

Novel SEM and DualBeam experiments with Python

R. Geurts¹, E.J.R. Vesseur¹

¹. Thermo Fisher Scientific, Achtseweg Noord 5, 5651 GG Eindhoven, Netherlands

* Corresponding author: remco.geurts@thermofisher.com

Python is one of the most popular scripting languages in research and the language of choice for data science. Python has excellent community supported open libraries for scientific computing, data analysis, image processing, machine learning and equipment interoperability. Thermo Fisher Scientific recently introduced AutoScript 4 as a toolkit for SEM and DualBeam automation. AutoScript 4 distinguishes itself by a comprehensive hierarchical set of microscope functions written as a importable Python package similar to any of the highly popular packages Numpy, SciPy and OpenCV. The clear syntax makes it easy for new users to adopt. With a minimum amount of coding it is now possible to fully automate the microscope for many advanced experiments. Python as such allows for easy interoperability of external devices, many of which already support Python. When required Python can be easily extended with additional, often open source libraries relevant to the user's application.

Here we illustrate some examples of how users can take advantage of SEM and DualBeam Python scripting to develop new innovative experiments.

With Python scripting it is now possible to perform advanced semi real-time image processing which can lead to many different types of outputs. One can search for specific edges, specific shapes, pores, etc. The outputs can be directly used to perform the next microscope action. This functionality can be applied in live feature tracking, feature or pixel level displacement measurements or adaptive scanning, where pixels with potential information are scanned (Fig. 1).

DualBeam workflows have gained an even higher level of automation flexibility than in the past, allowing for direct creation of new microstructures or nanopatterns. Parameterized mathematical shapes can be calculated and patterned on the fly (Fig. 2b), or alternatively the image can be used to determine the position of the next shape or type of next shape to be patterned (Fig. 2a). Any experimental condition set or measured can be stored and analyzed, or fed back to the experiment. Python allows for full insight into the experimental data of the acquired image, which can be saved to disk or visualized as a graph.

AutoScript 4 Python based scripting enables researchers to design fast reproducible experiments that enhance the quality of results and productivity. But even more important it allows to develop new unique experiments that were previously not possible.

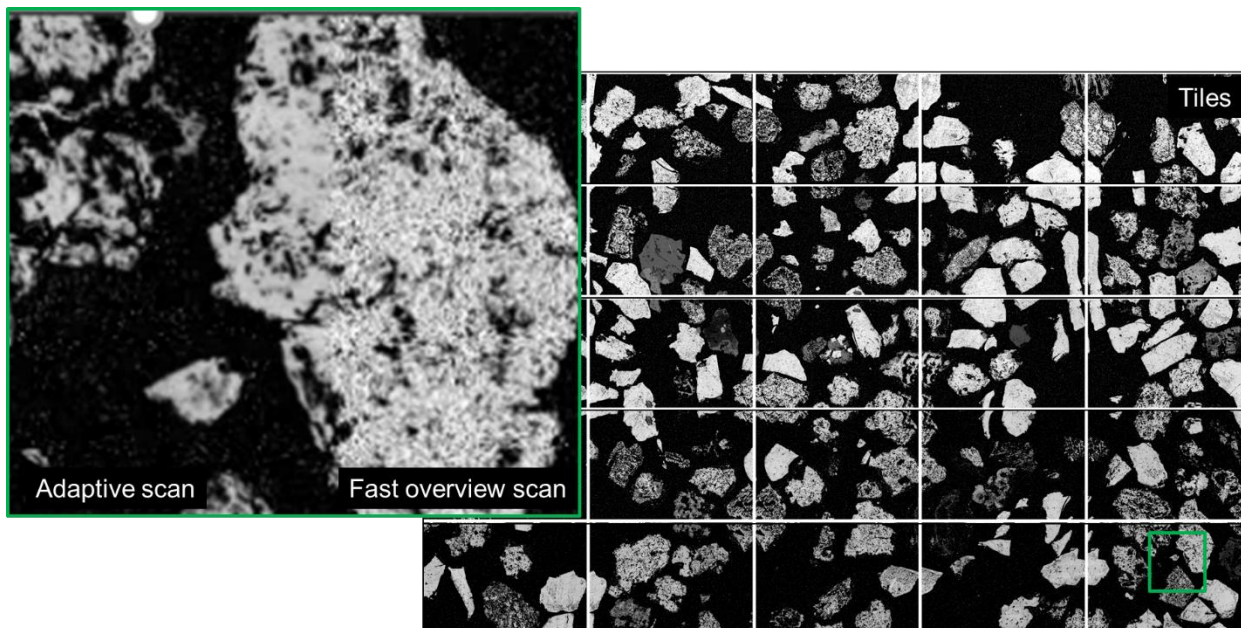


Figure 1. Combined adaptive scanning and stage tiling. From a mineral in epoxy the minerals (bright) are scanned while the epoxy resin (dark) is skipped. This results in a significant decrease of acquisition time.

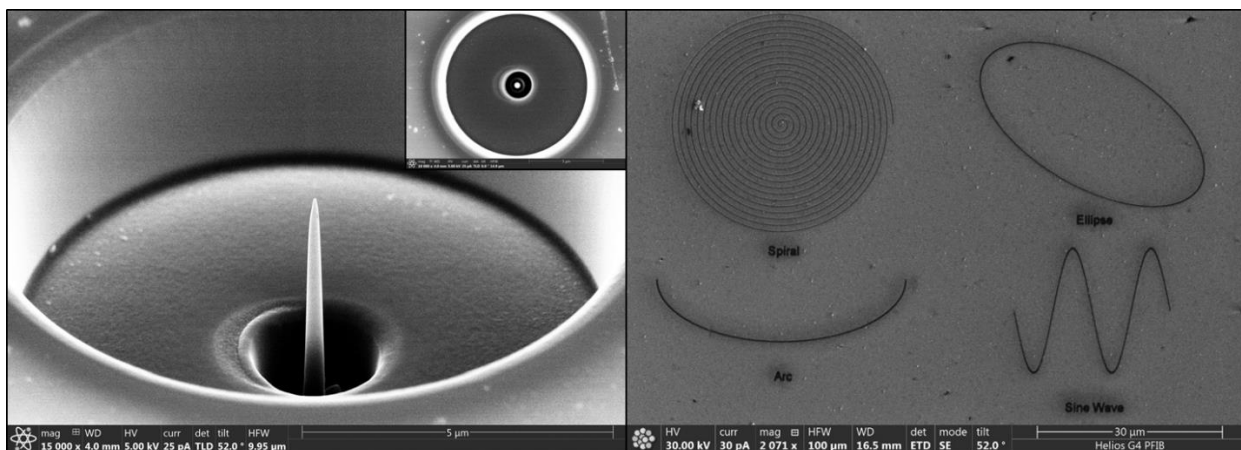


Figure 2. a. (left) Tip milling with opensource drift control algorithms, b. (right) Parameterized mathematically defined shapes are calculated by Python and instantly milled by FIB.