imtools Documentation

M. Jirik

Jan 25, 2019

Contents

1 imtools package **3**
- 1.1 Subpackages . . . 3
- 1.2 Submodules . . . 3
- 1.3 imtools.misc module . . . 3
- 1.4 imtools.qmisc module . . . 3
- 1.5 imtools.sample_data module . . . 3
- 1.6 imtools.segmentation module . . . 4
- 1.7 imtools.surface_measurement module . . . 4
- 1.8 imtools.tools module . . . 4
- 1.9 imtools.uiThreshold module . . . 4
- 1.10 imtools.image_manipulation module . . . 6
- 1.11 Module contents . . . 6

2 Indices and tables **7**

Python Module Index **9**

Contents:

CHAPTER 1

imtools package

Test text

1.1 Subpackages

1.2 Submodules

1.3 imtools.misc module

1.4 imtools.qmisc module

1.5 imtools.sample_data module

imtools.sample_data.**check_python_architecture**(*pythondir*, *target_arch_str*)
functions check architecture of target python

imtools.sample_data.**checksum**(*path*, *hashfunc='md5'*)
Return checksum given by path. Wildcards can be used in check sum. Function is strongly dependent on checksumdir package by 'cakepietoast'.

Parameters

- **path** –
- **hashfunc** –

Returns

imtools.sample_data.**donut**()
Generate donut like shape with stick inside

Returns datap with keys data3d, segmentation and voxelsize_mm

imtools.sample_data.**download_and_run**(*url*, *local_file_name*)

imtools.sample_data.**downzip**(*url*, *destination='./sample_data/'*)
Download, unzip and delete.

imtools.sample_data.**file_copy_and_replace_lines**(*in_path*, *out_path*)

imtools.sample_data.**generate**(*size=100*, *liver_intensity=100*, *noise_intensity=20*, *portal_vein_intensity=130*, *spleen_intensity=90*)

imtools.sample_data.**get**(*data_label=None*, *destination_dir='.'*)
Download sample data by data label. Labels can be listed by sample_data.data_urls.keys() :param data_label: label of data. If it is set to None, all data are downloaded :param destination_dir: output dir for data :return:

imtools.sample_data.**get_conda_path**()
Return anaconda or miniconda directory :return: anaconda directory

imtools.sample_data.**get_sample_data**(*data_label=None*, *destination_dir='.'*)
Same as get() due to back compatibility :param data_label: :param destination_dir: :return:

imtools.sample_data.**main**()

imtools.sample_data.**make_icon**()

imtools.sample_data.**remove**(*local_file_name*)

imtools.sample_data.**submodule_update**()

1.6 imtools.segmentation module

1.7 imtools.surface_measurement module

Measurement of object surface.

data3d: 3D numpy array segmentation: 3D numpyarray

imtools.surface_measurement.**bufford_needle_sond**(*data3d*, *voxelsize_mm*, *raster_mm*, *axis*, *aoi*)

imtools.surface_measurement.**find_edge**(*segmentation*, *axis*)

imtools.surface_measurement.**main**()

imtools.surface_measurement.**surface_density**(*segmentation*, *voxelsize_mm*, *aoi=None*, *sond_raster_mm=None*)

Segmentation is ndarray with 0 and 1

Voxelsize_mm is array of three numbers specifiing size of voxel for each axis

Aoi is specify area of interest. It is ndarray with 0 and 1

Sond_raster_mm unimplemented. It is parametr of sonds design

1.8 imtools.tools module

1.9 imtools.uiThreshold module

Purpose: (CZE-ZCU-FAV-KKY) Liver medical project

Author: Pavel Volkovinsky, Miroslav Jirik Email: volkovinsky.pavel@gmail.com

Created: 2012/11/08 Copyright: (c) Pavel Volkovinsky

imtools.uiThreshold.**main**()

imtools.uiThreshold.**make_image_processing**(*data*, *voxelsize_mm*, *seeds=None*, *sigma_mm=1*, *min_threshold=None*, *max_threshold=None*, *closeNum=0*, *openNum=0*, *min_threshold_auto_method=''*, *fill_holes=True*, *get_priority_objects=True*, *nObj=1*, *debug=True*)

imtools.uiThreshold.**prepare_threshold_from_seeds**(*data*, *seeds*, *min_threshold_auto_method*)

class imtools.uiThreshold.**uiThreshold**(*data*, *voxel*, *threshold=None*, *interactivity=True*, *number=100.0*, *inputSigma=-1*, *nObj=10*, *biggestObjects=True*, *useSeedsOfCompactObjects=True*, *binaryClosingIterations=2*, *binaryOpeningIterations=0*, *seeds=None*, *cmap=<Mock name='mock.cm.Greys_r' id='139675241087440'>*, *fillHoles=True*, *figure=None*, *threshold_auto_method=''*, *threshold_upper=None*, *debug=True*)

UI pro prahovani 3D dat.

buttonContinue(*event*)

buttonMaxNext(*event*)

buttonMaxNext5(*event*)

buttonMaxPrev(*event*)

buttonMaxPrev5(*event*)

buttonMaxUpdate(*event*, *value*)

buttonMinNext(*event*)

buttonMinNext5(*event*)

buttonMinPrev(*event*)

buttonMinPrev5(*event*)

buttonMinUpdate(*event*, *value*)

buttonNextClosing(*event*)

buttonNextOpening(*event*)

buttonPrevClosing(*event*)

buttonPrevOpening(*event*)

buttonReset(*event*)

debugInfo()

drawVisualization()
Vykresleni dat.

getBiggestObjects()
Vraceni nejvetsich objektu (nebo objektu, ktere obsahuji prioritni seedy).

returnLastThreshold()

run()
Spusteni UI.

updateImage(*val*)
Hlavni update metoda. Cinny kod pro gaussovske filtrovani, prahovani, binarni uzavreni a otevreni a vraceni nejvetsich nebo oznacenych objektu.

1.10 imtools.image_manipulation module

1.11 Module contents

CHAPTER 2

Indices and tables

- genindex
- modindex
- search

Python Module Index

i

imtools, 6
imtools.misc, 3
imtools.sample_data, 3
imtools.surface_measurement, 4
imtools.uiThreshold, 4

Index

B

bufford_needle_sond() (in module imtools.surface_measurement), 4
buttonContinue() (imtools.uiThreshold.uiThreshold method), 5
buttonMaxNext() (imtools.uiThreshold.uiThreshold method), 5
buttonMaxNext5() (imtools.uiThreshold.uiThreshold method), 5
buttonMaxPrev() (imtools.uiThreshold.uiThreshold method), 5
buttonMaxPrev5() (imtools.uiThreshold.uiThreshold method), 5
buttonMaxUpdate() (imtools.uiThreshold.uiThreshold method), 5
buttonMinNext() (imtools.uiThreshold.uiThreshold method), 5
buttonMinNext5() (imtools.uiThreshold.uiThreshold method), 5
buttonMinPrev() (imtools.uiThreshold.uiThreshold method), 5
buttonMinPrev5() (imtools.uiThreshold.uiThreshold method), 5
buttonMinUpdate() (imtools.uiThreshold.uiThreshold method), 5
buttonNextClosing() (imtools.uiThreshold.uiThreshold method), 5
buttonNextOpening() (imtools.uiThreshold.uiThreshold method), 5
buttonPrevClosing() (imtools.uiThreshold.uiThreshold method), 5
buttonPrevOpening() (imtools.uiThreshold.uiThreshold method), 5
buttonReset() (imtools.uiThreshold.uiThreshold method), 5

C

check_python_architecture() (in module imtools.sample_data), 3
checksum() (in module imtools.sample_data), 3

D

debugInfo() (imtools.uiThreshold.uiThreshold method), 5
donut() (in module imtools.sample_data), 3
download_and_run() (in module imtools.sample_data), 4
downzip() (in module imtools.sample_data), 4
drawVisualization() (imtools.uiThreshold.uiThreshold method), 5

F

file_copy_and_replace_lines() (in module imtools.sample_data), 4
find_edge() (in module imtools.surface_measurement), 4

G

generate() (in module imtools.sample_data), 4
get() (in module imtools.sample_data), 4
get_conda_path() (in module imtools.sample_data), 4
get_sample_data() (in module imtools.sample_data), 4
getBiggestObjects() (imtools.uiThreshold.uiThreshold method), 5

I

imtools (module), 6
imtools.misc (module), 3
imtools.sample_data (module), 3
imtools.surface_measurement (module), 4
imtools.uiThreshold (module), 4

M

main() (in module imtools.sample_data), 4
main() (in module imtools.surface_measurement), 4
main() (in module imtools.uiThreshold), 5
make_icon() (in module imtools.sample_data), 4
make_image_processing() (in module imtools.uiThreshold), 5

P

prepare_threshold_from_seeds() (in module imtools.uiThreshold), 5

R

remove() (in module imtools.sample_data), 4
returnLastThreshold() (imtools.uiThreshold.uiThreshold method), 5
run() (imtools.uiThreshold.uiThreshold method), 6

S

submodule_update() (in module imtools.sample_data), 4
surface_density() (in module imtools.surface_measurement), 4

U

uiThreshold (class in imtools.uiThreshold), 5
updateImage() (imtools.uiThreshold.uiThreshold method), 6