ITK: The Insight Segmentation and Registration Toolkit

SciPy Conference
July 11th, 2018

Matthew McCormick[1], Francois Budin[1], Dženan Zukić[1], Deepak Chittajallu[1], Beatriz Paniagua[1], Jean-Christophe Fillion-Robin[1]

[1]*Kitware, Inc.*

Kitware

Outline

1. Background
   a. Introduction to ITK
   b. Python Wrapping History
2. New Developments
   a. Python Packages
   b. Module Packages Developed on GitHub
   c. Pythonic Interface
   d. NumPy Bridge
   e. itk-jupyter-widgets
3. Learn More, Get Involved!

ITK: *Insight into Images*

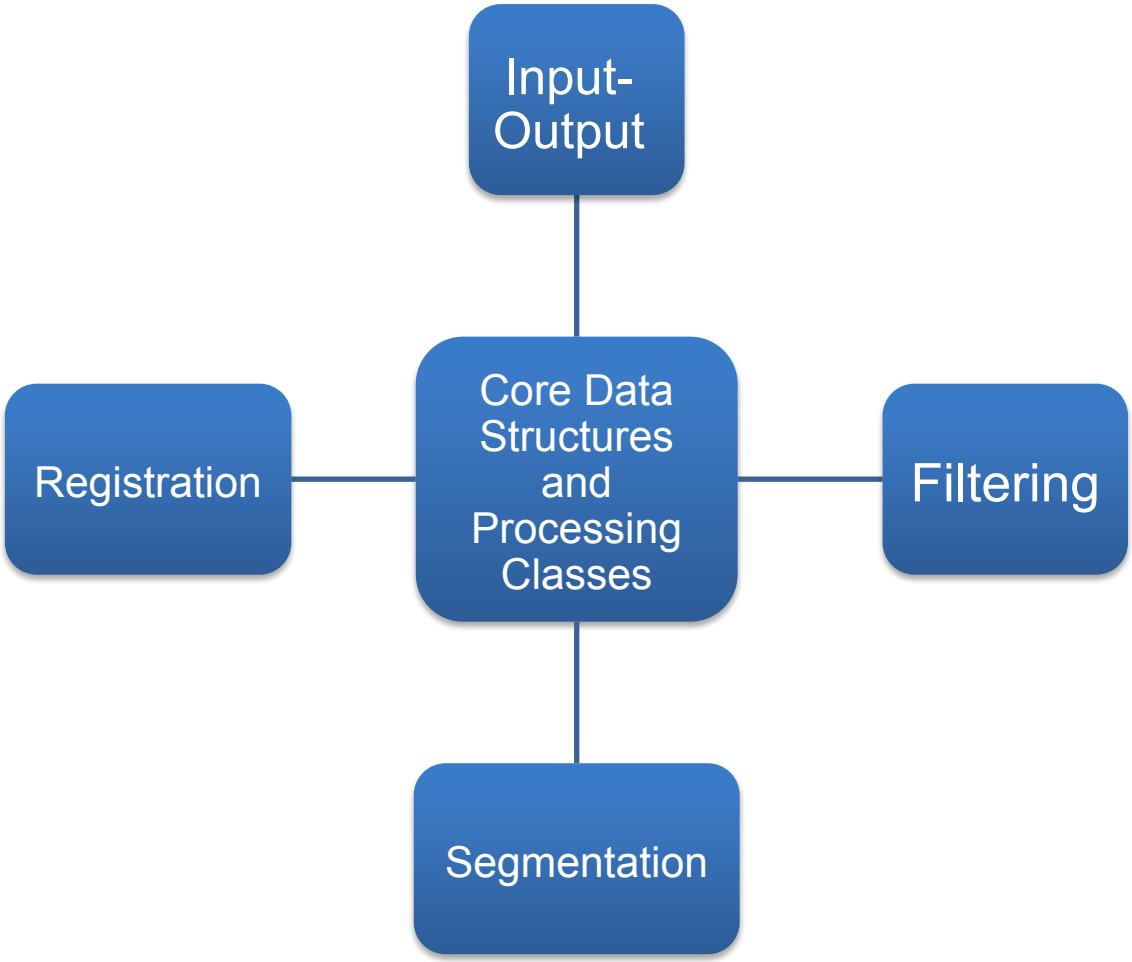
Kitware

What is the Insight Toolkit (ITK)?

- The Insight Segmentation and Registration Toolkit (ITK) is an open-source, freely available, cross-platform system for high-performance, N-dimensional image analysis
- Extensive suite of algorithms for processing, registering, segmenting, analyzing, and quantifying scientific data.
- *https://www.itk.org/*

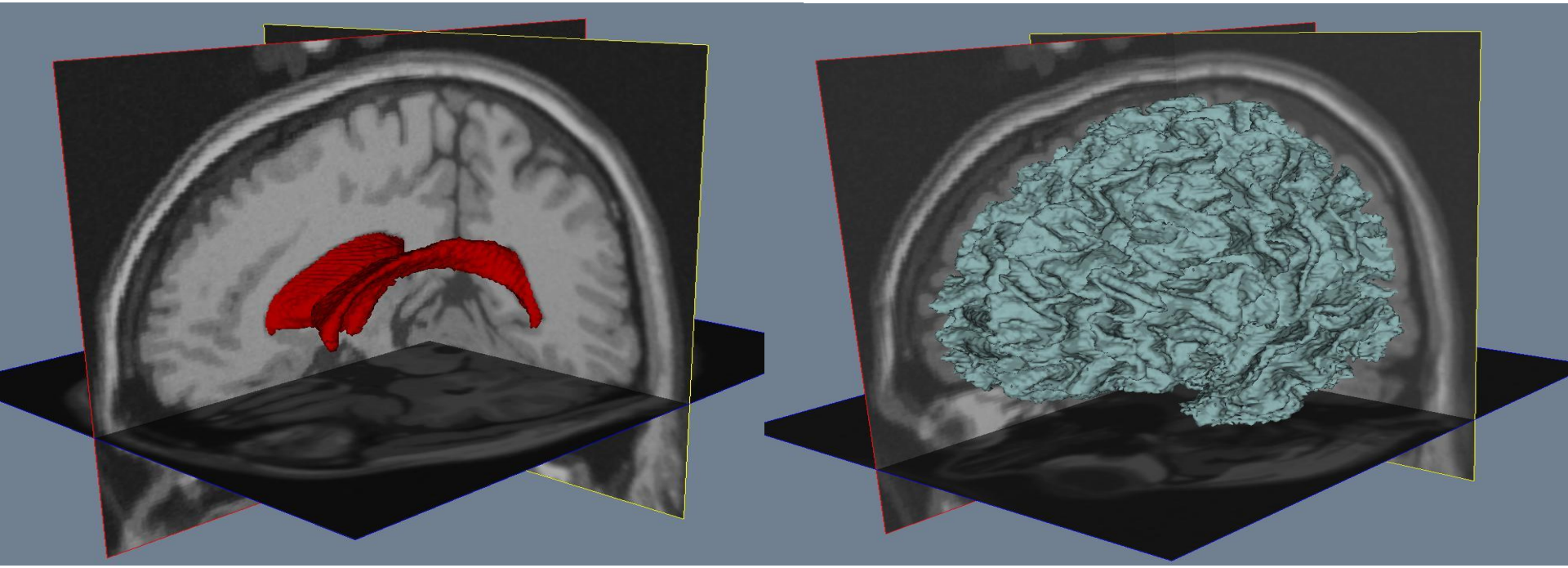
Kitware

What are ITK's primary features?



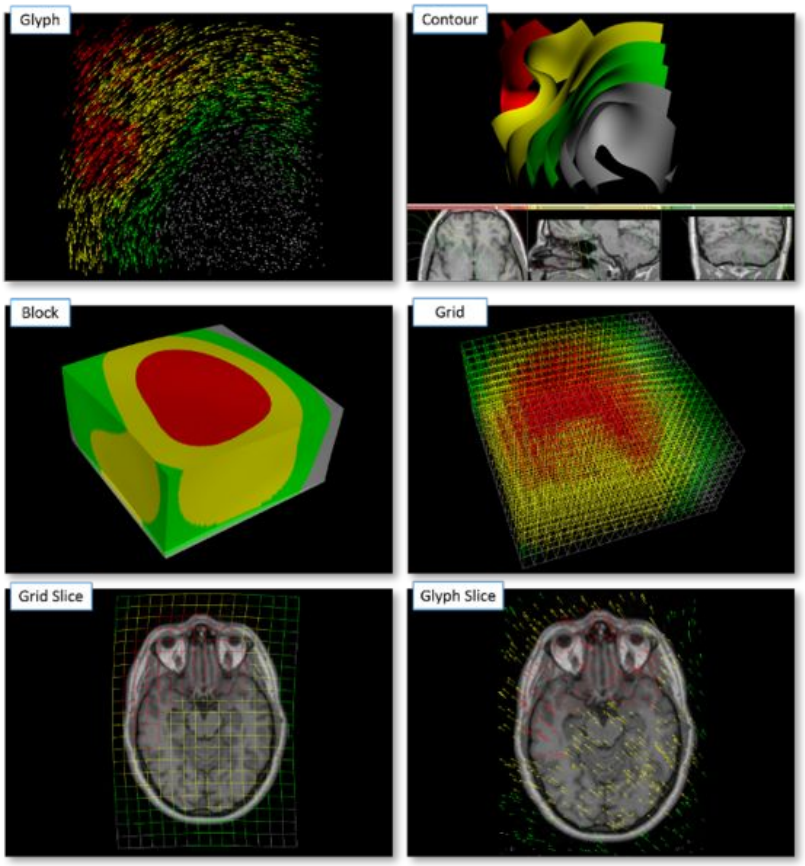
Kitware

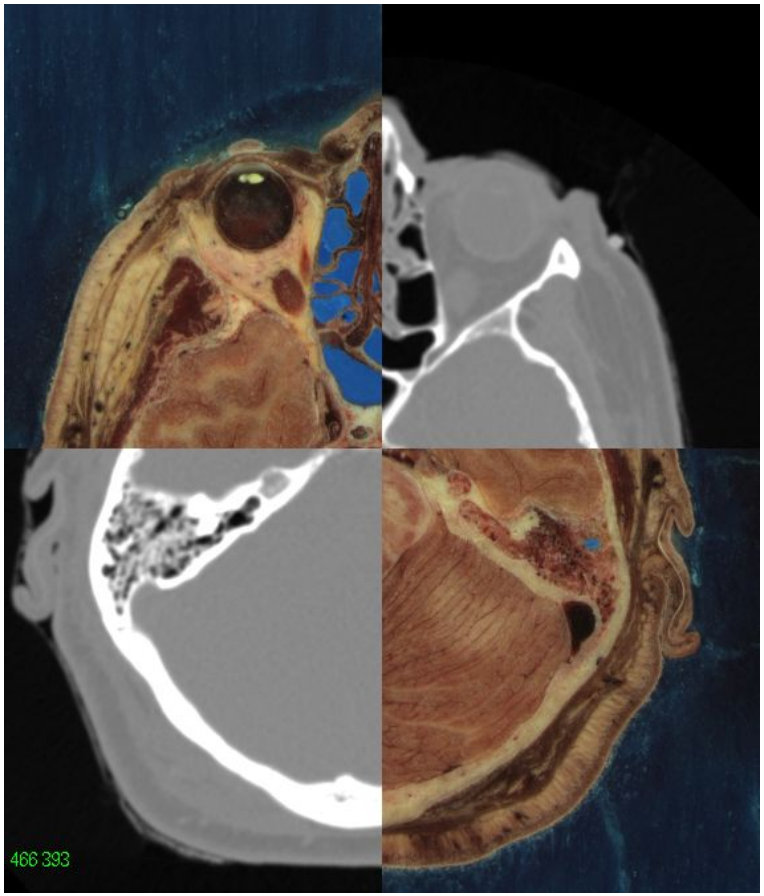
Segmentation



Kitware

Registration



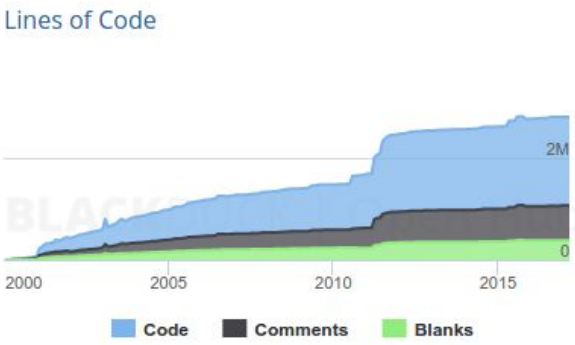


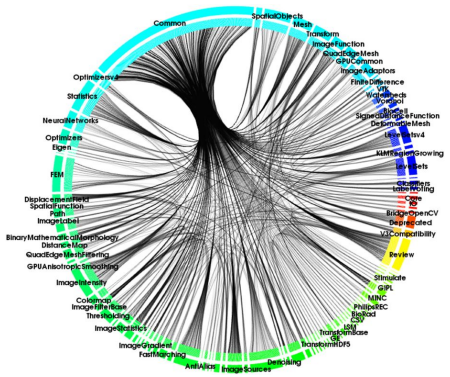
ITK Transforms Visualized in 3D Slicer

Kitware

Code and algorithms - how much?

- 1.7 million lines of code[1]
- Estimated 497 years of person-effort (COCOMO model)[1]
- First commit in January, 2000[1]
- 48,690 commits[2]
- Over 130 modules[3]





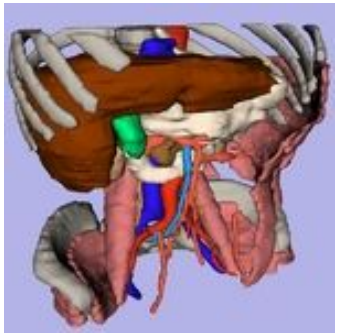
Module dependencies[2]

1 https://www.openhub.net/p/itk
2 https://github.com/InsightSoftwareConsortium/ITK
3 https://doi.org/10.3389/fninf.2014.00013

Kitware

Image input-output formats - 22

- BMP
- NIFTI
- MRC
- DICOM
- Stimulate
- GE formats
- TIFF
- VTK
- IPL
- GIPL
- LSM
- HDF5
- PNG
- MINC
- BioRad
- MetaImage
- NRRD
- RAW
- Siemens
- SCIFIO-supported
- JPEG
- PhilipsREC



Kitware

Types of filtering algorithms - 16

Comparison	Smoothing	Distance Map	Image Label
Thresholding	Mathematical Morphology	Gradient	Compose
FFT	Bias Correction	Interpolation	Noise
Denoising	Convolution	Deconvolution	Features

Kitware

Types of segmentation algorithms - 6

LevelSets

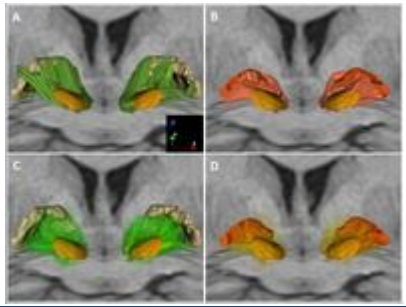
Watersheds

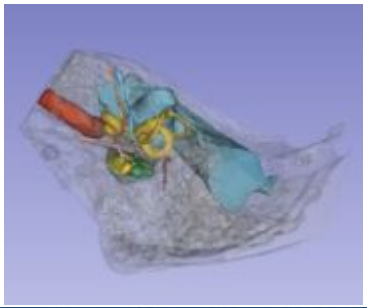
Connected Components

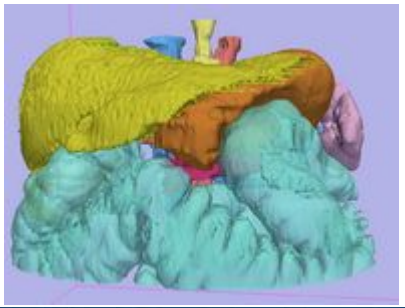
Label Voting

Region Growing

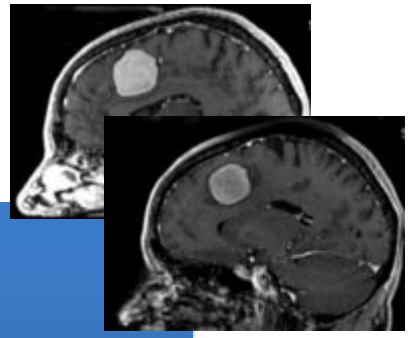
Classifiers





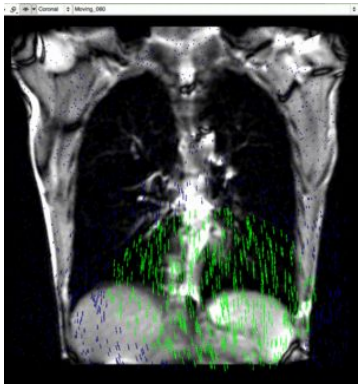


Types of registration algorithms - 3

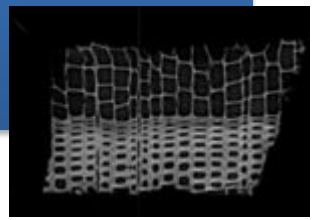


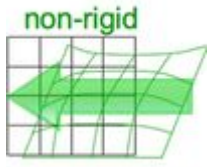
Registration Optimization Framework

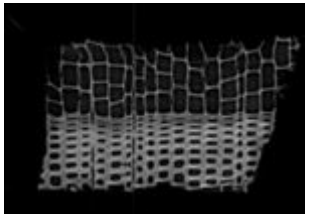
PDE Deformable Registration (Demons)



FEM Registration

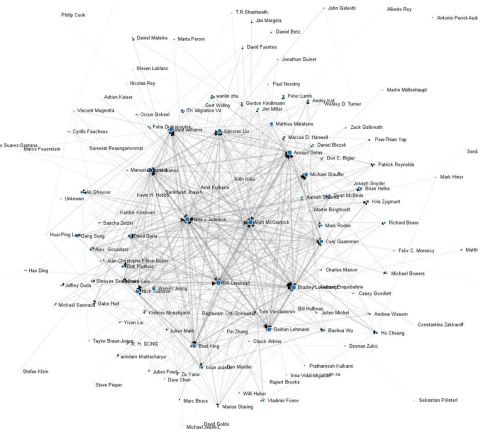




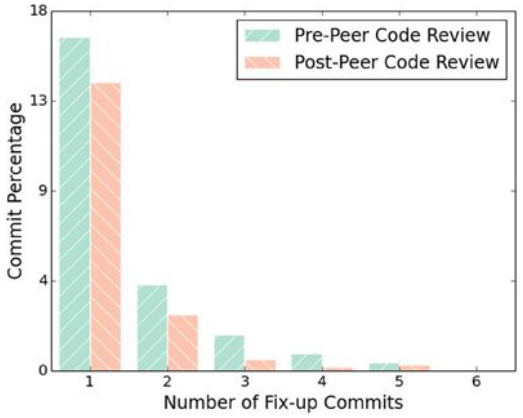


What is the size of the community and what are the software quality practices?

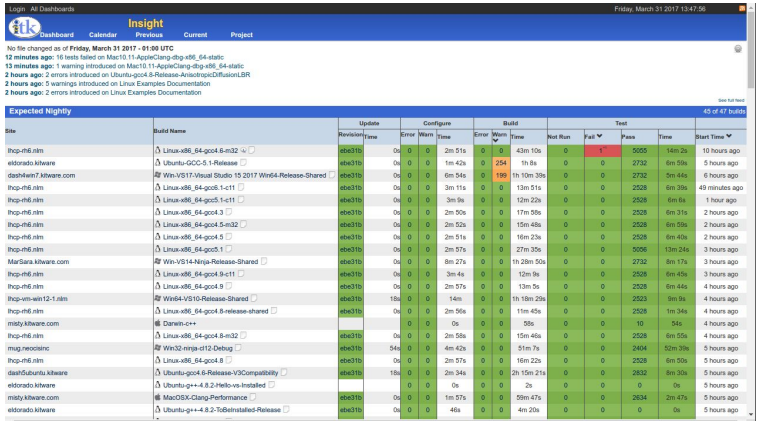
- 2,818 regression tests[1]
- 88.63% code testing coverage[1]
- Over 6,000 code reviews[2]



Peer code reviews over 3 years[2]



Fix-up commits before and after code review[2]



Nightly testing results on CDash[2]

1 https://open.cdash.org/index.php?project=Insight
2 https://doi.org/10.3389/fninf.2014.00013

Kitware

Where has ITK been in the SciPy community?

Initial ITK Python wrapping

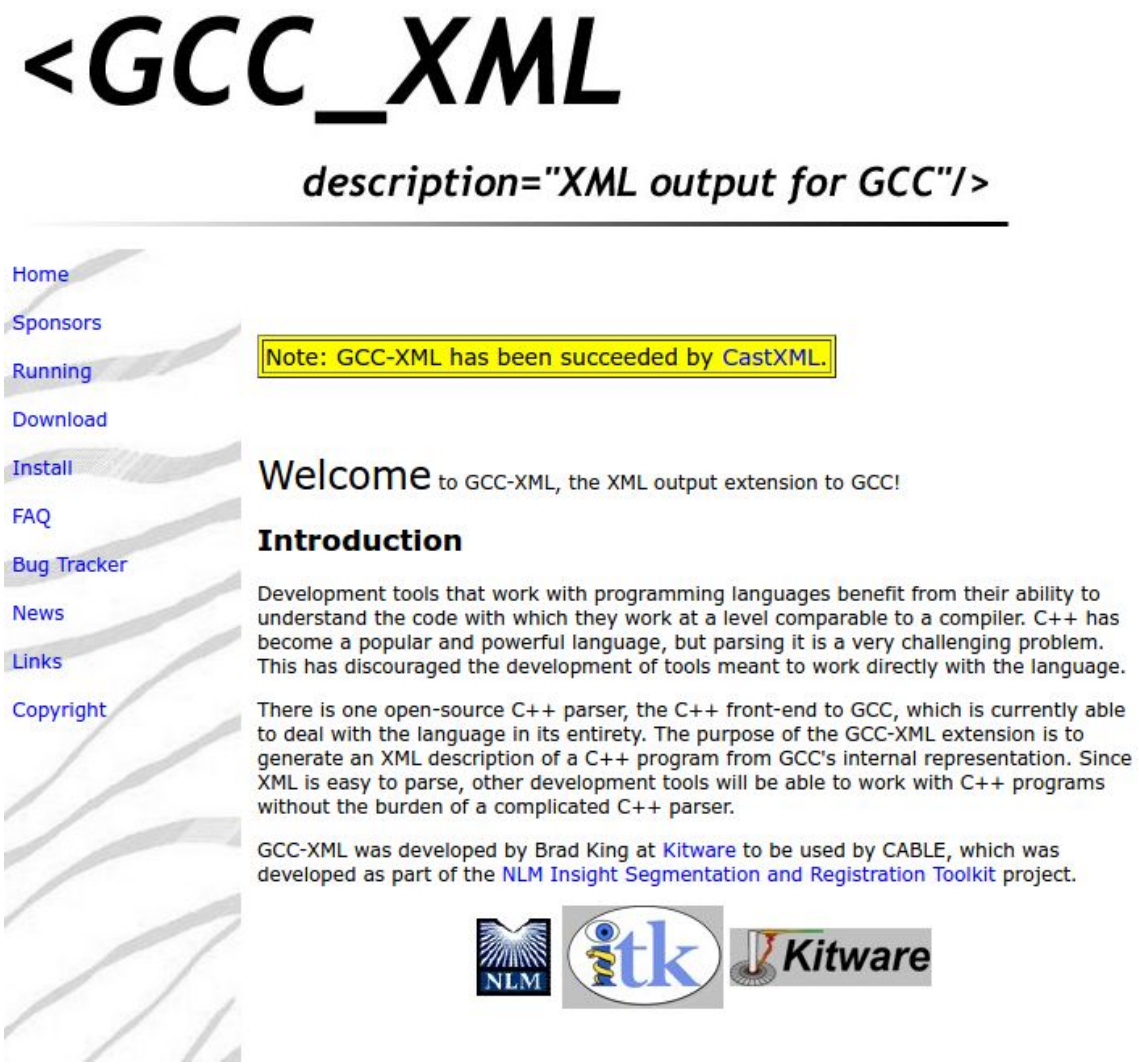
Brad King

Circa 2002

- **King B.**, Schroeder W., Automated Wrapping of Complex C++ Code, C/C++ Users Journal, 2003

Kitware

CableSWIG and GCC_XML

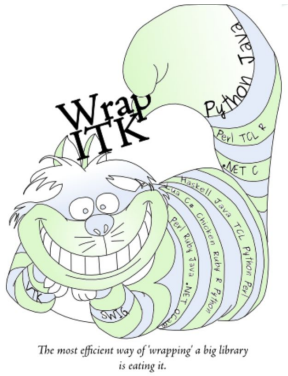


Kitware

Heroic efforts to improve the wrapping

New wrapping infrastructure:
WrapITK - 2008 to 2010

Gaëtan Lehmann



The most efficient way of 'wrapping' a big library is eating it.

pygccxml

- The original author is Roman Yakovenko (2004-2011).
- Forked multiple times by different authors to add Python 3 support.
- In May 2014, Michka Popoff and the Insight Software Consortium revived pygccxml.

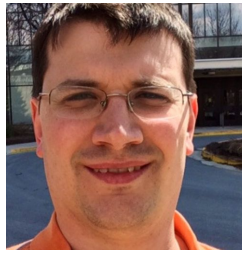
Michka Popoff

Francois Budin

Lucas Gandel

Thomas "Hastings" Greer

Mayeul Chassagnard



Bradley Lowekamp

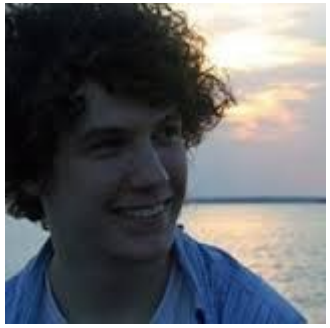
Linux and Homebrew ITK Python packages

Steve Robbins

Mario Ceresa

Gianfranco Costamagna

Gert Wollny



Christopher Mullins

fedora

archlinux

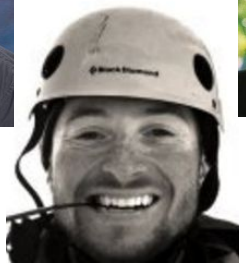
debian

ubuntu 16.04

Kitware

Breaking down barriers: scikit-build

Omar Padron



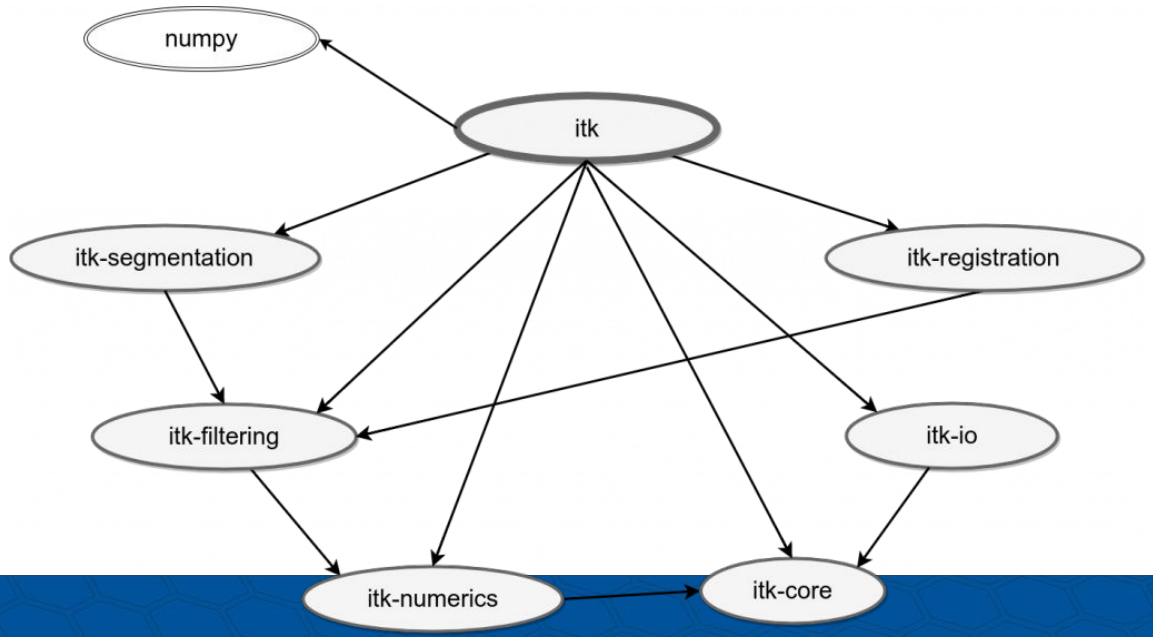
Jean-Christophe Fillion-Robin

Mike Sarahan

Anthony Scopatz

June, 2017: 🎉

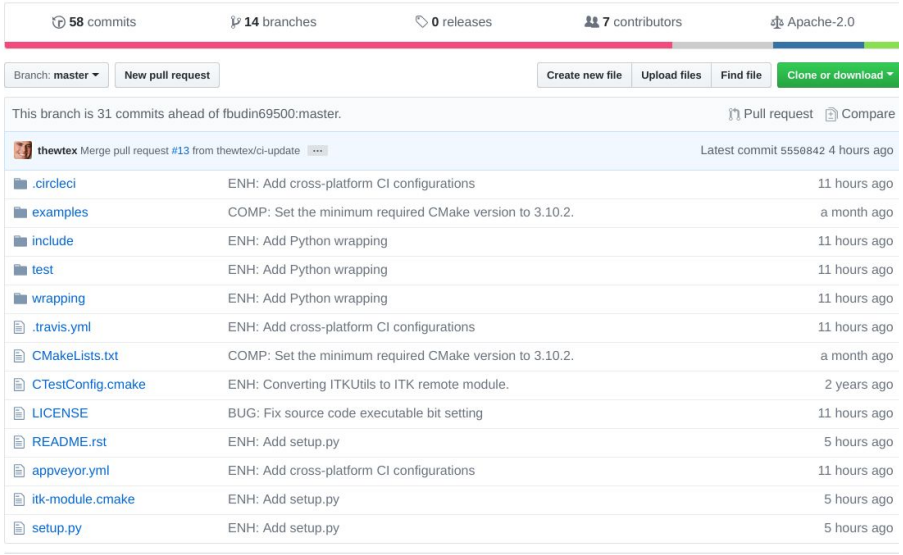
```
pip install itk

conda install -c conda-forge itk
```



Kitware

Python packages from ITK modules on GitHub

```
python -m pip install cookiecutter
python -m cookiecutter gh:InsightSoftwareConsortium/ITKModuleTemplate
# Fill in the information requested at the prompts
```



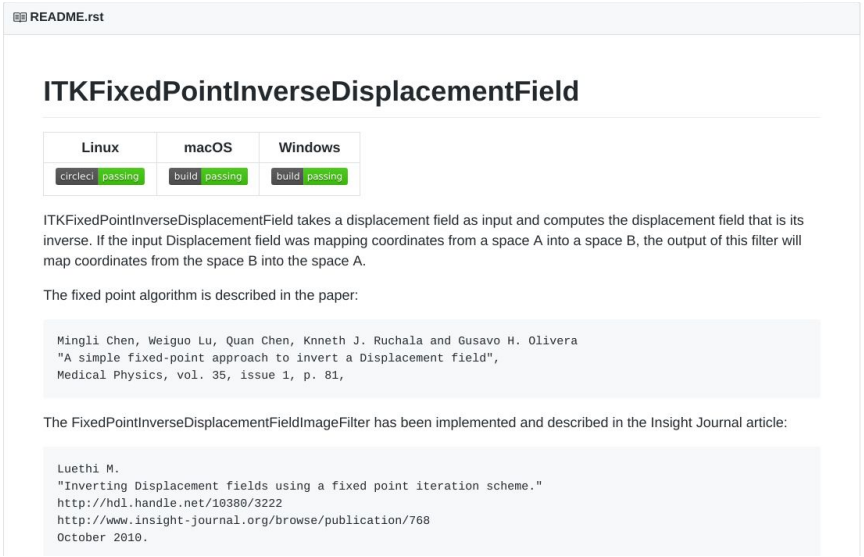
58 commits | 14 branches | 0 releases | 7 contributors | Apache-2.0

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

This branch is 31 commits ahead of fbudin69500:master. | Pull request | Compare

thewtex Merge pull request #13 from thewtex/ci-update ··· Latest commit 5550842 4 hours ago

.circleci	ENH: Add cross-platform CI configurations	11 hours ago
examples	COMP: Set the minimum required CMake version to 3.10.2.	a month ago
include	ENH: Add Python wrapping	11 hours ago
test	ENH: Add Python wrapping	11 hours ago
wrapping	ENH: Add Python wrapping	11 hours ago
.travis.yml	ENH: Add cross-platform CI configurations	11 hours ago
CMakeLists.txt	COMP: Set the minimum required CMake version to 3.10.2.	a month ago
CTestConfig.cmake	ENH: Converting ITKUtils to ITK remote module.	2 years ago
LICENSE	BUG: Fix source code executable bit setting	11 hours ago
README.rst	ENH: Add setup.py	5 hours ago
appveyor.yml	ENH: Add cross-platform CI configurations	11 hours ago
itk-module.cmake	ENH: Add setup.py	5 hours ago
setup.py	ENH: Add setup.py	5 hours ago



README.rst

ITKFixedPointInverseDisplacementField

Linux	macOS	Windows
circleci passing	build passing	build passing

ITKFixedPointInverseDisplacementField takes a displacement field as input and computes the displacement field that is its inverse. If the input Displacement field was mapping coordinates from a space A into a space B, the output of this filter will map coordinates from the space B into the space A.

The fixed point algorithm is described in the paper:

```
Mingli Chen, Weiguo Lu, Quan Chen, Knneth J. Ruchala and Gusavo H. Olivera
"A simple fixed-point approach to invert a Displacement field",
Medical Physics, vol. 35, issue 1, p. 81,
```

The FixedPointInverseDisplacementFieldImageFilter has been implemented and described in the Insight Journal article:

```
Luethi M.
"Inverting Displacement fields using a fixed point iteration scheme."
http://hdl.handle.net/10380/3222
http://www.insight-journal.org/browse/publication/768
October 2010.
```

Kitware

Python packages from ITK modules on GitHub

```
python -m pip install cookiecutter
python -m cookiecutter gh:InsightSoftwareConsortium/ITKModuleTemplate
# Fill in the information requested at the prompts
```

```
pip install %s
```

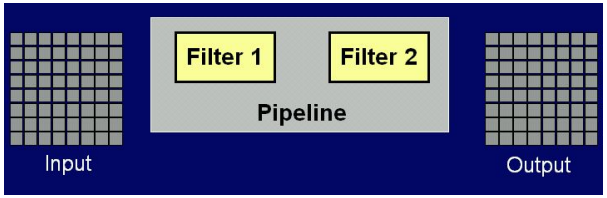
- itk-anisotropicdiffusionlbr
- itk-bonemorphometry
- itk-cuberille
- itk-isotropicwavelets
- itk-krcahsheetness
- itk-morphologicalcontourinterpolation
- itk-polartransform
- itk-ringartifact
- itk-texturefeatures
- itk-ultrasound
- itk-binarythinning3d

Kitware

Towards a more Pythonic API

Python interface reflects C++-based, object-oriented API for pipeline streaming:

```
median_filter = itk.MedianImageFilter[ImageType, ImageType].New()
median_filter.SetInput(input_image)
median_filter.SetRadius(radius)
median_filter.Update()
result = median_filter.GetOutput()
```

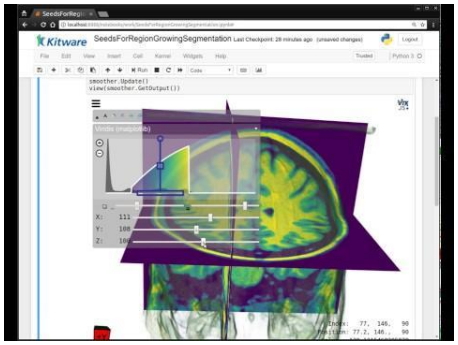


Optional, procedural, more Pythonic, snake_case interface (ITK 5):

```
result = itk.median_image_filter(input_image, radius=radius)
```

Kitware

NumPy Bridge



ITK

- Python Buffer Protocol
- Arrays
- Array Views

NumPy

OpenCV

DASK

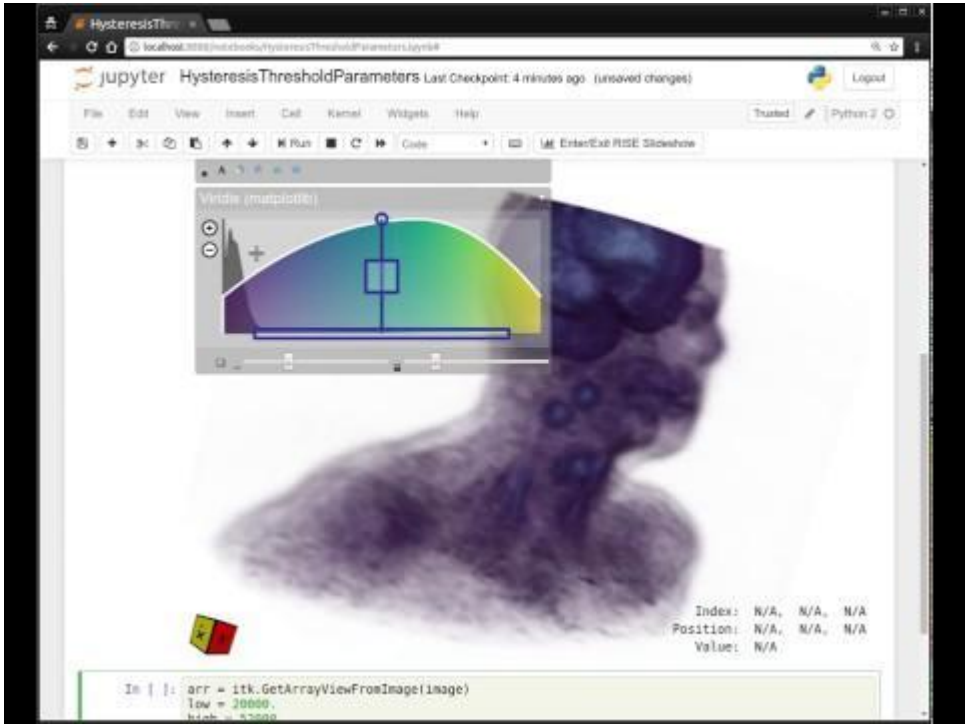
PYTORCH

TensorFlow

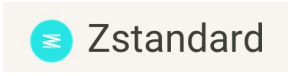
Keras

scikit learn

itk-jupyter-widgets



scikit-image: *In the context of thresholding, hysteresis means that areas above some **low** threshold are considered to be above the threshold if they are also connected to areas above a **higher**, more stringent, threshold.*



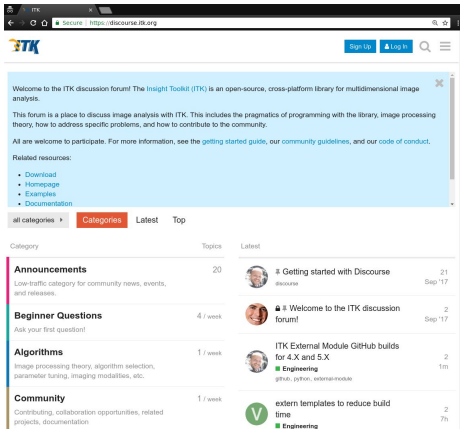
Learn More, Get Involved!

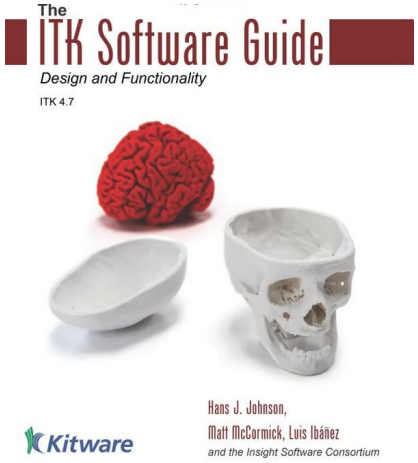
- Jupyter Tutorial - https://goo.gl/L1EwAf
- ITK Software Guide - https://itk.org/ITKSoftwareGuide/html/
- Sphinx Examples - https://itk.org/ITKExamples/
- Discourse - https://discourse.itk.org/

Slides: bit.ly/scipy2018-itk-talk

matt.mccormick@kitware.com

https://twitter.com/thewtex





Enjoy ITK!

Kitware