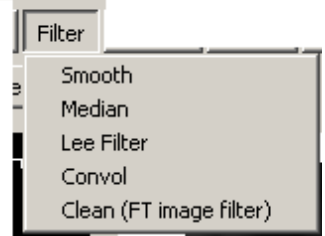


# Filtering images



**Smooth** - is an odd-number Savitsky-Golay filter

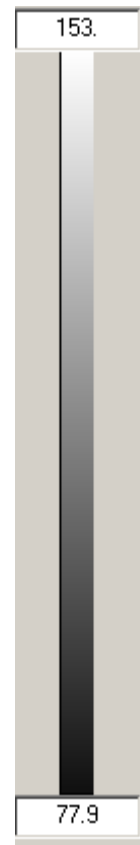
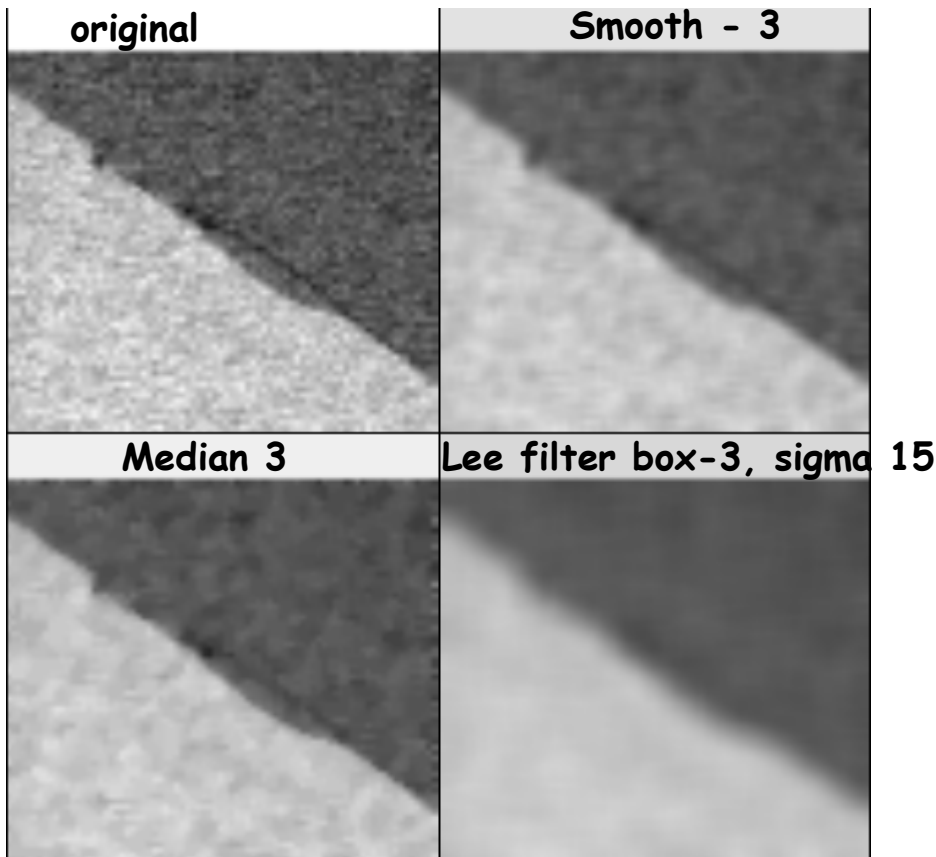
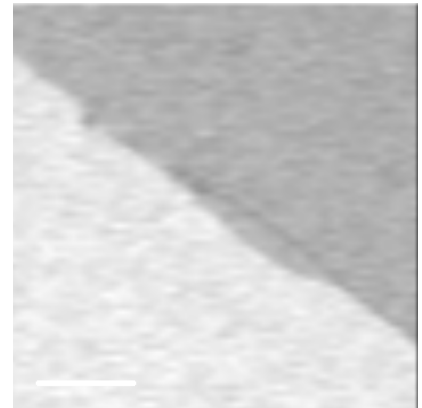
**Median** - similar to Smooth - less aggressive

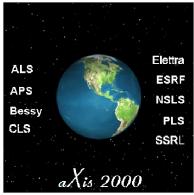
**Lee filter** - stronger smooth, 2 parameters  
 Filter - box (range in pixels);  
 Sigma - strength (high is less strong)

**Convol**

**FT-filter** - see next slide

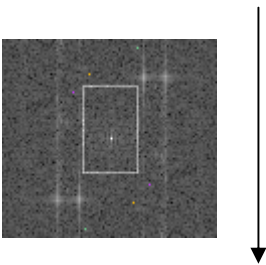
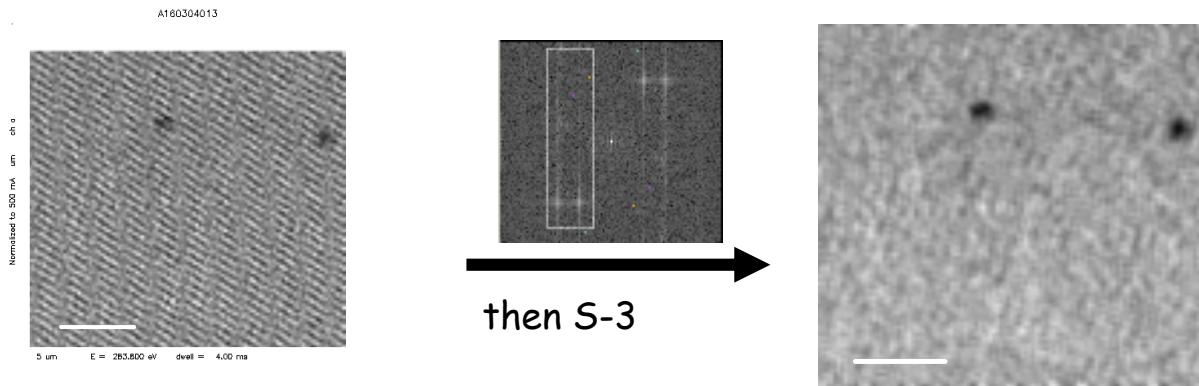
Convol  
 F-low 0.01  
 F-high 0.05  
 Order 3



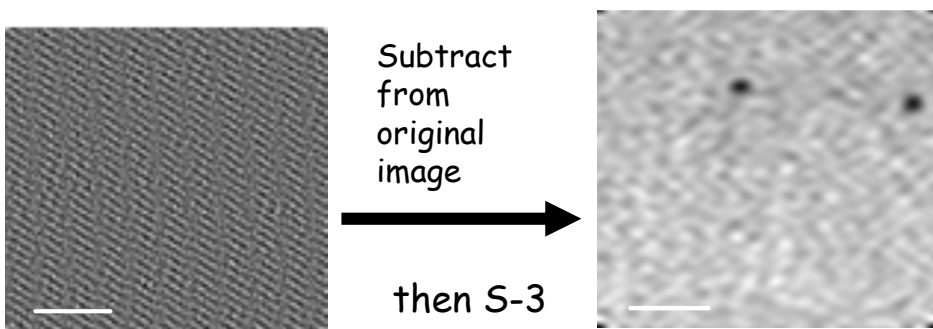


# FT filter

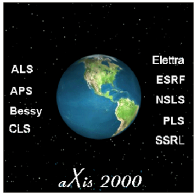
**Clean (FT image filter)** - 2d-FT filter. The FT is displayed on a 1:1 pixel format. Use the rubber-band style cursor (click, drag mouse, click a second time) to define the data in the complementary frequency domain to delete. The reverse transform of (all but the rejected data) is displayed in buffer 9. The centre of the FT image corresponds to 0 (dc), while positions farther from the centre correspond to higher frequency. Periodic (moiré) noise associated with aliasing (beating) of a systematic noise signal with STXM sampling can be cleanly removed by deleting strong (typically linear) signals in the FT. Noise with horizontal periodicity is vertical in the FT, and vice versa.



The **Clean (2d-FT Image filter)** can be used with subtraction of the filtered result to perform high-pass or selective band-pass filtering. For example

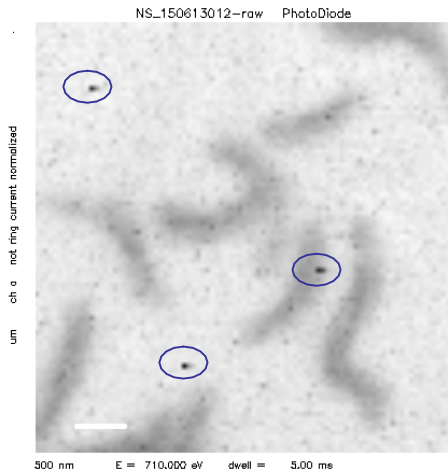


Difference gives smoothed signal; can isolate frequency components in x,y differently to remove systematic noise associated with scanning etc.

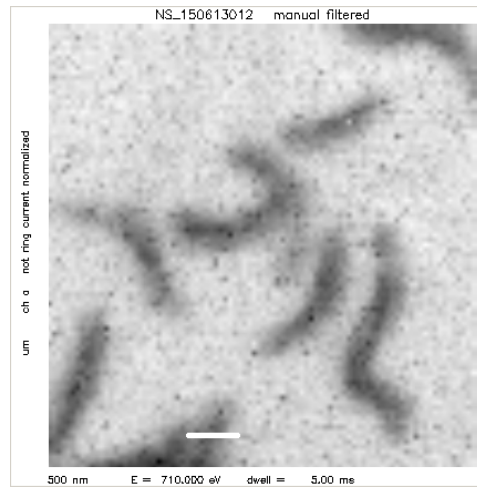


# Alternate filtering – salt & pepper

Salt-and-pepper type noise can be removed with `Emulator_filter` (only in post 7.1 versions of IDL)

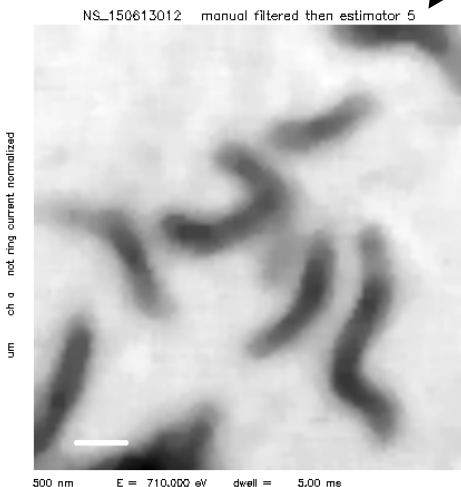
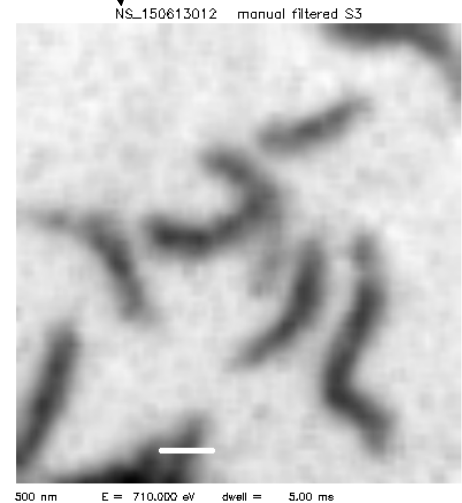
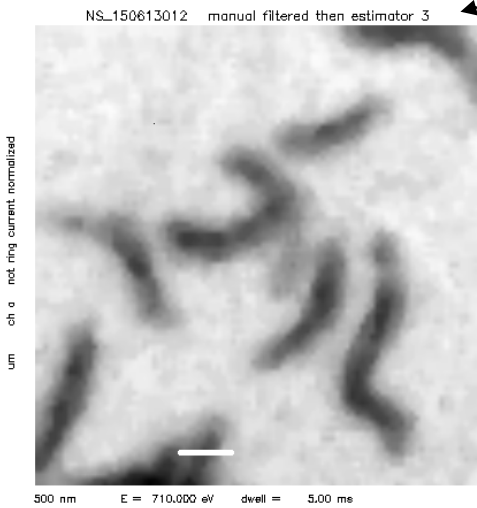


Manually  
modify 3  
spots



Smooth~3

`Emulator_filter(img,3)`



In this case, the result is less good than a simple 3-point smooth, in my opinion