The IST Programme
Project No. 10378

SimBio

SimBio - A Generic Environment for Bio-numerical Simulation

http://www.simbio.de
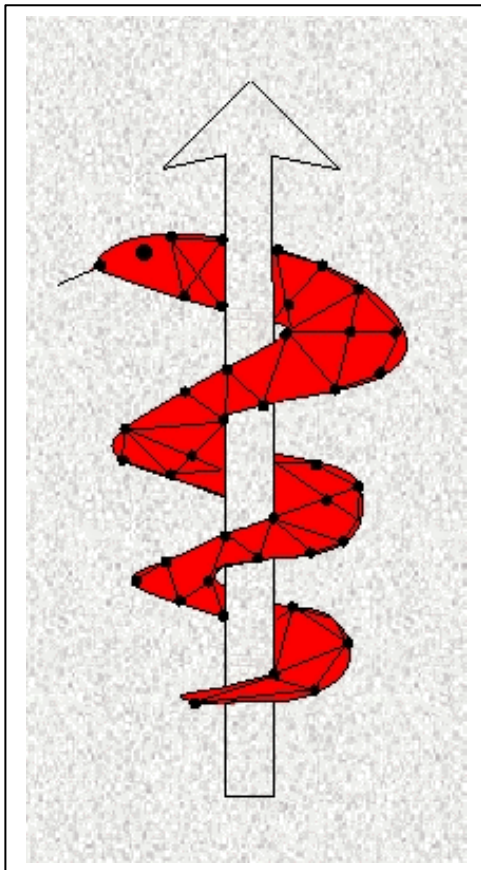
Deliverable D1.1b: Release Notes on the preliminary image-processing tool release

Status:	Final
Version:	2.0
Security:	Restricted
Responsible:	USFD
Authoring Partners:	USFD, MPI

Release History

Version	Date
0.1	01.09.00
1.0	29.09.00
2.0	23.10.01 (Deliverable)

SimBio
Subtask 1.1 - Image Processing

The purpose of subtask 1.1 is to take a patient specific image and segment this image into appropriate anatomical segments. In subtask 1.2 a Finite Element mesh is developed on the basis of these segments. Although mesh generation is not the formal concern of subtask 1.1 the generic strategy adopted in 1.1 lends itself to the creation of patient specific meshes and the software developments represented in this release have been pursued in collaboration with subtask 1.2.
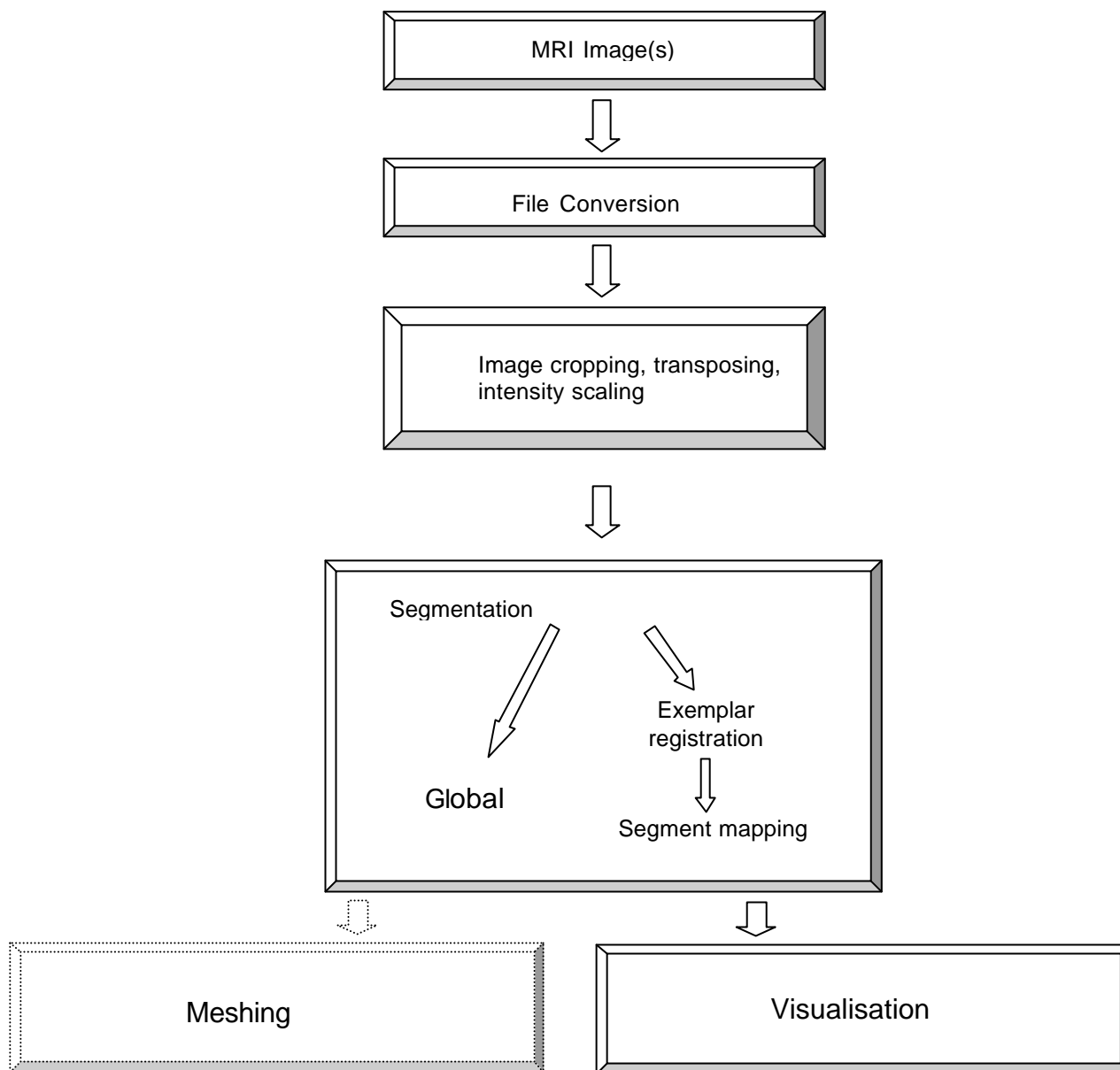
Whatever the short term requirement of SimBio, the aim must be to produce a generic approach to the problem of patient specific mesh generation. The strategy adopted in 1.1 is to segment an image through a process of image registration. Successful segmentation requires the inclusion of domain specific knowledge in the segmentation process.   This is often a mixture of low-level knowledge, that segments are characterised by constant intensity (within limits) or contained within boundaries defined by intensity gradients, and problem specific knowledge about how the low level operations are strung together, but generation of an optimum segmentation can be difficult except in a few special cases.   Within this release is a module (*vsegment3d*) which is optimised for intensity segmentation of brain images and this works well, but does not generalise to arbitrary images, especially the knee images used within SimBio.

With an image registration approach a previously hand segmented image is used as a reference image which can be registered to patient images. The mapping resulting from this registration is then used to map the image segments to the patient image. The domain knowledge is contained in the reference image and the segmentation and in this respect the segmentation is generic since this knowledge is distinct from the registration algorithm.   The segmentation of the reference image may be time consuming but only needs to be done once. Two registration algorithms are included in this release, *vreg3d* and *vreglocal3d.* The former is optimised for rigid registration (with optimal scaling) of data from the same subject, but includes the ability to register images from different modalities. The latter is a full non-linear registration algorithm for registering images from the same modality but where one of the images is a reference image. This is accompanied by a second module, *vtransform,* which applies the mapping function produced by *vreglocal3d* to either further images (such as segments represented as binary images) or a mesh. The intention behind the use of this approach is that the reference image should be registered to the patient image.

The aim of subtask 1.1 is to produce this segmentation, with the responsibility for meshing of this segment being that of subtask 1.2.   However, an FE mesh can be generated for the reference segments, most practically using a mixture of automatic segmentation algorithms with hand editing. Again, this may be time consuming, but only needs to be done once. Following this the registration mapping can be used to map the mesh to the patient image. This does not guarantee that the mapped mesh will be ideal but smoothness constraints within the registration algorithms will give some protection against collapse of the mesh.   Evaluation of this approach will be undertaken over the remaining nine months of subtask 1.1.

This release also contains some utility software including two format conversion routines (*brutov* and *dcmtov*) to convert image data to Vista format, and some image manipulation and intensity scaling modules (*vtranspose3d, vcrop3d, vintens3d* and *valign3d*). The last of these four is specifically designed for brain data.

All modules communicate through image and other data in Vista format. The flow of data follows the following data stream

```
                        ┌─────────────────────────┐
                        │      MRI Image(s)        │
                        └─────────────────────────┘
                                    ⇓
                        ┌─────────────────────────┐
                        │     File Conversion      │
                        └─────────────────────────┘
                                    ⇓
                        ┌─────────────────────────┐
                        │ Image cropping, transposing,
                        │ intensity scaling        │
                        └─────────────────────────┘
                                    ⇓
        ┌───────────────────────────────────────────────┐
        │  Segmentation                                  │
        │            ↘          ↘                         │
        │                   Exemplar                     │
        │                  registration                  │
        │                      ⇓                          │
        │     Global      Segment mapping                │
        └───────────────────────────────────────────────┘
              ⇓                          ⇓
   ┌────────────────────┐    ┌────────────────────────┐
   │     Meshing        │    │    Visualisation       │
   └────────────────────┘    └────────────────────────┘
```

Manpages for ST1.1 Modules

1.1 brutov - converts Bruker to Vista format

SYNOPSIS
brutov [-option ...] [infile] [outfile]

DESCRIPTION
brutov converts a group of Bruker data files into a Vista data file.

COMMAND LINE OPTIONS
brutov accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies a directory or a tar file that contains the source data.
-out *outfile*	Specifies the output file, which will be a Vista data file.
-white *number*	Specifies the percentage of voxels that will be mapped to white (default 0.5%).
-black *number*	Specifies the percentage of voxels that will be mapped to black (default 10%).
-ds 1 2 ...	Select datasets 1, 2, etc. from an experiment.

The keyword ``to'' can be used to specify a range of indices, as in ``-ds 1 to 2''. Axial and coronal slices are flipped from the radiologic convention into the natural convention, i.e. the left image side corresponds to the left body side. An attribute "convention: natural" is appended to the list to document this orientation. Note that sagittal slices are not flipped, i.e. they are found "nose left".

EXAMPLES
The following command line
    brutov -in WA1T961101.GB1 -out vista-file.v -rep 4
expects a directory tree with root WA1T961101.GB1 containing Bruker parameter and data files.
To select only a subset of the datasets in an experiment, use
    brutov -in WA1T961101.GB1 -out vista-file.v -ds 4 5 7 to 14
This will convert datasets 4, 5, and 7 to 14 only.

NOTES
The attribute *convention* is always set to *natural*, and the attribute *component_interp* set to *intensity*. Conversion of diffusion tensor images is unimplemented, so the attribute *component_repn* is always set to *scalar*.

AUTHOR
Frithjof Kruggel.

1.2 dcmtov - convert DICOM to Vista format

SYNOPSIS
dcmtov [infile] [-option ...] [outfile]

DESCRIPTION
dcmtov converts a series of DICOM files into a Vista file. Input filenames are expected to be in printf format (see below), with a basename and indices ranging from first to last. On output, a Vista volume dataset is generated. Individual series within a session are represented as separate Vista image objects within the Vista file. Each image object has its own set of information regarding the patient name, image parameters etc.

COMMAND LINE OPTIONS
dcmtov accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies the basename for a DICOM file series. This name should be given as printf format string, so that a specific filename may be generated from this format string and the values of first and last below.
-out *outfile*	Specifies the output file, which will be a Vista data file.
-first *index*	Specifies the index of the first input file, which is a DICOM file.
-last *index*	Specifies the index of the last input file, which is a DICOM file.
-type *ubyte*/*ushort*	Specifies the output image representation. (default ubyte).
-white *number*	Specifies the percentage of voxels that will be mapped to white (default 0.5%).
-black *number*	Specifies the percentage of voxels that will be mapped to black (default 10%).

The message "dcmtov: input file XXX.001 not in DICOM format" notifies that either the filename was misspelled or byte swapping is necessary to convert the file. Try the option -swap. The output file can be specified on the command line or allowed to default to the standard output stream.

EXAMPLE
The following command line
    dcmtov -in %03d.ima -out vista-file.v -first 1 -last 203
will convert the sequence of DICOM files "001.ima" to "203.ima" into a Vista file.

AUTHOR
Frithjof Kruggel

1.3 vtranspose3d - transpose a 3D dataset

SYNOPSIS
vtranspose3d [-option ...] [infile] [outfile]

DESCRIPTION
vtranspose3d transposes a Vista image file according to the specified co-ordinate changes and returns a re-dimensioned Vista volume dataset.

COMMAND LINE OPTIONS
vtranspose3d accepts the following options:

-help	Prints a message describing options.

-in *infile*	Specifies a Vista data file containing the input images.
-out *outfile*	Specifies where to write the output image.
-xyz *abc*	Specifies the co-ordinate transpositions. Available transpositions: xyz (default), xzy, zyx, zxy, yxz, yzx

Input and output files can be specified on the command line or allowed to default to the standard input and output streams.

EXAMPLE
To convert a sagittal 3D Vista dataset ("nose left") into a standard axial dataset ("nose up") use:

    vtranspose3d -in xx.v -out xx1.v -xyz zxy

AUTHOR
Christian Uhl, Frithjof Kruggel.

1.4 vcrop3d - crop a 3D dataset

SYNOPSIS
vcrop3d [-option ...] [infile] [outfile]

DESCRIPTION
vcrop3d is used to crop a 3D dataset, given a left upper corner and an extent. Both corner point and extent are expected to be given in voxels. The corner point may have negative components. Regions of the destination volume which were not present in the input are filled with the value 0.

COMMAND LINE OPTIONS
vcrop3d accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies the source image dataset in Vista format.
-out *outfile*	Specifies the output file, which will be a Vista image.
-corner *x y z*	Specifies the left upper corner (default: 0 0 0).
-extent *x y z*	Specifies the extent of the destination volume (default: 0 0 0).

Input and output files can be specified on the command line or allowed to default to the standard input and output streams.

AUTHOR
Frithjof Kruggel.

1.5 valign3d - align a 3D dataset with the stereotactical co-ordinate system

SYNOPSIS
valign3d [-option ...] [infile] [outfile]

DESCRIPTION
valign3d is used to align a 3D dataset with the stereotactical co-ordinate system. Relevant information, i.e., the position of the anterior (CA) and posterior commissure (CP) and the rotation angle must be specified manually.

COMMAND LINE OPTIONS

valign3d accepts the following options:

-help	Prints a message describing options.
-in infile	Specifies the source image dataset in Vista format.
-out outfile	Specifies the output file, which will be a Vista data file.
-ca *x y z*	Specifies the position of the commissura anterior.
-cp *x y z*	Specifies the position of the commissura posterior.
-angle *x y z*	Specifies the rotation around the x, y and z axis.
-transpose *true	false*	
	Transpose a set of sagittal slices into axial slices before adapting the co-ordinate system.

Input and output files can be specified on the command line or allowed to default to the standard input and output streams. CA and CP position are expected in mm, measured from the left upper corner of the image volume. The convention for brain datasets denotes the x axis as parallel to the ear-to-ear direction (from left to right), the y axis to the nose-to-back direction, and the z axis to the body axis (from top to bottom). A positive y rotation (i.e. -angle 0 3 0) nods the head 3 degrees right, and a positive z rotation (i.e. -angle 0 0 3) turns the head along the body axis 3 degrees right. If CA and CP are specified, the x rotation is ignored and determined from these co-ordinates.

AUTHOR

Frithjof Kruggel.

1.6 vreg3d - register two 3D datasets

SYNOPSIS

vreg3d [-option ...] [infile] [outfile]

DESCRIPTION

vreg3d is used to register a 3D dataset with a reference dataset. Three different registration plans are imple mented: one for intra-modal registration problems (such as T1-T1 registration of head datasets), one for cross-modal datasets (for registering a set of T2-weighted slices to a T1-weighted head dataset), and a "manual" plan, where the registration method, an initial rotation and translation should be specified.

COMMAND LINE OPTIONS

vreg3d accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies the source image dataset in Vista format.
-out *outfile*	Specifies the output file, which will be a Vista data file.
-ref *file*	Specifies the reference image dataset in Vista format.
-plan *intra	cross	manual*	
	Specifies the registration plan (default: intra).
-resort *true	false*	Specifies whether the stacking order of the source dataset should be reversed (default: false).
-scaling *true	false*	
	Specifies whether re-scaling is allowed (default: false).
-angle *x y z*	Specifies the initial rotation angle (default: 0 0 0).
-trans *x y z*	Specifies the initial translation (default: 0 0 0).
-func *nmi	cc*	Specifies the type of cost function: normalised mutual information (nmi) or correlation coefficient (cc) (default: nmi).

Input and output files can be specified on the command line or allowed to default to the standard input and output streams. For plans *intra* and *cross*, any values given for *-rt*, *-tr*, or *-method* are ignored. Note that the translation is expected to be given in mm. Typical registration problems need 10-30 min computation time on a PC equipped with a 500 MHz Intel Pentium III processor.

AUTHOR
Frithjof Kruggel.

1.7 vreglocal3d - Non-linear image registration

SYNOPSIS
vreglocal3d [-option ...] [infile] [outfile]

DESCRIPTION
vreglocal3d performs a non-linear registration of two 3D data sets. It returns the registered data set and also a mapping function which when supplied to **vtransform** will apply the mapping to an image or mesh.

COMMAND LINE OPTIONS
vreglocal3d accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies the source image dataset in Vista format.
-out *outfile*	Specifies the output file, which will be a Vista image.
-ref *reffile*	Specifies the target image for registration.
-roi *roifile*	Specifies the region of interest image for registration. Default: +/- 10% of the input image dimensions on each side.
-map *mapfile*	Specifies whether to write out the mapping function.
-lamda *number*	Specifies the value of the smoothness constraint. Default: 1.
-affine *logical*	Specifies whether to carry out an affine transform first. Default: FALSE.
-xn *number*	Specifies the number of mesh nodes along the x-axis. Default: 10.
-yn *number*	Specifies the number of mesh nodes along the y-axis. Default: 10.
-zn *number*	Specifies the number of mesh nodes along the z-axis. Default: 10.

Input and output files can be specified on the command line or allowed to default to the standard input and output streams.

AUTHOR
Barber & Wood.

1.8 vtransform - mapping of segments

SYNOPSIS
vtransform [infile] [outfile] [mapfile]

DESCRIPTION
vtransform transforms the input data set using the mapping function provided by **vreglocal3d**. If input is a mesh, the vertices of the mesh will be transformed, nothing will be

done to other components such as vertex normals. If the input is an image, an image will be returned.

COMMAND LINE OPTIONS
vtransform accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies the source image dataset in Vista format.
-out *outfile*	Specifies the output file, which will be a Vista image.
-map *mapfile*	Specifies whether to write out the mapping function.

Input and output files can be specified on the command line or allowed to default to the standard input and output streams.

AUTHOR
Barber & Wood.

1.9 vintens3d - correct intensities a 3D dataset according to a reference

SYNOPSIS
vintens3d [-option ...] [infile] [outfile]

DESCRIPTION
vintens3d corrects image intensities a 3D dataset, given a reference image. First, intensities in both images are scaled to fit in a 256x256 joint histogram. A regression line is fitted to the joint histogram, and intensities in the input dataset are transformed according to the regression parameters. A threshold may be specified to leave out the background during the computation of the regression line.

COMMAND LINE OPTIONS
vintens3d accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies the source image dataset in Vista format.
-out *outfile*	Specifies the output file, which will be a Vista image.
-ref *file*	Specifies the reference image.

Input and output files can be specified on the command line or allowed to default to the standard input and output streams.

AUTHOR
Frithjof Kruggel.

1.10 vsegment3d - segment a 3D dataset

SYNOPSIS
vsegment3d [-option ...] [infile] [outfile]

DESCRIPTION

vsegment3d segments a 3D dataset based on intensity criteria into a set of classes. Optionally, it tries to correct for the intensity variations due to inhomogeneities of the B1 field of the MR scanner. For input data sets aligned with the stereotactical co-ordinate system, the argument -opt true specifies to correct within a minimal subvolume only, for a threefold reduction in computation time.

COMMAND LINE OPTIONS

vsegment3d accepts the following options:

-help	Prints a message describing options.
-in *infile*	Specifies the source image dataset in Vista format.
-out *outfile*	Specifies the output file, which will be a Vista image.
-nc *number*	Specifies the number of tissue classes. Default: 3.
-cls *clsfile*	Specifies where to store the classified image.
-win *number*	Specifies the processing window width. Default: 8
-lim *number*	Specifies the intensity limit of the background class. Default: 35
-lambda1 *number*	Specifies the value of the first regularisation constant. Default: 200000.
-lambda2 *number*	Specifies the value of the second regularisation constant. Default: 2000000.

Input and output files can be specified on the command line or allowed to default to the standard input and output streams.

AUTHOR

Frithjof Kruggel.

Pham DL, Prince JL (1999) An adaptive fuzzy C-means algorithm for image segmentation in the presence of intensity inhomogeneities. Pattern Recognition Letters 20, 57-68.