doi:10.1017/S1431927613006089

Microsc. Microanal. 19 (Suppl 2), 2013

Programming for Microscopy and Microanalysis

P.A. Shade[1], M.D. Uchic[1], A.R. Shiveley[2] and M.A. Groeber[1]

[1.] Air Force Research Laboratory, Materials & Manufacturing Directorate, Wright-Patterson AFB, OH 45433, USA
[2.] UES, Inc., 4401 Dayton-Xenia Rd., Dayton, Ohio 45432, USA

Programming or scripting of data collection is common practice for most experimental methodologies. This perhaps has not been as widespread in the context of user-developed automation control of OEM electron- and ion-optic microscopes, although this paradigm is changing rapidly in order to accomplish complex and unique experiments. In our experience, the challenge of programming electron- and ion-microscopes is usually condensed down to mastery of a few key operations. These include optimization of the imaging environment, image collection, image post-processing (including pattern recognition algorithms), stage/beam positioning, and remote triggering of ancillary equipment.

One example where microscope programming has proved extremely fruitful over the past ten years has been in the area of small-sample fabrication, where the dual beam FIB-SEM has been utilized to machine micron-scale test structures into the surface of bulk materials [1]. Microscope scripts have been developed to perform a combination of stage and beam positioning operations, with pattern-recognition-based optimization for accurate placement of milling patterns, resulting in precise control of the test sample dimensions. While these procedures could be performed by a skilled operator, the ability to conduct them in succession through automated loops allows the microscope to operate continuously, which can dramatically increase the rate of production. Furthermore, the implementation of algorithms such as pattern-recognition based positioning enables more precise and consistent milling beyond what one could manually achieve due to human subjectivity. Example structures are shown in Figures 1A and 1B.

A second example is the development of scripts to automate the collection of 3D microstructure data via serial sectioning experiments [2]. Here, many of the same basic functions from the previous example have been utilized, with the additional requirement to be able to remotely trigger ancillary equipment such as an EBSD system. An example 3D EBSD dataset is shown in Figure 1C. The use of network communication (such as remote triggering) in microscopy programming provides users with tremendous flexibility to design custom experiments that can incorporate multiple OEM detectors or other ancillary systems. Looking to the future, microscopy programming will also include on-the-fly analysis to enable real-time optimization of the modality, frequency, and spatial location of collected data, which is especially critical for experiments like serial sectioning that consume the analysis volume [3]. Additional examples will be discussed as time allows.

References

[1] M.D. Uchic and D.M. Dimiduk, Mat Sci Eng A **400-401** (2005) 268-278.
[2] M.D. Uchic *et al.*, Scripta Mater. **55** (2006) 23-28.
[3] M.D. Uchic *et al.*, 1st International Conference on 3D Materials Science (2012) 195-202.

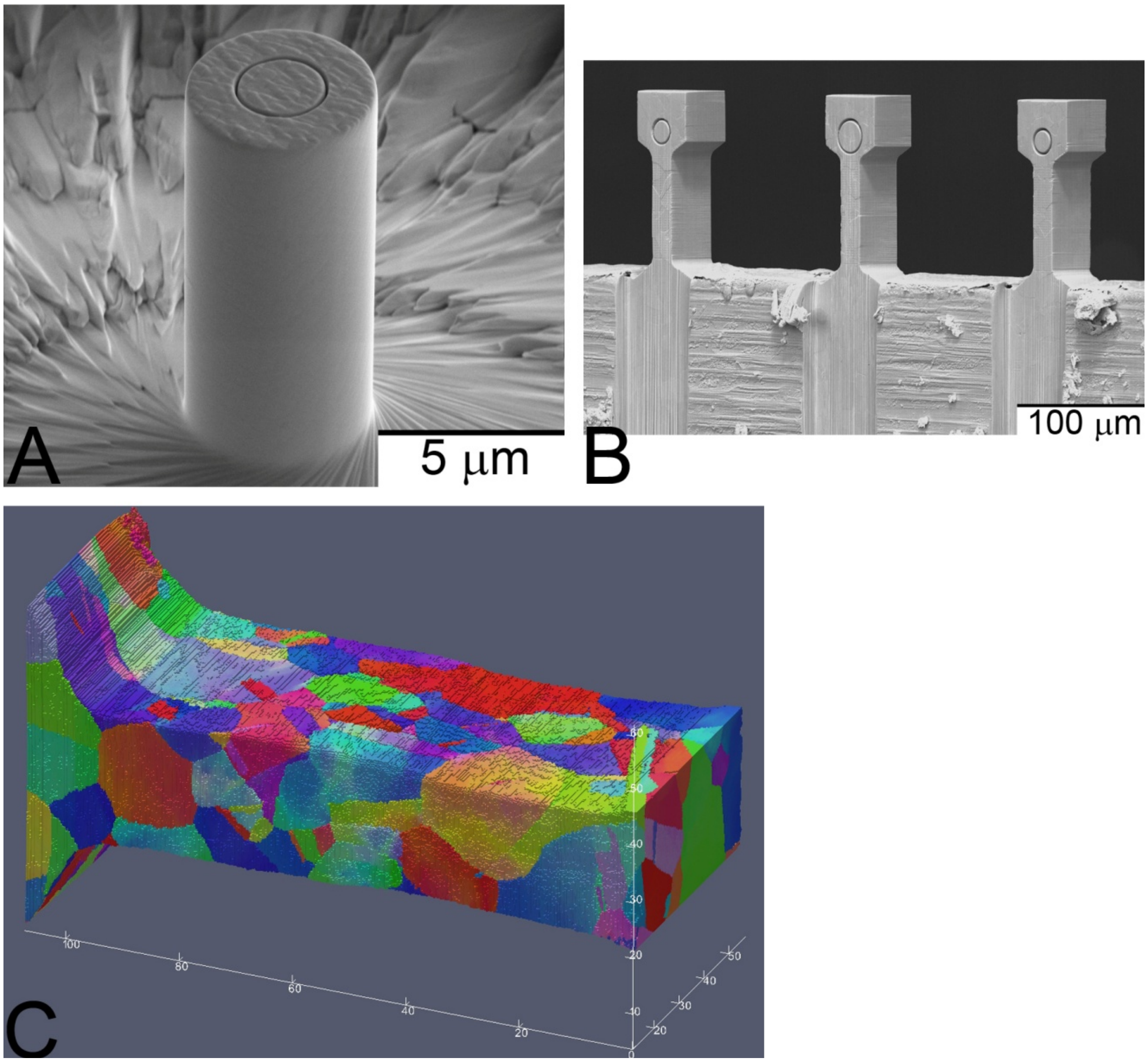

Figure 1. A) 5 μm diameter compression specimen machined into the surface of a single crystal Ni-base superalloy bulk sample with a dual beam FIB-SEM utilizing an automated lathe milling procedure. B) Array of micro-tensile specimens machined from a 50 μm thick Ni polycrystalline foil using a dual beam FIB-SEM and an automated procedure to iteratively cross-section mill the specimen perimeter while maintaining a biased back-tilt of 1°, producing nearly perfectly orthogonal sidewalls. C) Surface reconstruction of one of the specimens in B (~ 21 x 38 x 80 μm gage volume) after deforming to a total axial strain of 12.1%, where the colors represent different crystallographic orientations. The 3D data was collected using a dual beam FIB-SEM equipped with an EBSD system and a suite of scripts to automate the process of serial sectioning, EBSD data collection, and communication between the independent control systems.