doi:10.1017/S1431927617001805
Microsc. Microanal. 23 (Suppl 1), 2017

Pycroscopy – An Open Source Approach to Microscopy and Microanalysis in the Age of Big Data and Open Science

Suhas Somnath[1,2], Chris R. Smith[1,2], Stephen Jesse[1,2], and Nouamane Laanait[1,2]

[1.] The Institute of Functional Imaging of Materials, Oak Ridge National Laboratory, Oak Ridge, Tennessee, USA37831.
[2.] The Center for Nanophase Materials Sciences, Oak Ridge National Laboratory, Oak Ridge, Tennessee, USA37831.

Over the past few years, microscopy and microanalysis have undergone profound changes to enable breakthroughs in science and technology. Many of these changes are mainly driven by continued improvements in instrumentation hardware [1] as well as the increased accessibility to high-performance computing (HPC) resources[2], and more sophisticated computer algorithms [3]. These advancements have led to unprecedented proliferation in microscopy datasets both in dimensionality and size. However, in many cases the softwares supplied with the microscopes are typically very expensive, lack advanced or user-defined data analysis routines, and store experimental data in proprietary formats. Consequently, these proprietary software and data formats not only impede access to and analysis of data but also hinder continued research and instrument development, especially in the age of "big data" and open science. Therefore, ushering the promise of data-intensive microscopy and microanalysis research requires general and robust data curation, and analysis platforms that are HPC-ready and open source.

We have developed a software package called *pycroscopy* that uses an open-source approach to analyzing and storing multidimensional microscopy data. *pycroscopy* is freely available via popular software repositories PyPi (https://pypi.python.org/pypi/pyCroscopy/) and GitHub (https://github.com/pycroscopy/pycroscopy), and therefore lifts any financial burden for handling data. It uses an open and hierarchical data format (HDF) that can be accessed easily using many programming languages, scales well from kilobyte to terabyte sized datasets, and is commonly used in HPC environments unlike proprietary data formats. In *pycroscopy*, the data is stored as a two dimensional matrix (position x spectral value) and is linked to auxiliary datasets that describe the position and spectral value at any point. This approach facilitates the storage of data of arbitrarily large position or spectroscopic dimensionality, thereby making it an instrument-independent and universal data format. This data format greatly simplifies the correlation of data acquired from multiple instruments, a necessary ingredient in comprehensive studies of materials. More crucially, *pycroscopy*'s data format is curation-ready and therefore both meets the guidelines for data sharing and satisfies the implementation of digital data management issued to federally funded agencies.

Another advantage of an instrument- and dimensionality-independent data format is that the data is immediately compatible with most existing analysis and processing functions. The generality of *pycroscopy* provides microscopists access to a vast and growing library of community-driven data processing and analysis routines that far exceed those provided by instrument manufacturers and are desperately needed in the age of big data. Although there are many open-source software packages, most are instrument- or mode- specific, limited to 2D images or specific kinds of 3D data, are not fundamentally designed to handle datasets of large size or dimensionality, do not support scalable computation from laptops to HPCs, are not well packaged for easy installation, or do not have detailed

and comprehensive documentation. The use of python as the base language of *pycroscopy* allows the user to enjoy the best of both worlds - the ease of writing complex code in Python's simple syntax, and the ease of integrating code written in other languages into *pycroscopy*.

Currently, *pycroscopy* hosts a large collection of functions including image processing, atom-finding functions, simple harmonic oscillator -based analysis of spectroscopic AFM data, Bayesian inference methods to extract the actual resistance from high-speed scanning probe microscopy current-voltage measurements, etc. In Figure 1, we demonstrate a simple example of *pycroscopy*'s capabilities, whereby leveraging of multivariate statistical analysis allows us to substantially reduce noise in images in a rigorous and quantitative manner.

Until recently, the microscopy and microanalysis communities suffered due to the limited access to advanced algorithms or a slow rate of software development. This was partly due to the inability to share code since most research groups independently (re-)wrote similar code to solve similar problems in similar but incompatible languages. *pycroscopy* is aimed at serving as a central repository for the microscopy and microanalysis communities to openly share data processing and analysis algorithms and collectively accelerate scientific discoveries in the age of big data and open science.

This research was conducted at the Center for Nanophase Materials Sciences, which is sponsored at Oak Ridge National Laboratory by the Scientific User Facilities Division, Office of Basic Energy Sciences, U.S. Department of Energy.

References:

[1] S. V. Kalinin, et al., ACS Nano (2016), p. 9068-9086.
[2] A. Klimentov, et al., presented at the Journal of Physics: Conference Series, 2015 (unpublished).
[3] S. B. G. Kalinin S.V., Archibald R.K., Nat Mater **14** (2015), p. 973.

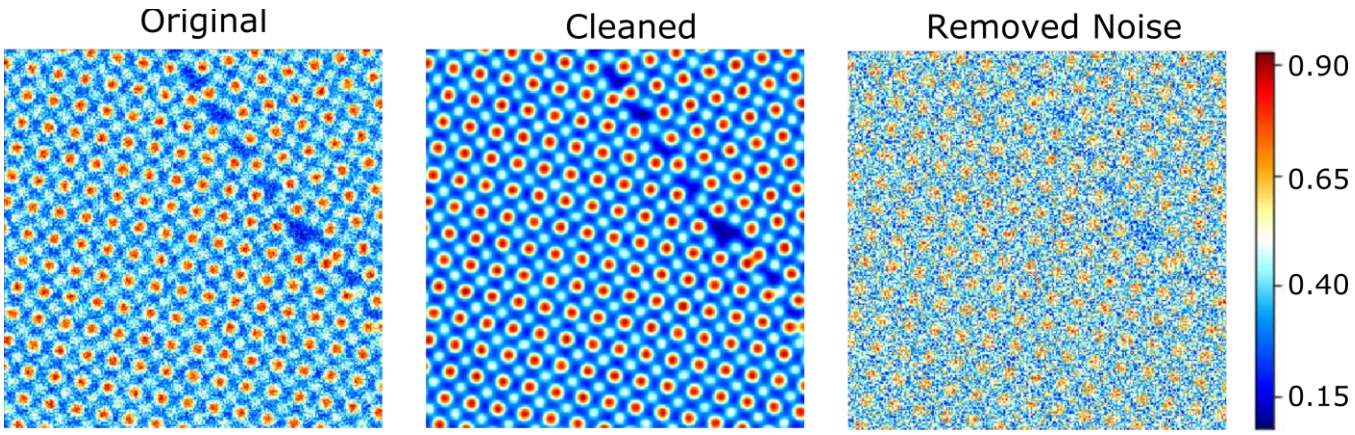

Figure 1. Image denoising using *pycroscopy*. (left) Original, raw scanning transmission electron micrograph showing multiple atoms. (center) Image cleaned using functions in *pycroscopy* showing substantially reduced noise. (right) Noise removed from original image.