by specifying intermediate points. Vessel segmentation can take a while to complete. The segmented vessel tree is shown overlayed on the Source and Target image in the colour of the currently active tissue.

Save... allows to save the currently displayed vessel structure in a humanly readable file format that can later be imported into SEMCAD X and used, e.g., as a vessel in thermal simulations. Douglas-Peucker simplification is employed to reduce the number of points on the vessel path [14, 6].

Figure 65: The Vessel tab.

5.5.15 Picker

The Picker tab (s. Figure 66) can be used to copy and erase regions. Copying can be used to transfer segmented regions from one slice to another. All the functions are based on the current region-selection. The contours of the region-selection are displayed in the colour of the currently active tissue.
Clicking a point will define the region in which the point is contained as the region selection. Clicking on another point will change the region-selection. If the Shift -key is pressed, clicking on a point outside the current region-selection will add the region of which the point is part to the region selection without removing the previously selected regions from the selection. If the Shift -key is pressed and a point inside the current selection is pressed, the region of which the point is part will be subtracted from the current region-selection. Regions are based on the tissue distribution or the Target image depending on whether Target or Tissue is selected.

The following operations can be performed:

- **Copy**: Copies an image on the clip-board. If Target is selected, the target image in the region-selection is copied onto the clip-board. If Tissue is selected the tissue distribution in the region-selection is copied. Subsequently the Paste button becomes available.

- **Cut**: If Cut is pressed the tissues (resp. target image pixels) inside the region-selection are erased but a copy is placed on the clip-board. If Erase is active, the deleted regions will simply be left empty (black, resp. no tissue assignment). Fill however, will fill the resulting hole based on the neighboring regions (tissues resp. grey values).

- **Delete**: If Delete is pressed the tissues (resp. target image pixels) inside the region-selection are erased. If Erase is active, cut or deleted regions will simply be left empty (black, resp. no tissue assignment). Fill however, will fill the resulting hole based on the neighboring regions (tissues resp. grey values).

- **Paste**: Paste is only active it there is content on the clip-board. After pressing the button, this content is pasted into the current slice (Target image or tissue distribution).

![Figure 66: The Picker tab.](image)
5.6 Img Clipboard

The **Img Clipboard** can be used to temporarily store images. It can be used to save images from being overwritten, to copy them to another location/slice or to move images between the **Source** and **Target** image as well as the tissue distribution. Images can be exported into files and read from files, which makes it possible to exchange images between iSeg projects (e.g., when multiple blocks of image data are segmented, and continuity between the blocks has to be maintained).

**Copy Target**, **Copy Source** and **Copy Tissue** copy the corresponding image from the current slice on the stack. The user can specify a name for the image on the stack and the slice number will automatically be appended to simplify keeping track of the image origin. **Copy Tissue** does copy an image of the tissue distribution of the currently selected tissue to the stack. **Paste Target** and **Paste Source** paste the selected stack image into the current slice of the **Target** or **Source** image. **Paste Tissue** adds all regions having a nonzero value on the selected stack image to the currently selected tissue on the active slice.

**Delete**: This function deletes the selected image from the stack.

**Save Item**: The selected stack image is saved in a .stk file including information about the image dimension. This can be used to export images from one project and import them into another project (e.g., when multiple blocks of image data are segmented, and continuity between the blocks has to be maintained).

**Load Item**: This permits loading a .stk file saved using the **Save Item** function (s. above).

![Image Clipboard](image.png)

**Figure 67**: The Image Clipboard.
5.7 Shortcuts

In addition to the commonly available shortcuts (such as the Tab button to shift the focus...), a series of iSeg specific shortcuts exist:

- **PgUp** and **PgDn** can be used to move up and down through the image slices. The same functionality can be accessed using the Ctrl+Left and Ctrl+Right keys. This can be valuable in a control element that assigns an alternative function to **PgUp** and **PgDn**.

- **Ctrl+Q** quits the application (prompting the user to save the project).

- **Ctrl+Up** and **Ctrl+Down** can be used to zoom in and out (affecting the display in all 2D viewers including the cross-section viewers (s. 5.3.1)).

- **Ctrl+ ‘+’ and Ctrl+ ‘-’** are shortcuts from pressing the + and the - buttons.

- **Ctrl+Z** and **Ctrl+Y** activate the Undo and the Redo functionality. **Ctrl+Esc** is another shortcut for the Undo function.

6 Acknowledgement

We would like to thank the following groups for their help in developing iSeg:

- The ‘Hyperthermia Unit’ at the Daniel den Hoed Cancer Center, Erasmus MC, Rotterdam, NL for extensive and early testing, clinical validation, valuable discussions and fruitful suggestions.

- The segmentation group at the IT’IS Foundation for Information Technologies in Society, Zurich, CH for extensive and early testing, valuable discussions and fruitful suggestions.

- Hansueli Gerber for extensive improvement suggestions concerning the GUI.

- Various groups worldwide for using and testing iSeg before the release.
7 License Notes

7.1 VTK

VTK 5.2.1 is used for the 3D rendering. The following licensing conditions apply:

Copyright (c) 1993-2008 Ken Martin, Will Schroeder, Bill Lorensen All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. * Neither name of Ken Martin, Will Schroeder, or Bill Lorensen nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

7.2 GDCM

GDCM 2.0.10 is used to load RTstruct dicoms. The following licensing conditions apply:

Program: GDCM (Grassroots DICOM). A DICOM library Module: URL
Copyright (c) 2006-2008 Mathieu Malaterre Copyright (c) 1993-2005 CREATIS (CREATIS = Centre de Recherche et d’Applications en Traitement de l’Image) All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither name of Mathieu Malaterre, or CREATIS, nor the names of any contributors (CNRS, INSERM, UCB, Université Lyon 1), may be used to endorse or promote products derived from this software without specific prior written permission.

* Modified source versions must be plainly marked as such, and must not be misrepresented as being the original software.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

7.3 Qt

Qt 4.5.1 has been used to creat the GUI. Only dynamic linking with the unmodified Qt library is performed. This is permissible under the LGPL v2.1 licensing scheme offered by Qt. The following licensing conditions apply and must be included:

GNU LESSER GENERAL PUBLIC LICENSE Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc. 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

PREAMBLE

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages—typically libraries—of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author’s reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict
the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the 'Lesser' General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a 'work based on the library' and a 'work that uses the library'. The former contains code derived from the library, whereas the latter must be combined with the library.
in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called 'this License'). Each licensee is addressed as 'you'.

A 'library' means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The 'Library', below, refers to any such software library or work which has been distributed under these terms. A 'work based on the Library' means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term 'modification'.)

'Source code' for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.
4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

    If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a 'work that uses the Library'. Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

    However, linking a 'work that uses the Library' with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a 'work that uses the library'. The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

    When a 'work that uses the Library' uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

    If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

    Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a 'work that uses the Library' with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer’s own use and reverse engineering for debugging such modifications.

71
You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable 'work that uses the Library', as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user’s computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the 'work that uses the Library' must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library
side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply, and
the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents
or other property right claims or to contest validity of any such claims; this
section has the sole purpose of protecting the integrity of the free software
distribution system which is implemented by public license practices. Many
people have made generous contributions to the wide range of software dis-
tributed through that system in reliance on consistent application of that sys-
tem; it is up to the author/donor to decide if he or she is willing to distribute
software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a
consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain
countries either by patents or by copyrighted interfaces, the original copy-
right holder who places the Library under this License may add an explicit
geographical distribution limitation excluding those countries, so that distri-
bution is permitted only in or among countries not thus excluded. In such
case, this License incorporates the limitation as if written in the body of this
License.

13. The Free Software Foundation may publish revised and/or new ver-
sions of the Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may differ in
detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and 'any later
version', you have the option of following the terms and conditions either
of that version or of any later version published by the Free Software Foun-
dation. If the Library does not specify a license version number, you may
choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free pro-
grams whose distribution conditions are incompatible with these, write to the
author to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals of pre-
serving the free status of all derivatives of our free software and of promoting
the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE,
THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT
PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE
STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER
PARTIES PROVIDE THE LIBRARY 'AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the 'copyright' line and a pointer to where the full notice is found.

one line to give the library's name and a brief idea of what it does.©

Copyright (C) ©year ©name of author©

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.
8 Usage Recommendation

This section gives usage recommendation based on the experience of multiple iSeg users.

References


