SCIENTIFIC COMPUTING AND IMAGING INSTITUTE

ShapeWorks Documentation

ShapeToolKit : Image Pre-processing Tools Documentation

Shireen Elhabian, Praful Agrawal, Riddhish Bhalodia,
Joshua Cates, Manasi Datar, Ross Whitaker

June 27, 2018

Abstract

This document covers the 3D image processing tools which are required for preparing the data prior to running shapeworks, it consists of breif tool description and it's usage.

Contents

1 License

ShapeWorks and its graphical user interface ShapeWorksStudio are available for free and are open source under the MIT License. For more information, please refer to `https://www.sci.utah.edu/software/shapeworks.html`.

2 Acknowledgement

If you use ShapeWorks and/or ShapeWorksStudio in work that leads to published research, we humbly ask that you add the following to the 'Acknowledgments' section of your paper: *"This project was supported by the National Institute of General Medical Sciences of the National Institutes of Health under grant number P41 GM103545-18."*

2.1 Isotropic Resampling

This supports both binary as well as raw 3D volumes and resamples it to the given specifications with isotropic voxel spacing.
The command line tool can be used as follows:

./ResampleVolumesToBeIsotropic.sh –inFilename $1 –outFilename $2 –isoSpacing $4 –isBinaryImage $5 –isCenterImageOn $6 –sizeX $7 –sizeY $8 –sizeZ $9

It uses the following input arguments:

1. inFilename : The filename of the input image to be resampled.
2. outFilename : The filename of the output resampled image.
3. isBinaryImage : A flag to treat the input image as a binary image (specialized resampling pipeline) [default disabled].
4. isoSpacing : The isotropic spacing in all dimensions.
5. sizeX : Image size in x-direction (optional, if set to 0, the size is autmatically estimated from the input image).
6. sizeY : Image size in y-direction (optional, if set to 0, the size is autmatically estimated from the input image).
7. sizeZ : Image size in z-direction (optional, if set to 0, the size is autmatically estimated from the input image).
8. isCenterImageOn : A flag to center the image, i.e. change the origin in the image header to the physcial coordinates of the first voxel (lower left corner) [default disabled].

We also provide an associated script which can be used for batch processing a set of volumes, it's usage is given as follows:

./ResampleVolumes.sh –data_dir $1 –data_prefix $2 –out_dir $3
–iso_spacing $4 –is_binary_image $5 –is_center_on $6
–sizeX $7 –sizeY $8 –sizeZ $9
Parameter description :

- `--data_dir`: The directory where all the inputs lie.
- `--data_prefix`: The prefix of the files which will be the one to be selected for resampling.
- `--out_dir`: Output directory where resampled volumes will be saved.
- `--is_binary_image`: A flag to treat the input image as a binary image (specialized resampling pipeline) [default disabled].
- `--iso_spacing`: The isotropic spacing in all dimensions.
- `--sizeX`: Image size in x-direction (optional, if set to 0, the size is automatically estimated from the input image).
- `--sizeY`: Image size in y-direction (optional, if set to 0, the size is automatically estimated from the input image).
- `--sizeZ`: Image size in z-direction (optional, if set to 0, the size is automatically estimated from the input image).
- `--is_center_on`: A flag to center the image, i.e. change the origin in the image header to the physical coordinates of the first voxel (lower left corner) [default disabled].

2.2 Constant Padding

./PadVolumeWithConstant.sh –inFilename $1 –outFilename $2
–paddingSize $3 –paddingValue $4
Pad the volume in x y and z directions with a constant value.
The tool can be used as follows:

It uses the following input arguments:

1. inFilename : Image file to be padded.
2. outFilename : Padded Image file.
3. paddingSize : Number of voxels to be padded in each direction.
4. paddingValue : Value to be used to fill padded voxels.

We also provide an associated script for batch processing whose usage is given as follows

./PadVolumes.sh –data_dir $1 –img_prefix $2 –seg_prefix $3
–out_dir $4 –seg_suffix $5 –process_raw $6
–padding_size $7 –padding_value $8
Parameter description :

- `--data_dir`: The directory where all the inputs lie.
- `--img_prefix`: The prefix of the raw image files which will be the one to be selected for padding.
- `--seg_prefix`: The prefix of the binary segmentation image files which will be the one to be selected for padding.
- `--out_dir`: Output directory where padded volumes will be saved.
- `--seg_suffix`: The suffix for binary volumes, this is needed to mark all the pre-processing steps involved before the current step
- `--padding_value`: The constant padding value [Default 0]
- `--padding_size`: The padding size
- `--process_raw`: A flag to which processes the raw volumes along with binary images in the same way.

2.3 Find Largetst Bounding Box and Cropping

These two tools are usually used in conjunction. A command line tool which computes the largest bounding box for the shape population needed for cropping all the images. And a volume cropping tool which takes in teh bounding box parameters and crops any given volume to specified dimensions.
The tools are used as follows

./FindLargestBoundingBox –inFilename $1 –outPrefix $2 –paddingSize $3 It uses the following input arguments

1. inFilename : A text file with the file names for which the largest size has to be computed.
2. outPrefix : output prefix to be used to save the parameters for the estimated bounding box
3. paddingSize : number of extra voxels in each direction to pad the largest bounding box, checks agains min image size is performed to make sure that this padding won't get out of bounds for the smallest image in the file names provides

./CropImages –inFilename $1 –outFilename $2
–MRIinFilename $3 –MRIoutFilename $4 –bbX $5
–bbY $6 –bbZ $7 –startingIndexX $8
–startingIndexY $9 –startingIndexZ $10 The parameters are as follows

1. inFilename : Input image file name
2. outFilename : Output image file name
3. MRIinFilename : MRI image file name
4. MRIoutFilename : MRI output image file name
5. bbX : bounding box value in X direction
6. bbY : bounding box value in Y direction
7. bbZ : bounding box value in Z direction
8. startingIndexX : starting index in X direction
9. startingIndexY : starting index in Y direction
10. startingIndexZ : starting index in Z direction

We also provide associated scripts for batch processing, they are given as follows

./FindLargestBoundingBox.sh –data_dir $1 –seg_prefix $2
–seg_suffix $3
Parameter description :

- `--data_dir`: The directory where all the inputs lie.
- `--seg_prefix`: The prefix of the binary segmentation image files which will be the one to be selected for largest bounding box computation.
- `--seg_suffix`: The suffix for binary volumes, this is needed to mark all the pre-processing steps involved before the current step.

./CropVolumes.sh –data_dir $1 –img_prefix $2 –seg_prefix $3
–out_dir $4 –seg_suffix $5 –process_raw $6
Parameter description :

- `--data_dir`: The directory where all the inputs lie.
- `--img_prefix`: The prefix of the raw image files which will be the one to be selected for cropping.
- `--seg_prefix`: The prefix of the binary segmentation image files which will be the one to be selected for cropping.
- `--seg_suffix`: The suffix for binary volumes, this is needed to mark all the pre-processing steps involved before the current step.
- `--process_raw`: A flag to which processes the raw volumes along with binary images in the same way.
- `--out_dir`: Output directory where aligned volumes will be saved.

2.4 ThresholdImages

This tool performs segmentation of input volume based on the specified upper and lower bounds. The command line usage is given as follows :

./ThresholdImages –inFilename $1 –outFilename $2
–lowerThresholdLevel $3 –upperThreholdLevel $4
–insideValue $5 –outsideValue $6

The parameters are as follows

1. inFilename = Input image file path.
2. outFilename = The filename of the output threshold image.
3. lowerThresholdLevel = The lower threshold level (optional, default = FLT_MIN)
4. upperThresholdLevel = The upper threshold level (optional, default = FLT_MAX)
5. insideValue = The inside pixel value after threshold
6. outsideValue = The outside pixel value after threshold

2.5 TopologyPreservingSmoothing

This tool smoothes the distance transforms while preserving the shape's topology, the usage is given as follows:

./TopologyPreservingSmoothing $parameterFile

The parameter file is the xml file with the folwwing tags

1. inputs : The filenames of the input distance transforms to be smoothed.
2. dtFiles : The filenames of the output smoothed distance transforms.
3. outputs : The filenames of the output smoothed isosurface images.

4. verbose : Show each intermediate step [default 0].
5. isoValue : Isovalue to be used to define the surface in the input distance transform [default 0.0].
6. smoothing_iterations : Number of iterations to perform smoothing [default 10].
7. alpha : Smoothing parameter in I' = (max-min). $\frac{1}{1+exp(-\frac{1-\beta}{\alpha})}$ + min [default 10.5].
8. beta : Smoothing parameter in I' = (max-min). $\frac{1}{1+exp(-\frac{1-\beta}{\alpha})}$ + min [default 10.0].
9. propagationScale : The PropagationScaling parameter can be used to switch from propagation outwards (POSITIVE) versus propagating inwards (NEGATIVE).