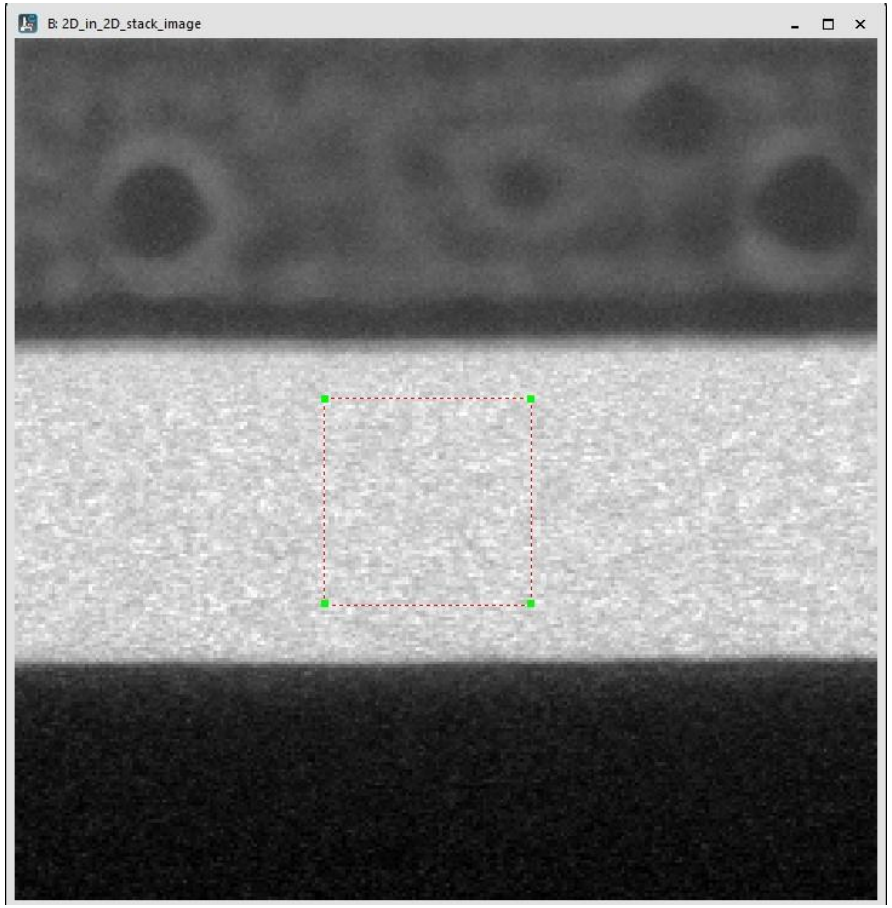
Transformation into a spectrum image

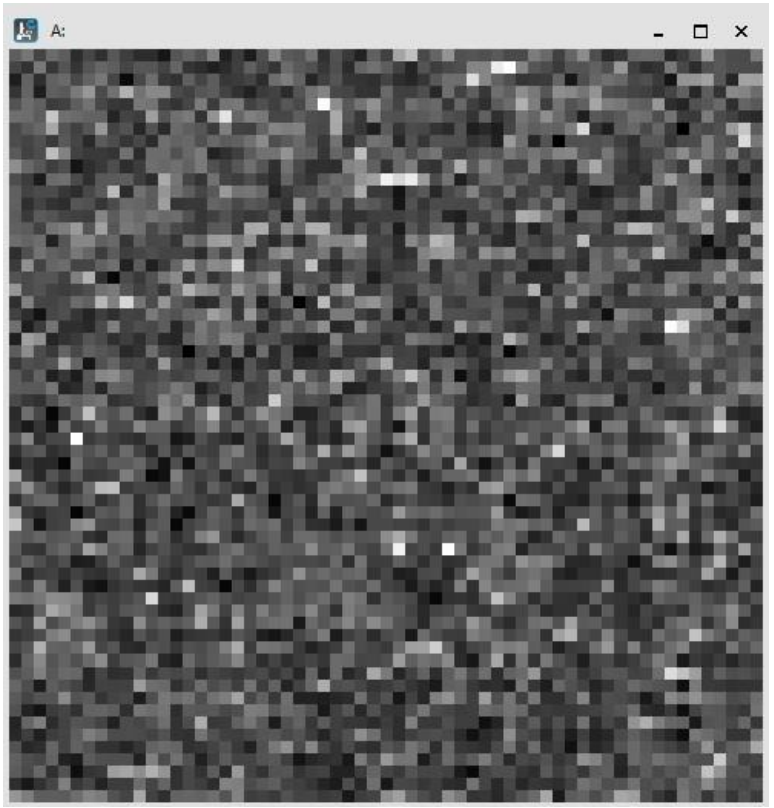
Requirements: Python-integrated GMS 3, Numpy, Scipy, Tifffile, Tkinter

1) **load 4D-STEM data in GMS 3 (dimensions = (sx, sy, dsx, dsy))**
2) **crop 4D-STEM data for a ROI (optional)**
3) transform 4D-STEM data into a spectrum image

- dimensions of 4D-STEM data = (sx, sy, dsx, dsy)
- sx, sy → STEM scanning size
- dsx, dsy → diffraction pattern (DP) size



ROI_crop_SI.s

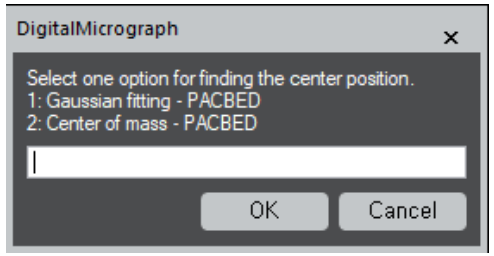


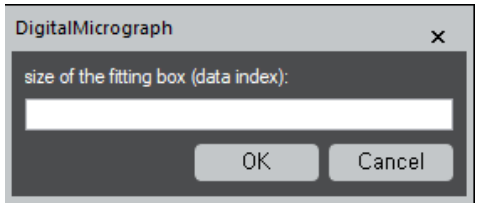
Transformation into a spectrum image

Requirements: Python-integrated GMS 3, Numpy, Scipy, Tifffile, Tkinter

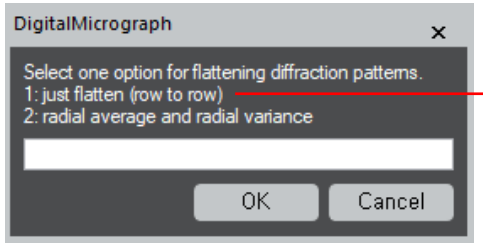
1) load 4D-STEM data in GMS 3 (dimensions = (sx, sy, dsx, dsy))
2) crop 4D-STEM data for a ROI (optional)
3) **transform 4D-STEM data into a spectrum image**

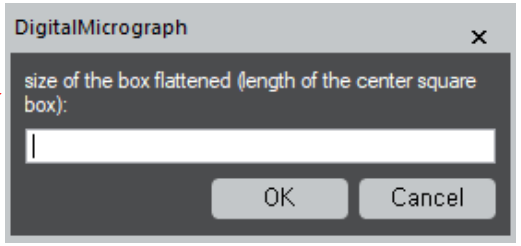
- find the center (restrict the fitting region)



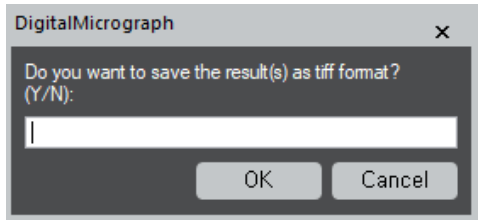


- choose a transformation option
  - just flatten (flatten row to row for a specified box)
  - radial average profile and radial variance profile





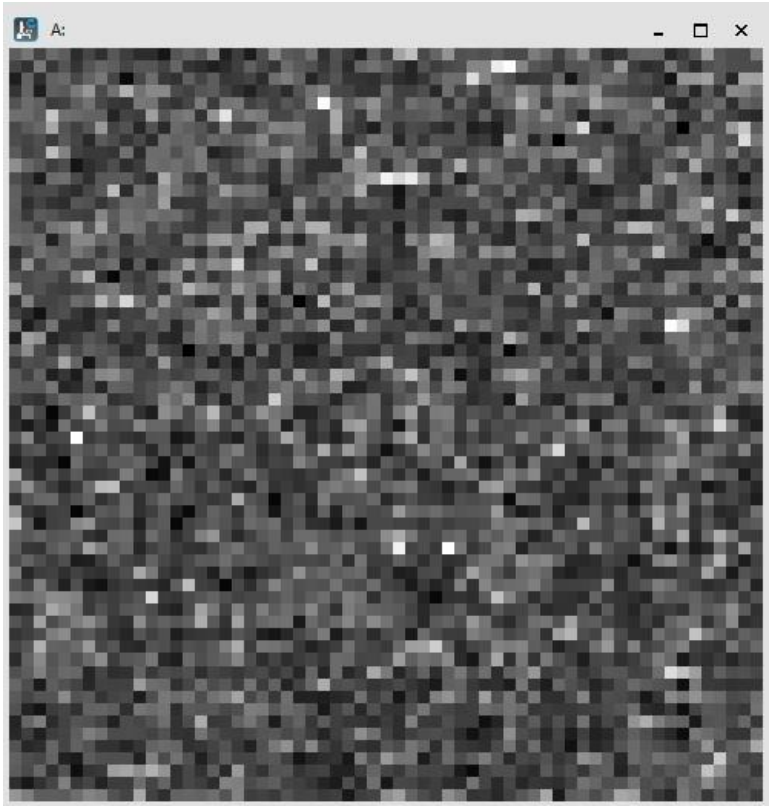
- also save the result(s) as tiff format



Transformation into a spectrum image

Requirements: Python-integrated GMS 3, Numpy, Scipy, Tifffile, Tkinter

1) load 4D-STEM data in GMS 3 (dimensions = (sx, sy, dsx, dsy))
2) crop 4D-STEM data for a ROI (optional)
3) **transform 4D-STEM data into a spectrum image**



[4D-STEM]_SI_transformation.py

uncheck
“Execute on Background Thread"