

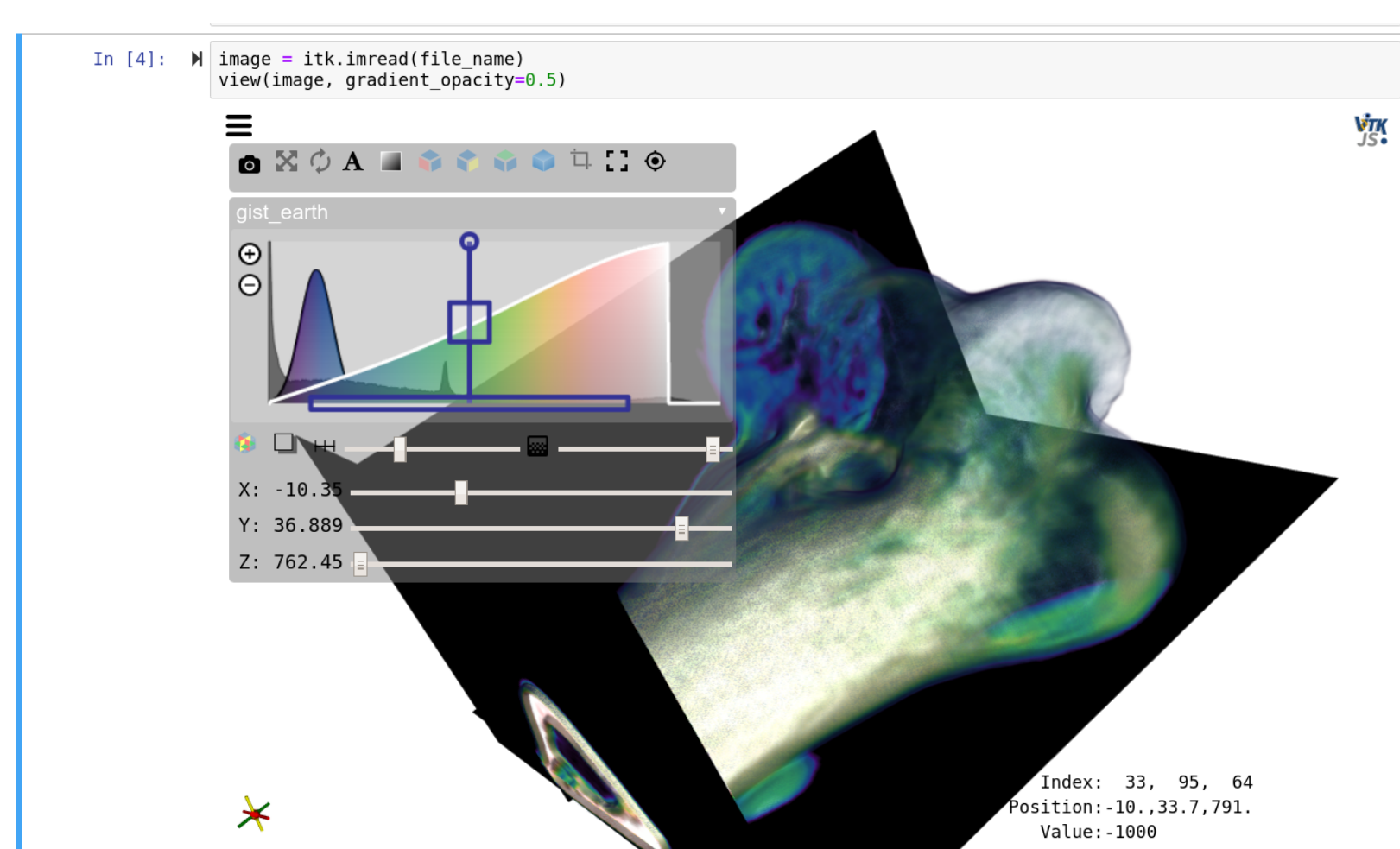
itk-jupyter-widgets: Interactive 2D and 3D Visualization for Jupyter

Matthew McCormick^{1*}, Sebastien Jourdain¹, Ken Martin¹, Alexis Girault¹,
Michael Grauer¹, Jean-Christophe Fillion-Robin¹
¹ Kitware, Inc, USA;
* matt.mccormick@kitware.com

Introduction

We present a new package for interactive visualization and inspection of 2D and 3D images, point sets, and geometry, e.g. meshes, in the Jupyter Notebook and JupyterLab: **itkwidgets**.

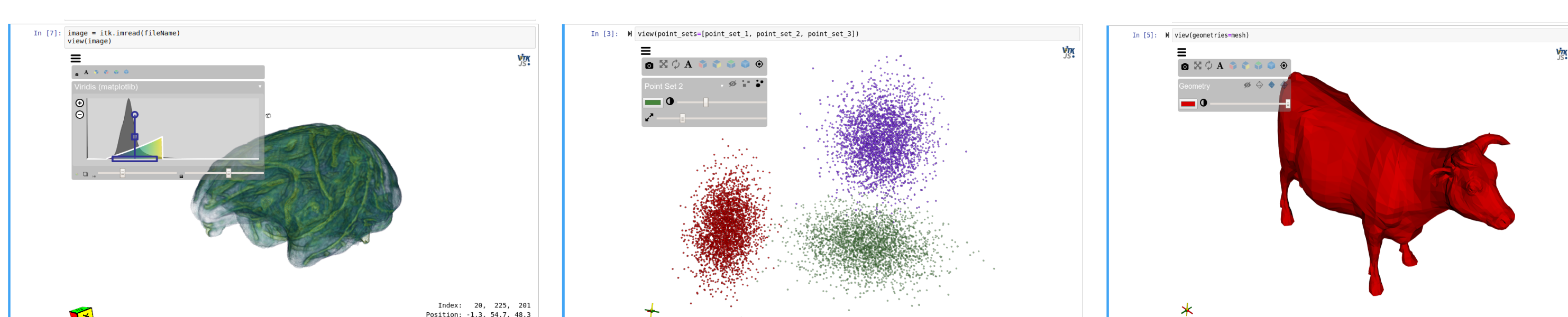
These widgets meet the needs of researchers working in the medical, biological, materials science, geoscience, and similar scientific domains who want to obtain **insights from spatial datasets**.



Bone x-ray computed tomography volume rendering and slices. Upper left: user interface controls for window/level, opacity transfer function, slice position, etc. Lower right: data probe image index, position, and value.

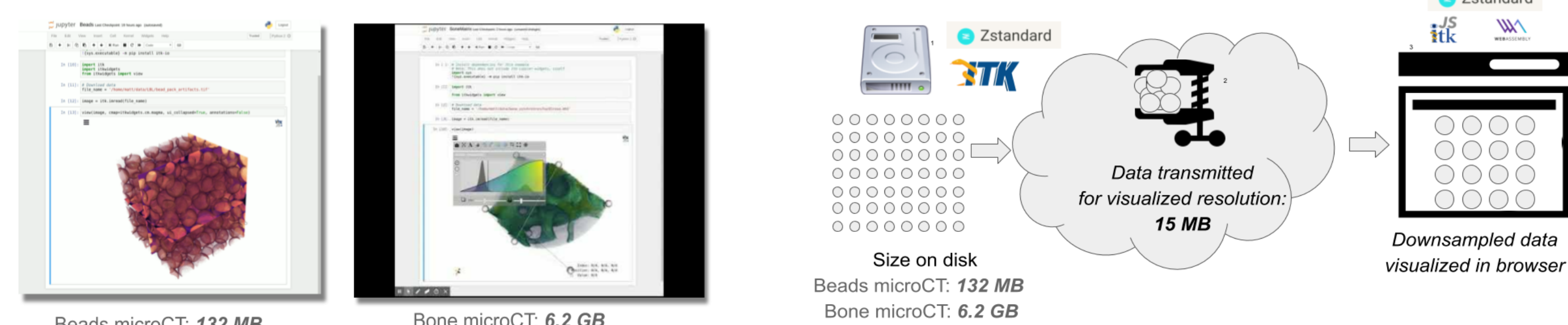
Methods

Leveraging server-side parallelism, client-side WebAssembly and WebGL, these *ipywidgets* are built browser-side on *itk.js* and *vtk.js*, JavaScript and WebAssembly interfaces and implementations of the Insight Toolkit (ITK) and the Visualization Toolkit (VTK). Standard Python Jupyter widget interfaces are provided, backed by NumPy and ITK Python processing.



Supported data types: images / volumes, point sets, and geometry / meshes.

Principles of scientific visualization are held with a focus on spatial properties in a representative physical coordinate system. This enables images with anisotropic pixel spacing, oriented images, and offset images to render correctly. This approach allows images, point sets, and meshes to all maintain their relative positions in the world coordinate system.

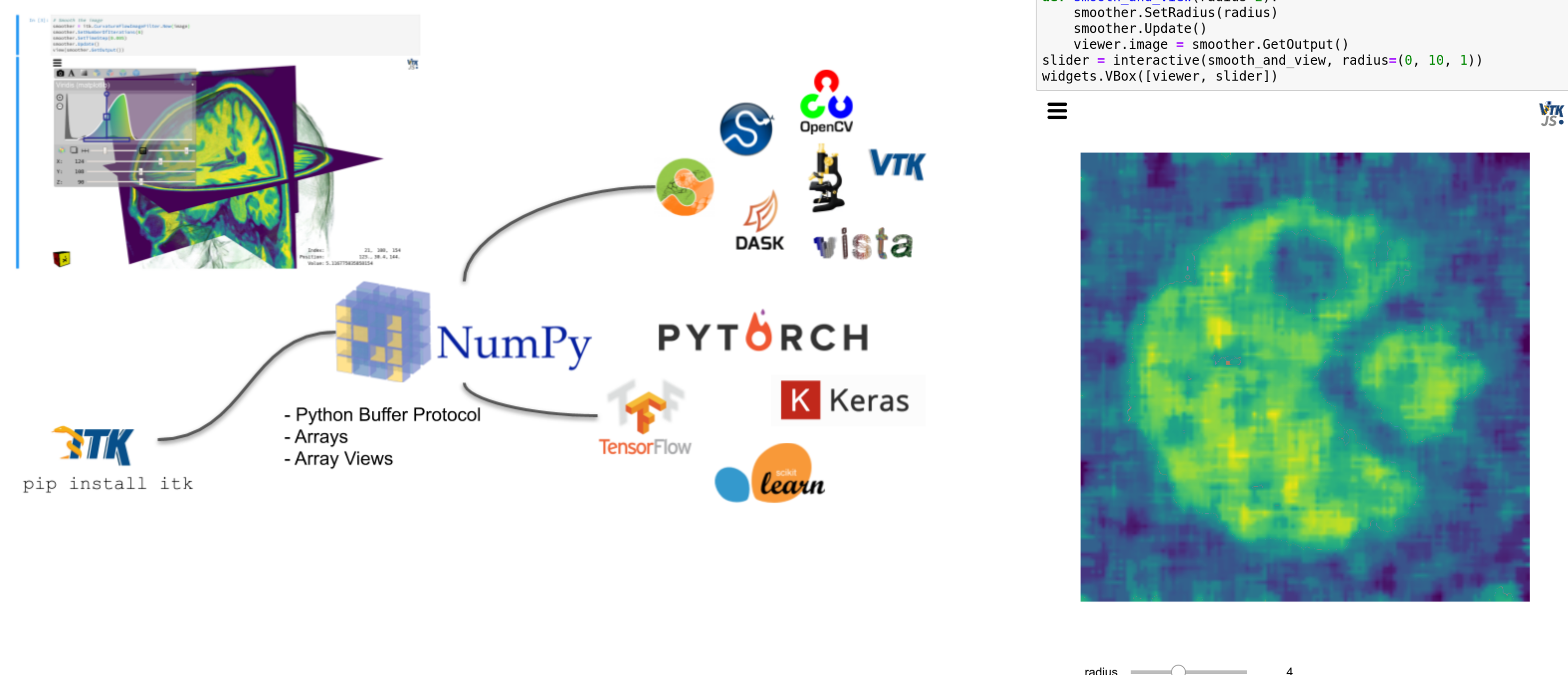


Gigabyte-scale images are supported through dynamic data reduction and compression. As a result, in situ data can be examined remotely with lightweight clients. We have found that a mobile phone can effectively render images located 600 miles away.

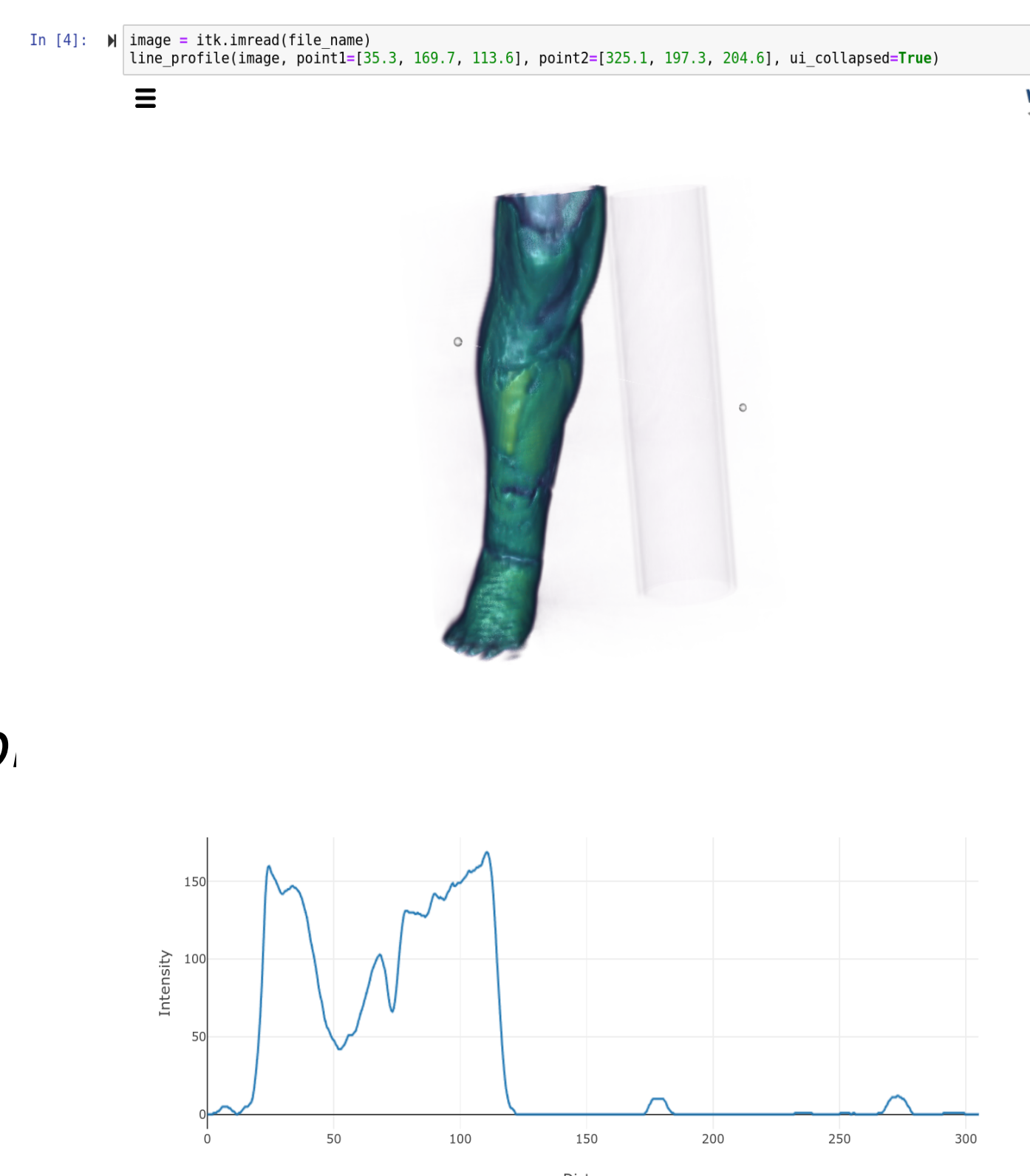
The package is open source, licensed with the Apache 2.0 license, and developed with the SciPy community.

Results

The itk-jupyter-widgets are designed to provide insights into real-world spatial data by with interactive visualizations that can be coupled with a wide variety of powerful Python tools, such as *scikit-image*, *scipy.ndimage*, *ImageJ-ImgLib2* via *imglyb*, *Dask*, *OpenCV*, *pyvista*, or *VTK*, as a result of ITK's NumPy interoperability layer. Reactive *ipywidget traitlets* coupled with an interactive HTML and WebGL interface makes custom exploration of algorithm parameters possible in only a few lines of Python.



The *ipywidget* architecture enables integration with standard widgets for interactivity and organization, Or the itk-jupyter-widgets can be mixed with many *ipywidget* extensions that provide related or alternative visualizations. For example, a *line_profile* widget is provided that combines the interactive image visualization with *ipypm*, *plot.ly*, or *bqplot*. The image intensity over a line is interactively selected and dynamically plotted.



Discussion and Future Work

By leveraging WebGL and WebAssembly, *itkwidgets* provides high-performance, interactive visualization of scientific datasets in the web browser. Synergy with the powerful, open-source image and mesh analysis libraries available in Python with this Jupyter extension facilitates a rapid understanding of spatial data.

Future work will target multi-component volume rendering, support for rendering geometry point data and cell data, and time series. There is also a need for additional interactive widget to define seed points for segmentation, landmarks for evaluation of spatial registrations, editing of segmentations, and definition and refinement of registration transforms.

`pip install itkwidgets`

`conda install -c conda-forge itkwidgets`

<https://github.com/InsightSoftwareConsortium/itk-jupyter-widgets>



Videos: <http://bit.ly/itkwidgets-videos>