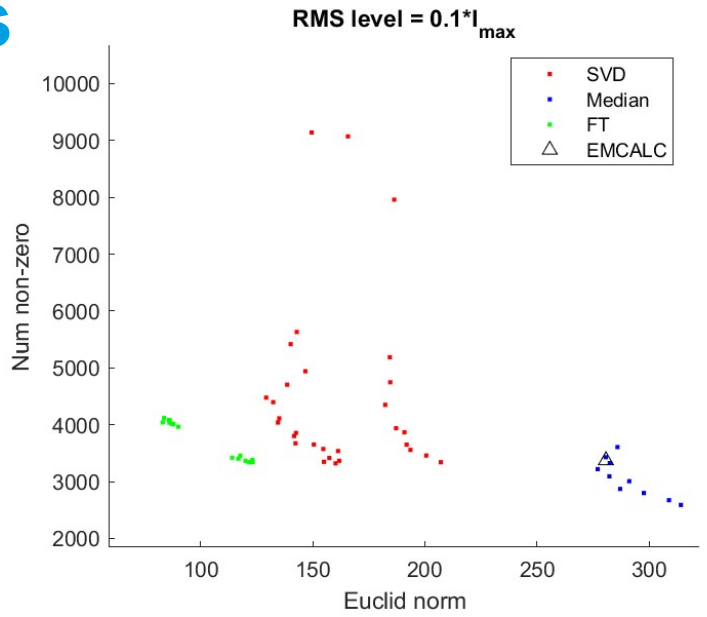
Noise processing using Singular Value Decomposition

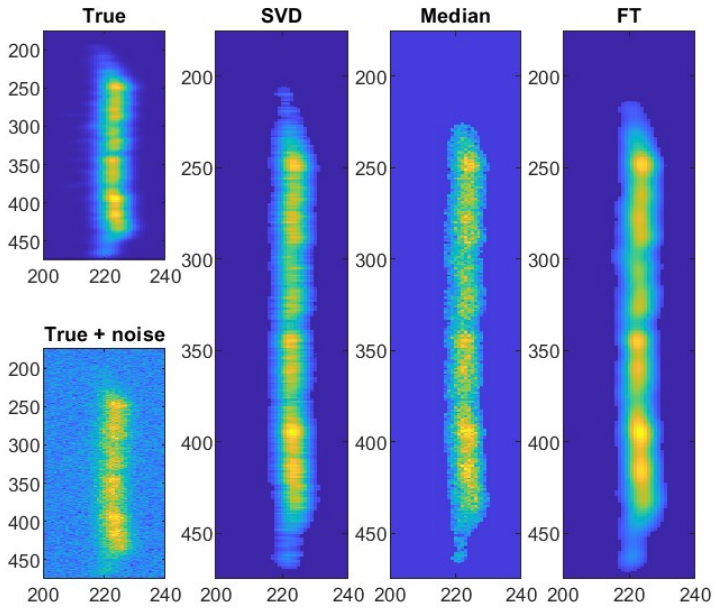
PPS 13.10.2022

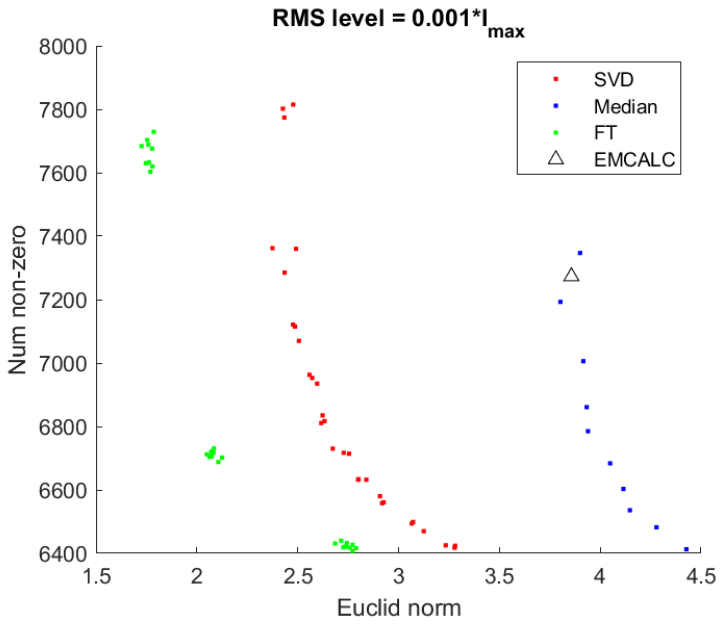
Chris Richard

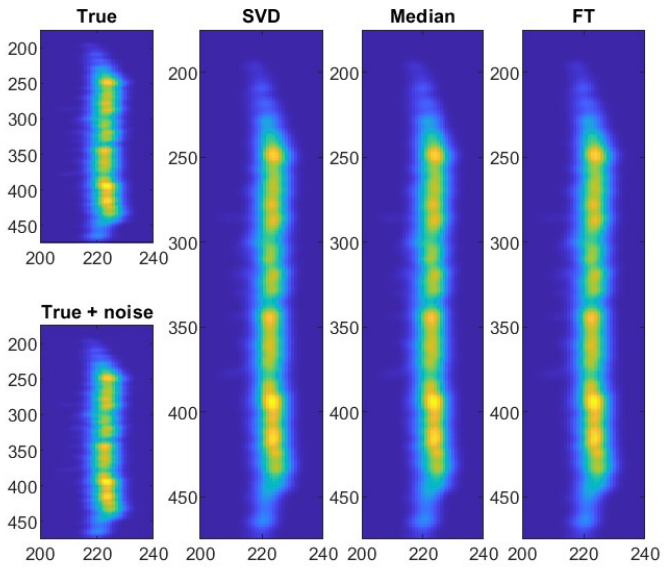
SVD filter vs other filters

- Optimize filters by minimizing euclidean norm (i.e. minimize error from true distribution) and minimize the number of non-zero pixels (i.e. ensure the noise is removed)
- Median-mask filter
  - Fast and simple
  - But worst results
- Fourier transform fitler
  - Gives best results
  - But slowest, most complex, and challenging to make flexible
- SVD fitler
  - Middle ground in performance, complexity, and speed
  - Easy to develop a robust filter



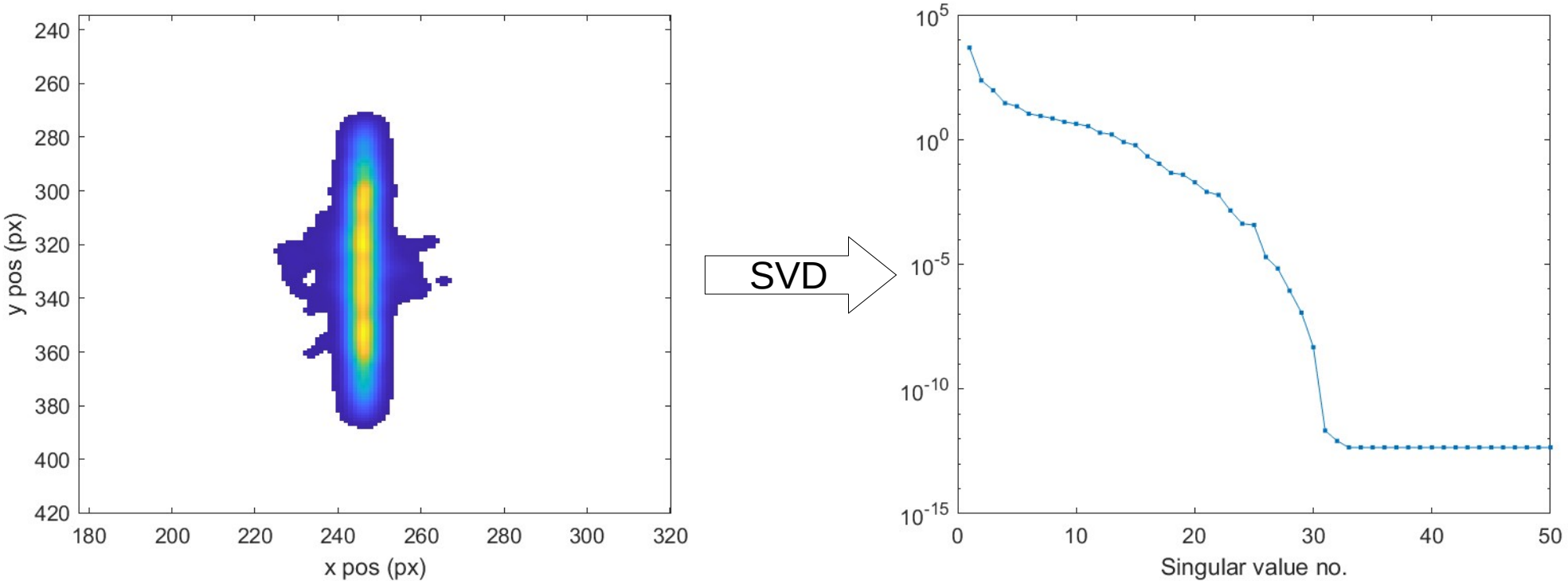






Singular value decomposition

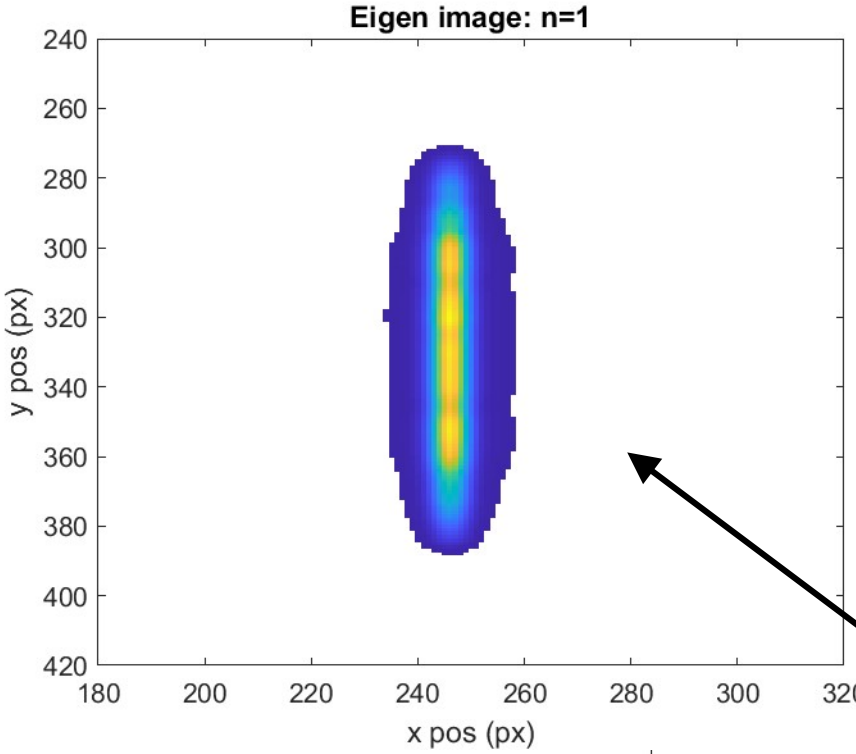
- SVD uniquely diagonalizes a mxn matrices
  - $I = USV^T = \sum_k \mathbf{u}_k^{\mathrm{col}} \otimes \mathbf{v}_k^{\mathrm{col}} s_k$ where U is a unitary mxm matrix, S is a nxm diagonal matrix containing the (non-negative) singular values (in descending order), V is a nxn unitary matrix
  - Interpreted as generalization of eigen decomposition to rectangular matrices
- Outer product of U and V creates set of eigen images. i.e. original image can be reconstructed by summing the eigen images weighted by the corresponding singular values

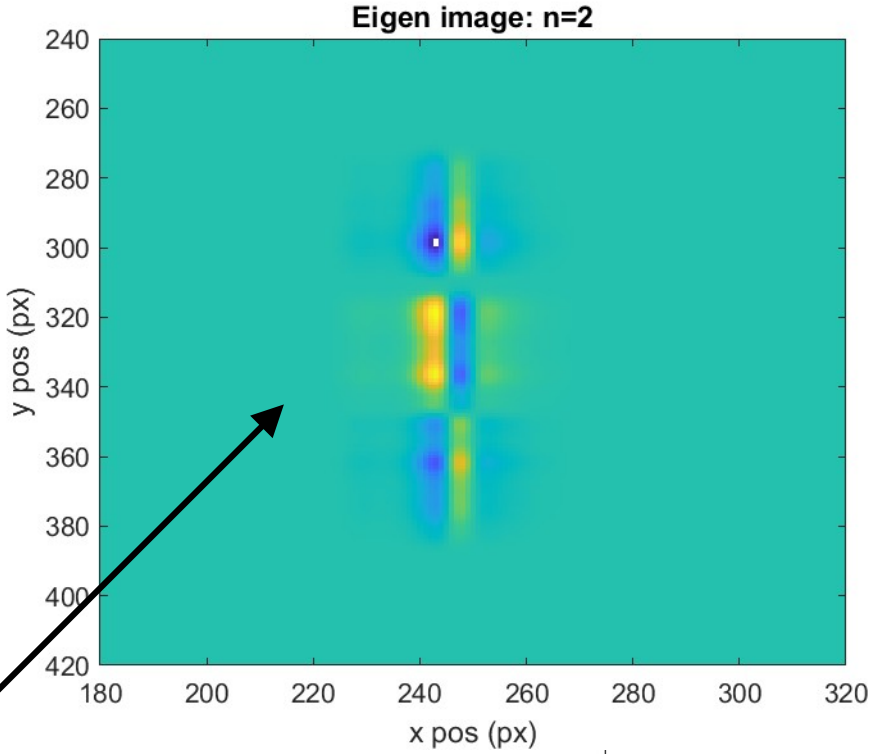


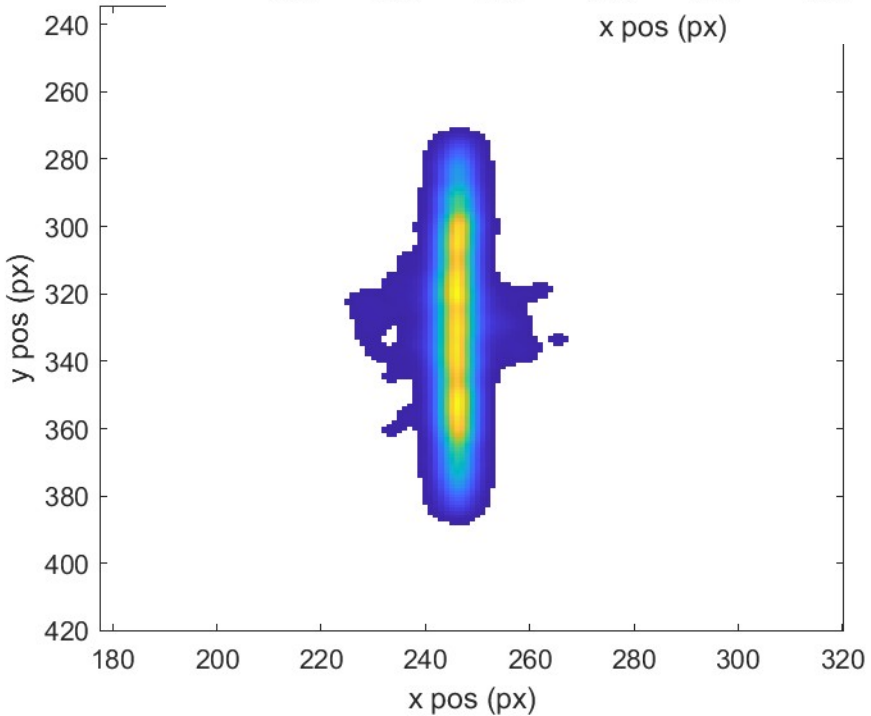
Singular

- SVD unique
  - $I = US$
    negative
  - Interprete
- Outer produ
  the eigen im

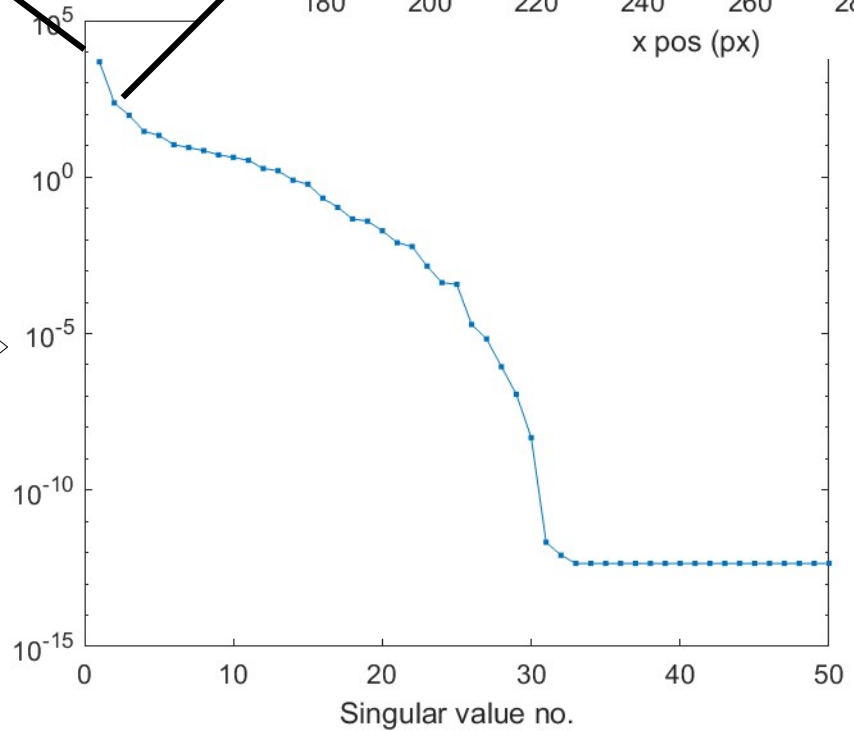
m matrix, S
nitary mat
angular ma
original
values





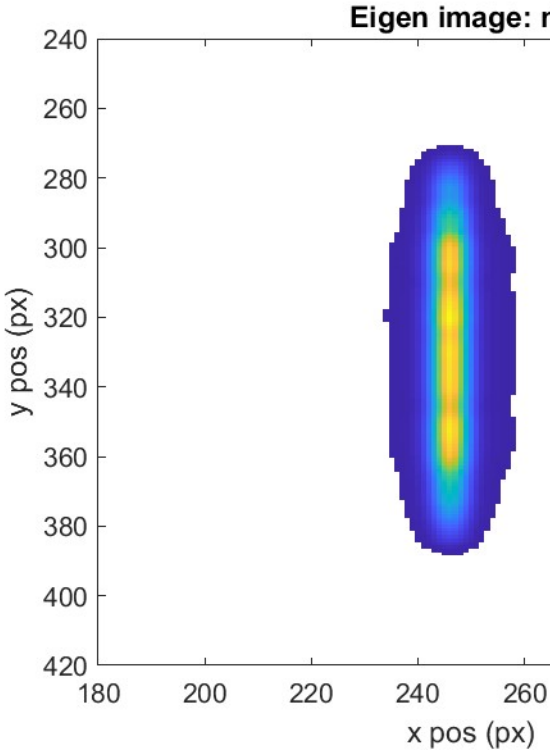


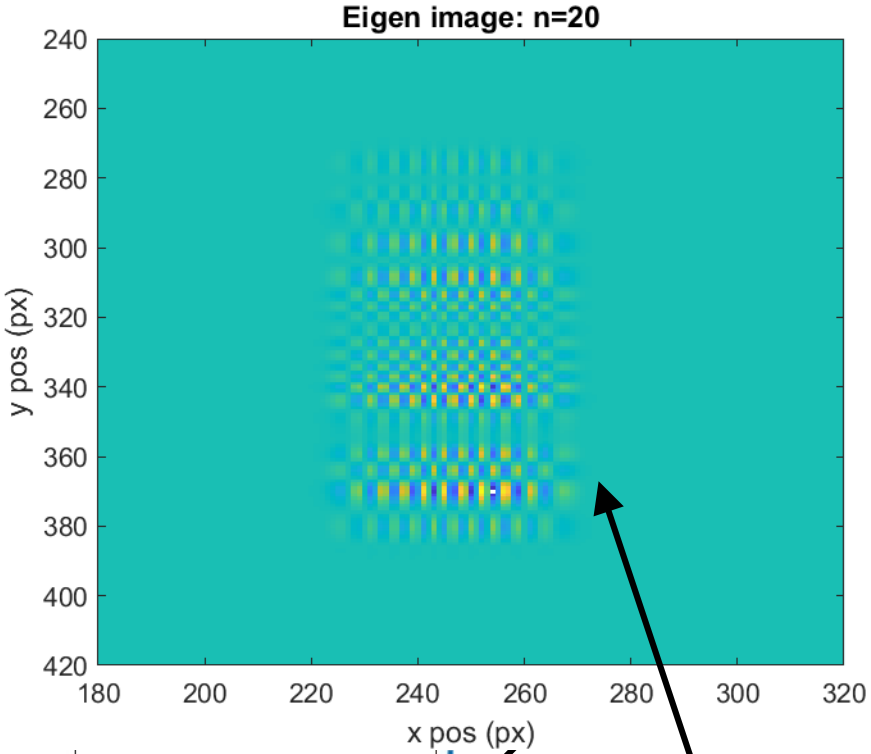
SVD

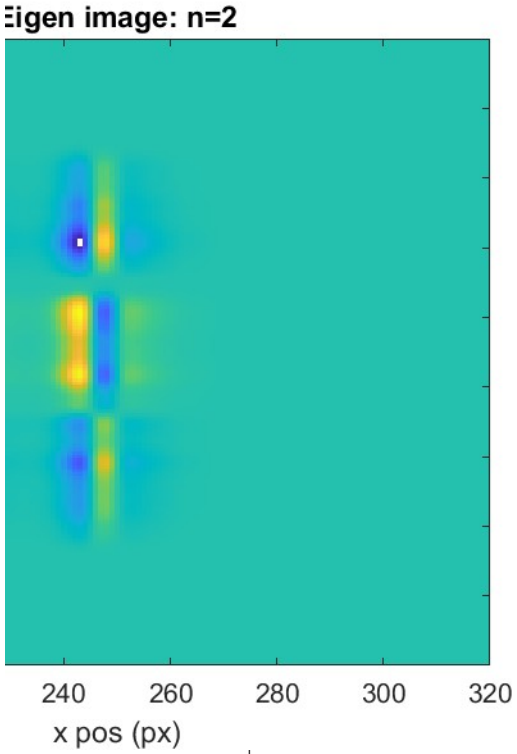


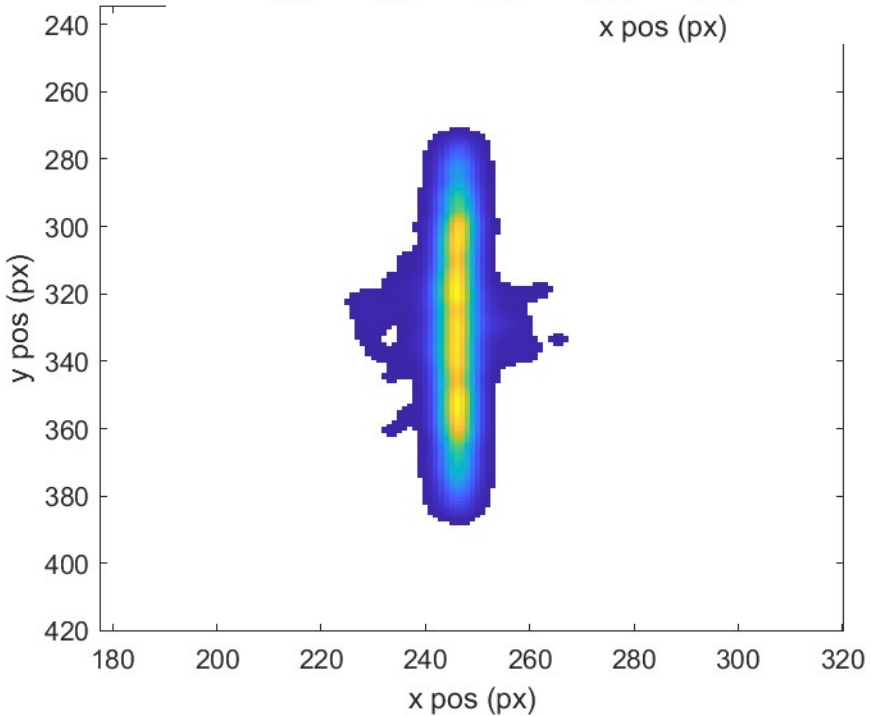
Singular

- SVD unique
  - $I = US$
    negative
  - Interprete
- Outer produ
  the eigen im

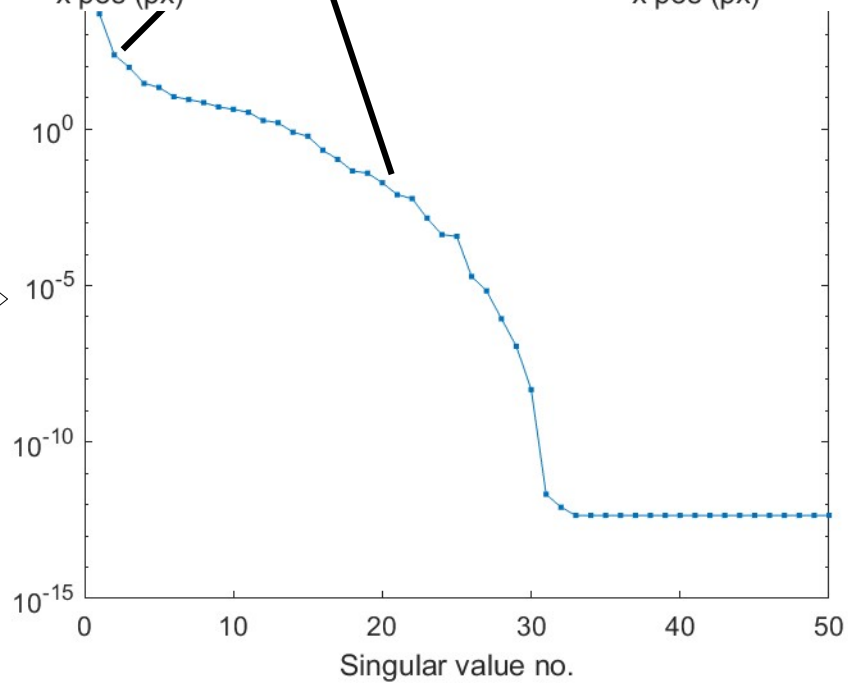






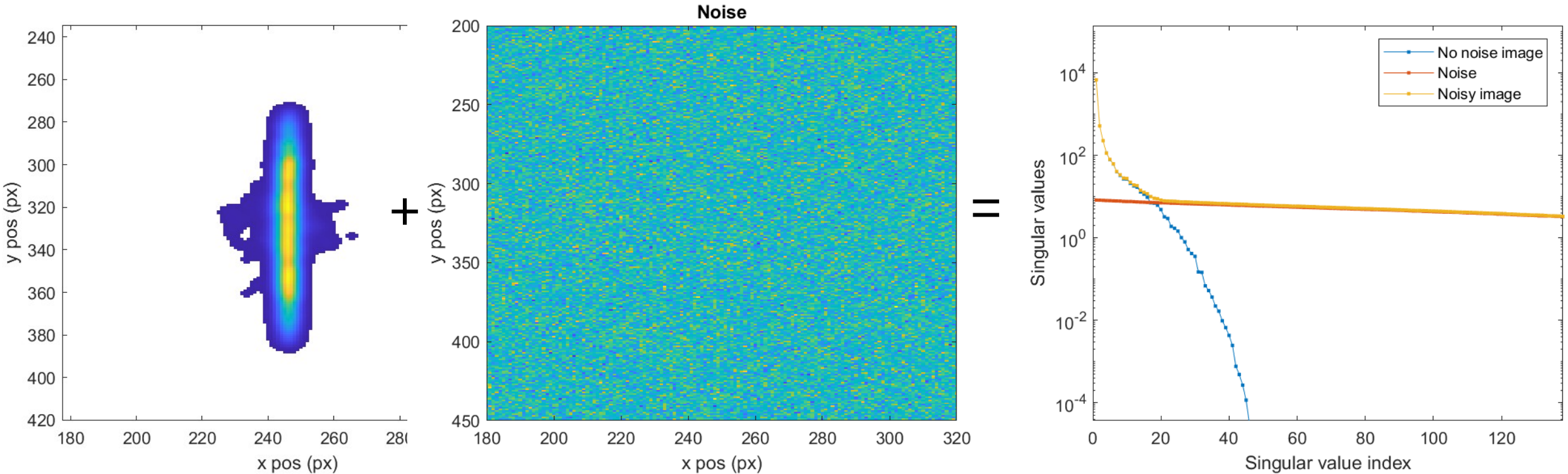


SVD



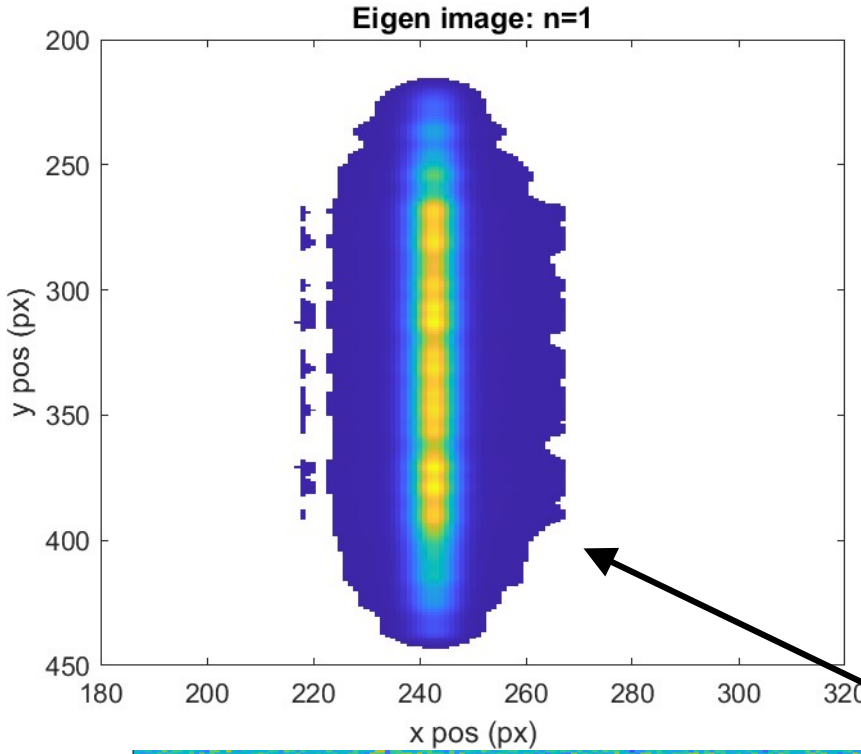
Noise filtering using SVD

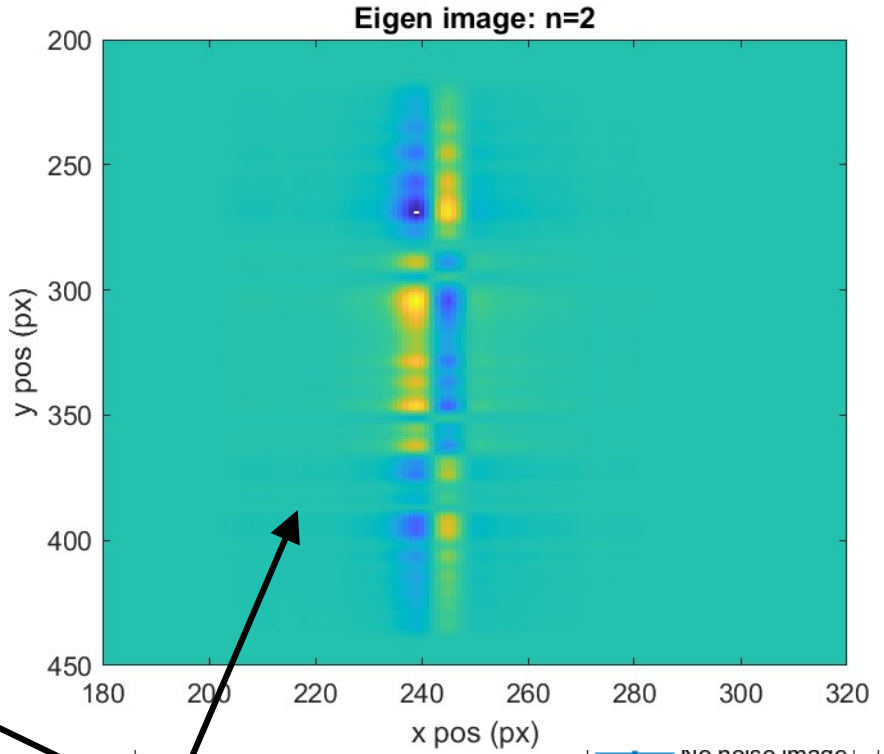
- When noise is added to an image, the singular values become the sum of the clean image singular values and the noise singular values
  - Note: this isn't mathematically rigorous because the SVD of the noisy image gives a different set of eigen images
- Cleaning method: take SVD of noisy image, compare singular values to those of noise image, set to zero singular values corresponding to noise eigen images, reconstruct image
  - Simplifies problem to a 1D filter. Significantly easier than filtering 2D images

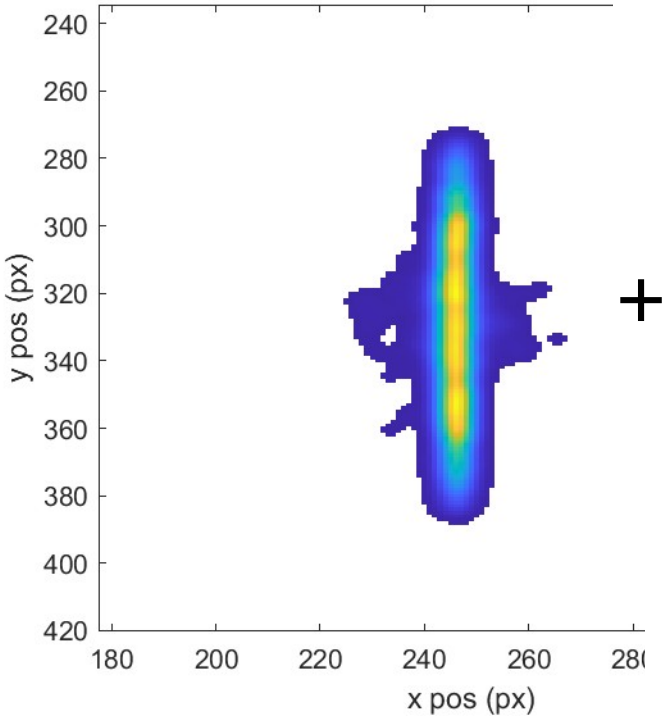


Noise filtering

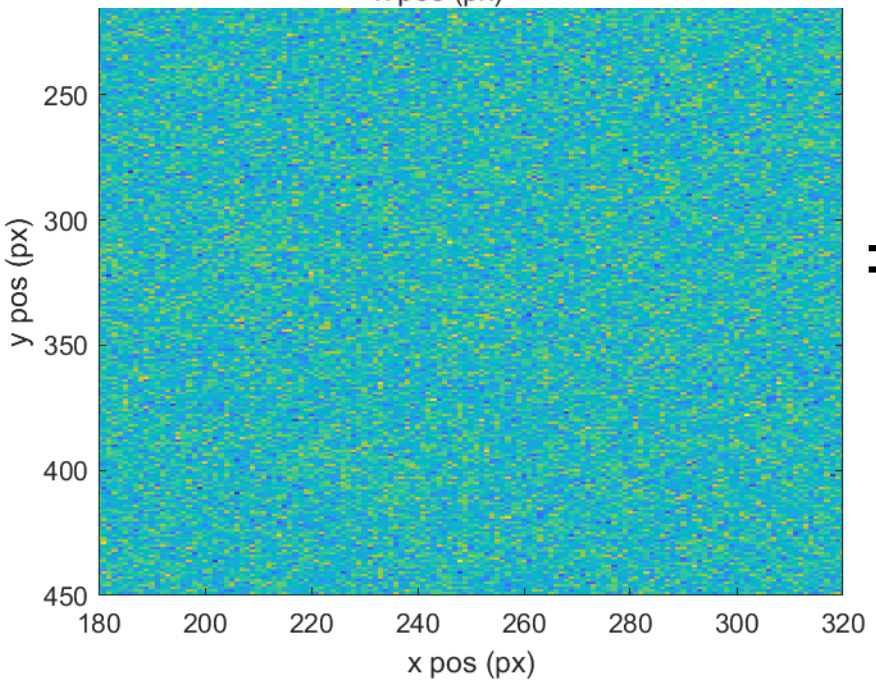
- When noise is added
  and the noise singular
  - Note: this isn't math
- Cleaning method: take
  singular values corres
  - Simplifies problem t



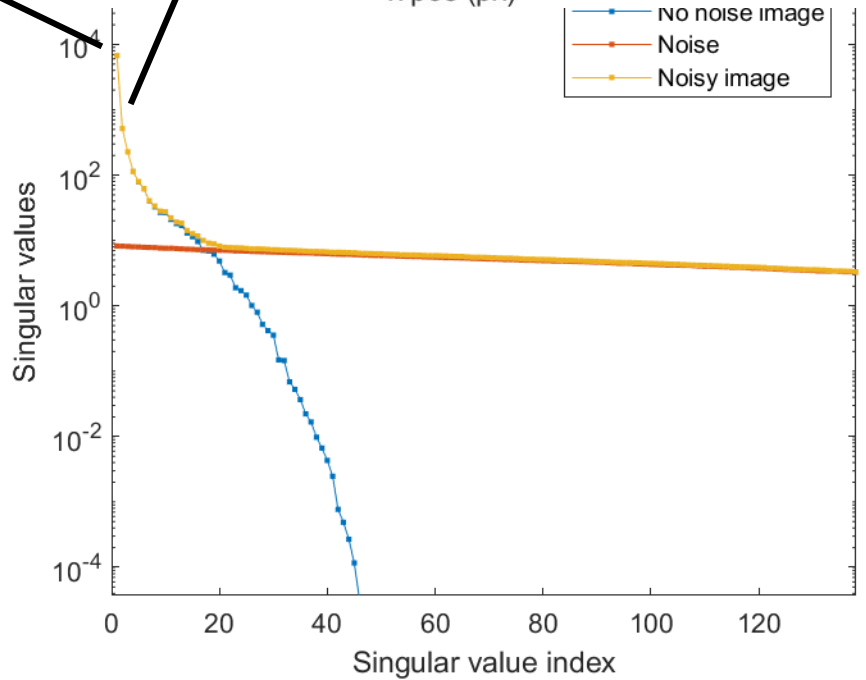




+

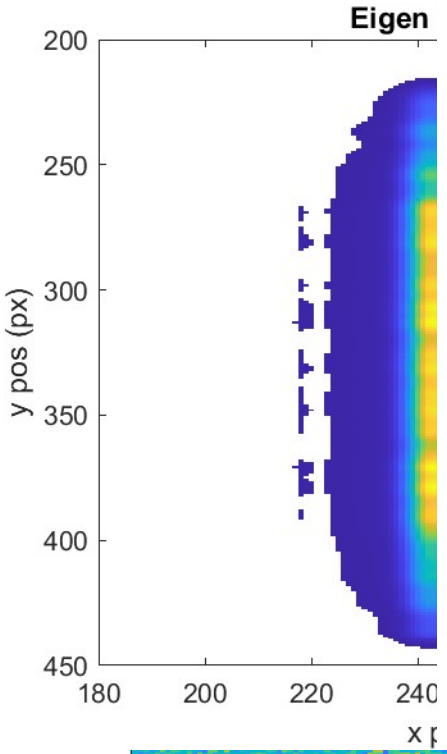


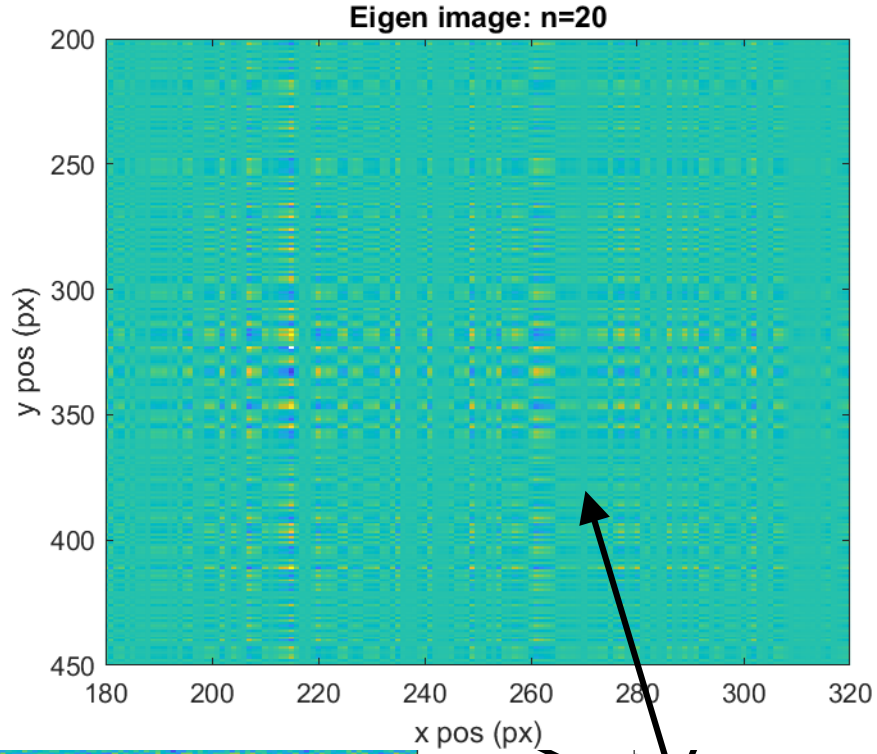
=

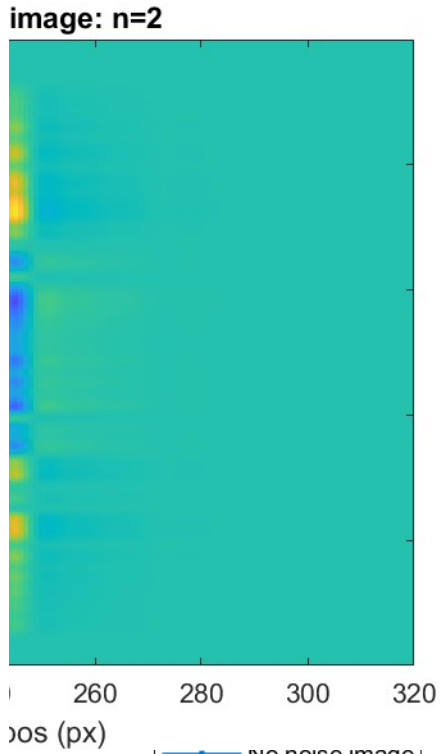


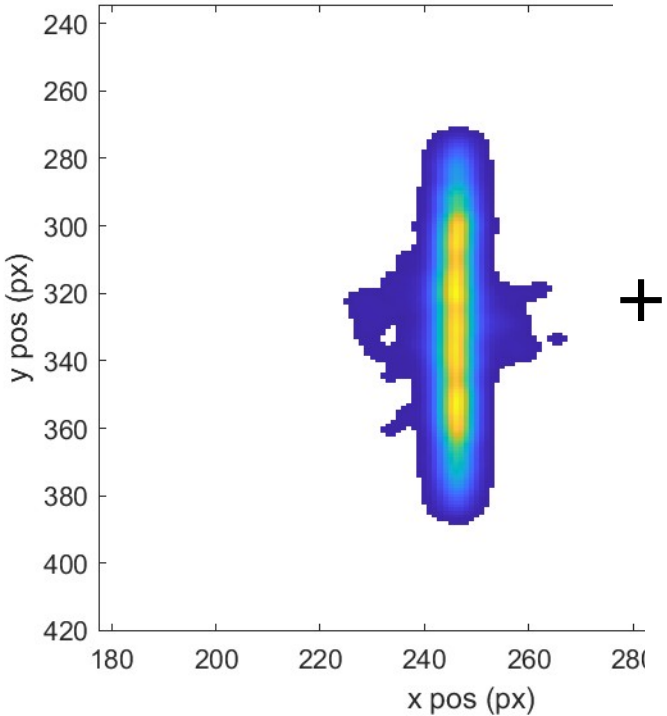
Noise filtering

- When noise is added
  and the noise singular
  - Note: this isn't math
- Cleaning method: take
  singular values corres
  - Simplifies problem t

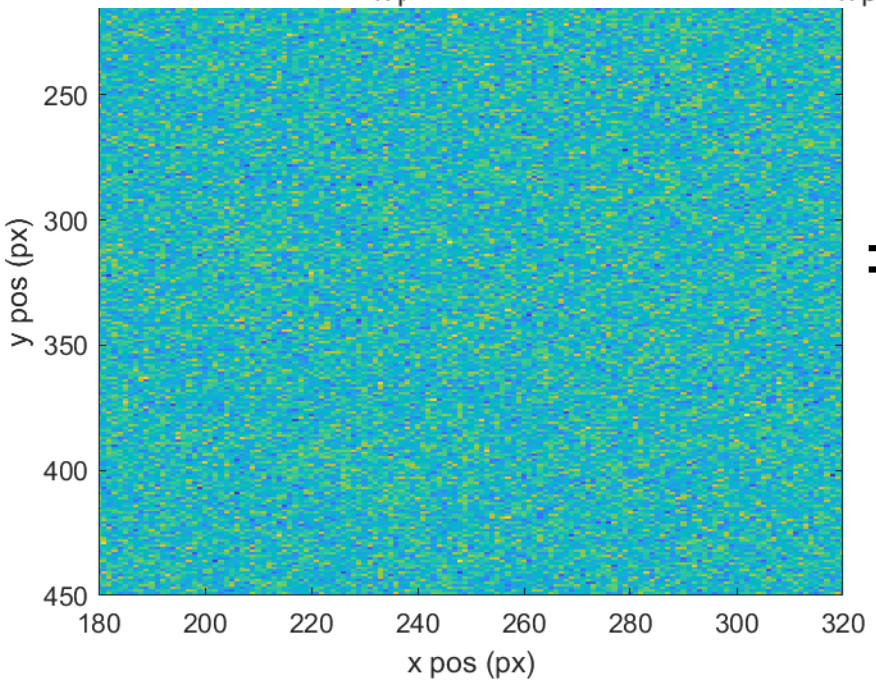




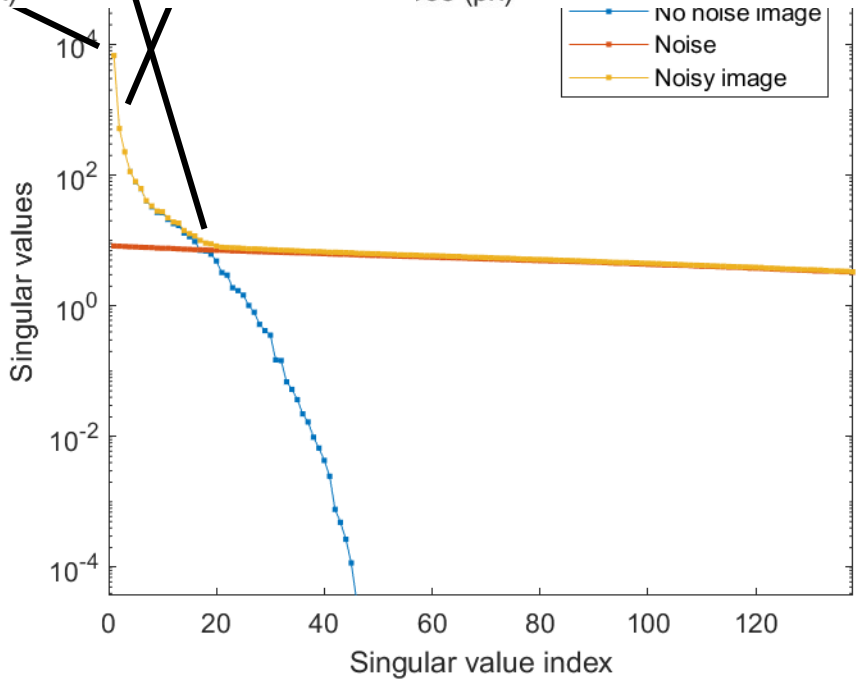




+

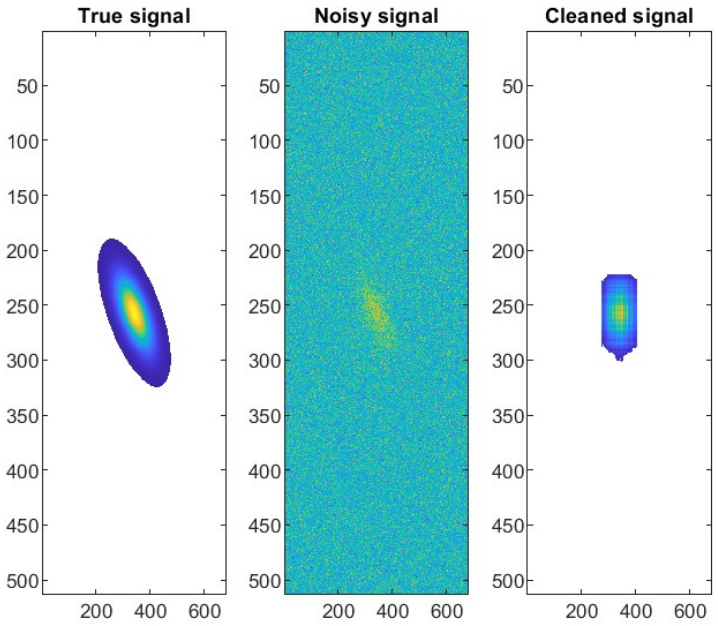


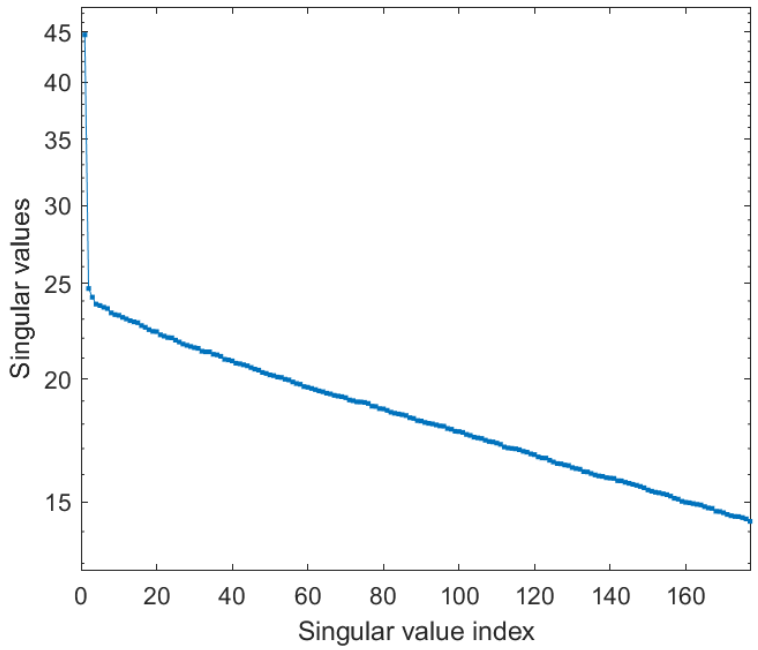
=

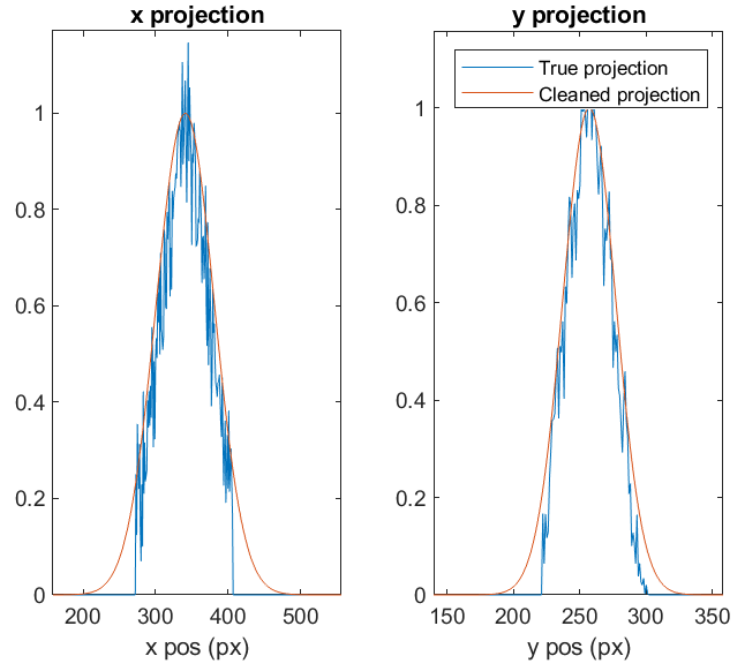


Low SNR behavior of SVD filter

- When the signal is barely above the noise floor, only one eigen image is not noise
- Results in a blocky reconstructed image
  - Eigen image is the outer product of two vectors → it is impossible to have x-y correlations
- Projections are still reasonable even if 2D distribution is not
  - So still useful (e.g. for low intensity EMSY beamlets)







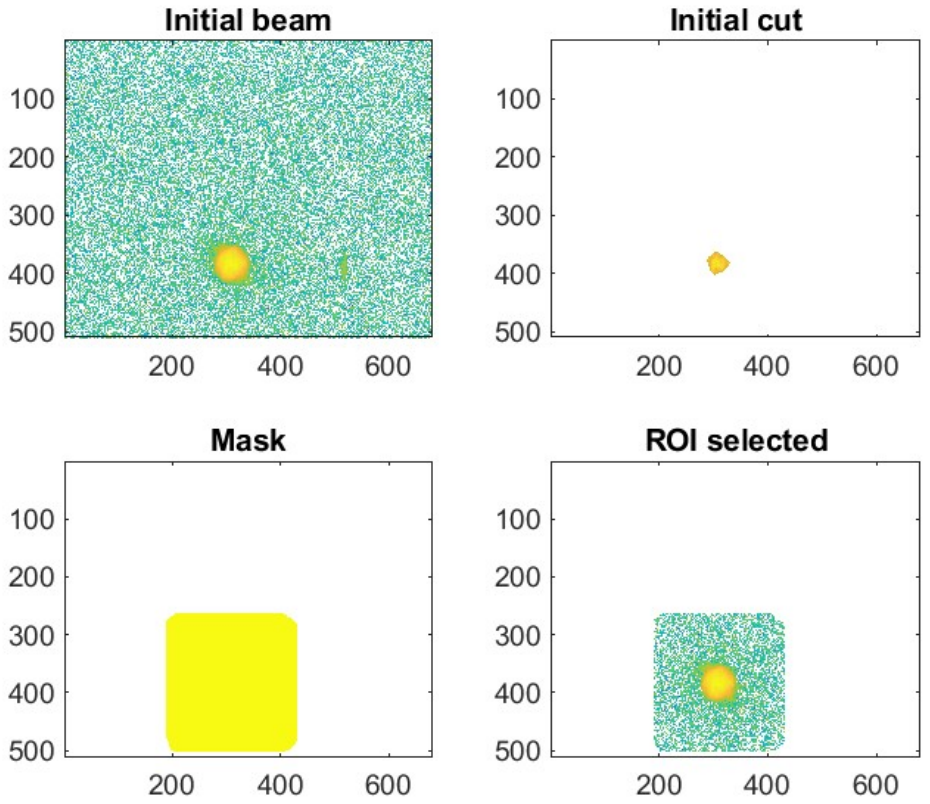
Overview of SVD_filter_noise_fit.m

Step 0: Inputs

- Input beam images as mxnxp array for p mxn images
- Input background images as array with same format as beam images.
  - Optional and number of background images doesn’t have to be equal to the number of beam images
  - Only used to subtract the mean pixel-by-pixel background
- MOI flag – true if a MOI has already been applied to the images

Step 1: Region of interest (ROI) filter

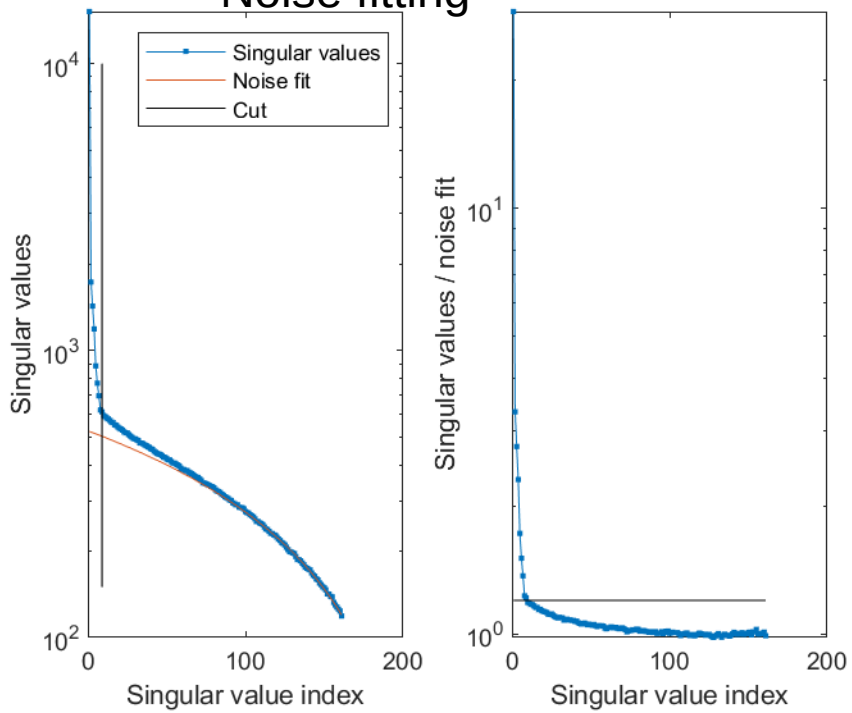
- Selects rough region around beam signal to remove signals far from the beam
  - Estimate peak intensity by the median of the 30 highest intensity pixels
    - Helps avoid salt and pepper noise
  - Make binary mask. 1 if pixel intensity > 0.5*peak intensity estimate, 0 else
    - Large cut to only select large signals
  - Apply 5x5 median filter to mask to remove salt and pepper noise
  - Dilate mask to increase selected area. This is purposefully large to ensure tails



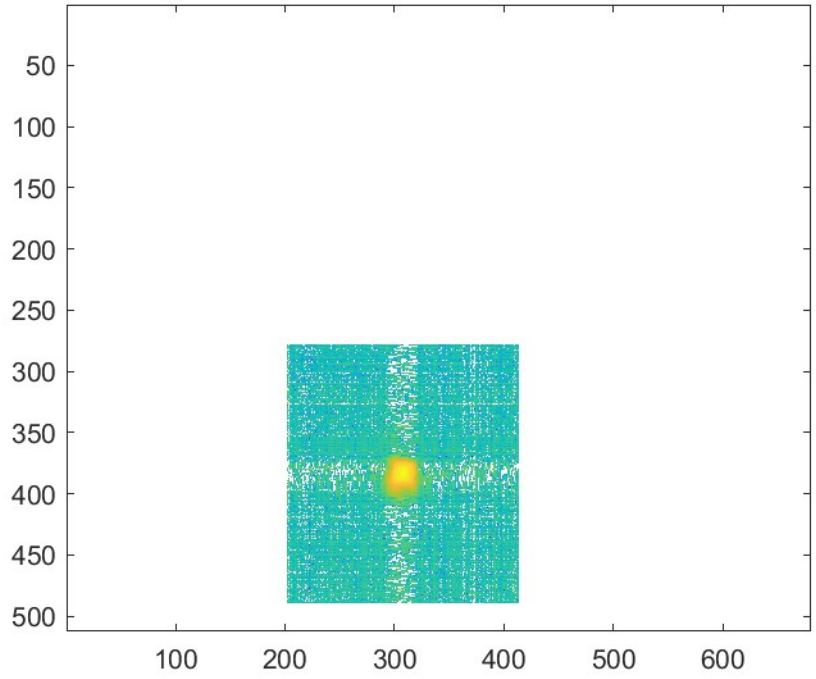
Step 2: SVD filter

- This is the noise removal sequence
  - Take SVD of beam image and noise image
  - Define ratio = beam singular values / noise singular values
  - Set to 0 all beam singular values corresponding to ratios < 1.2.
    - If no ratios are about the threshold then set the entire image to 0
  - Reconstruct the image using the modified beam singular values

Noise fitting

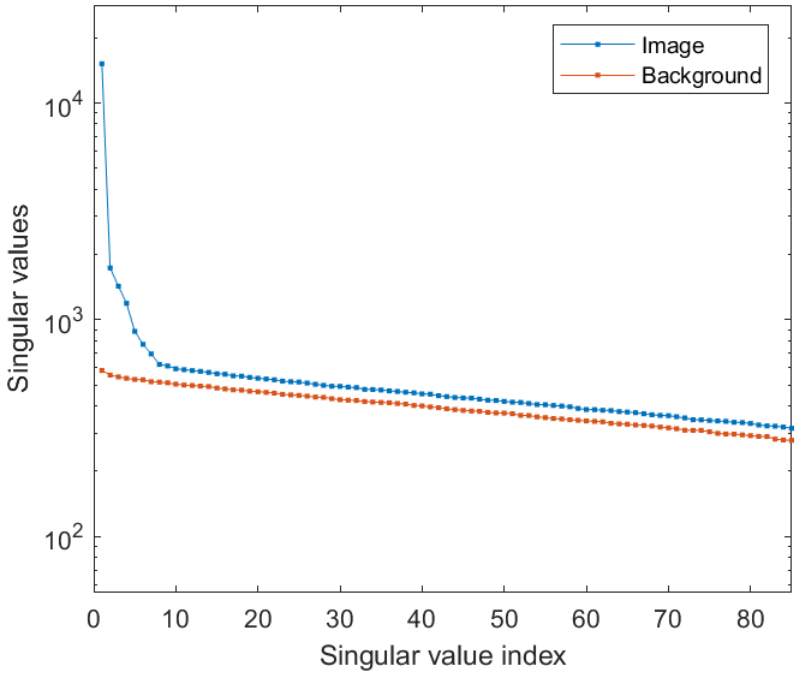


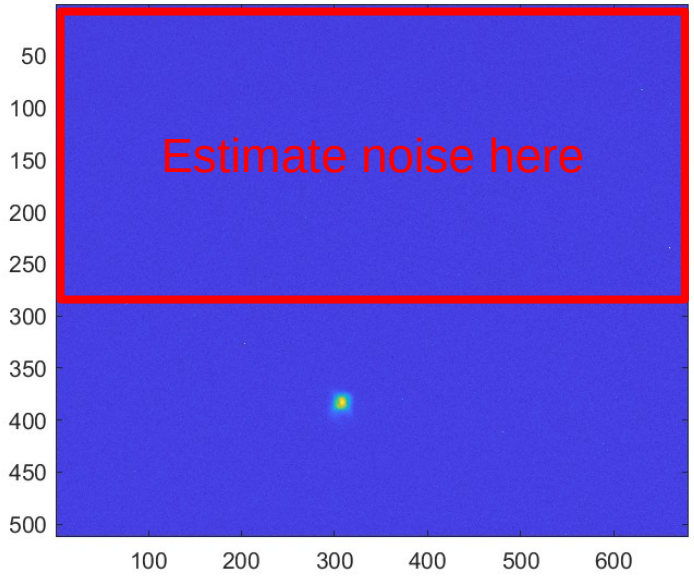
Reconstructed image



Aside: Noise level discrepancy

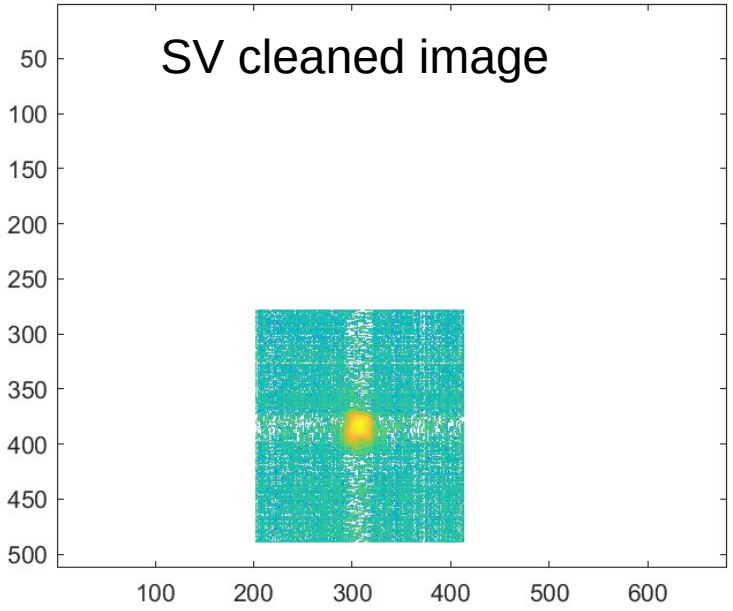
- Ideally, the filter would not use fitting the estimate the noise level, but use the singular values of the background image
- But, the singular values of the background images are ~10% than the beam image ones
- I verified this discrepancy by measuring the rms noise of beam image in region without beam and comparing the the rms noise in background images. The rms noise levels differ by ~10%
- Cannot rely on the background images to be an accurate representation of the noise!
  - May be caused by gamma ray showers from beam loss

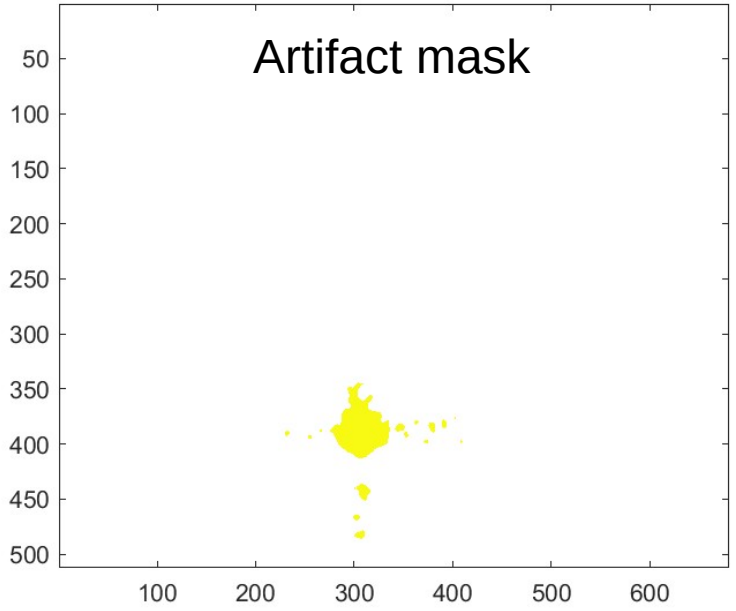


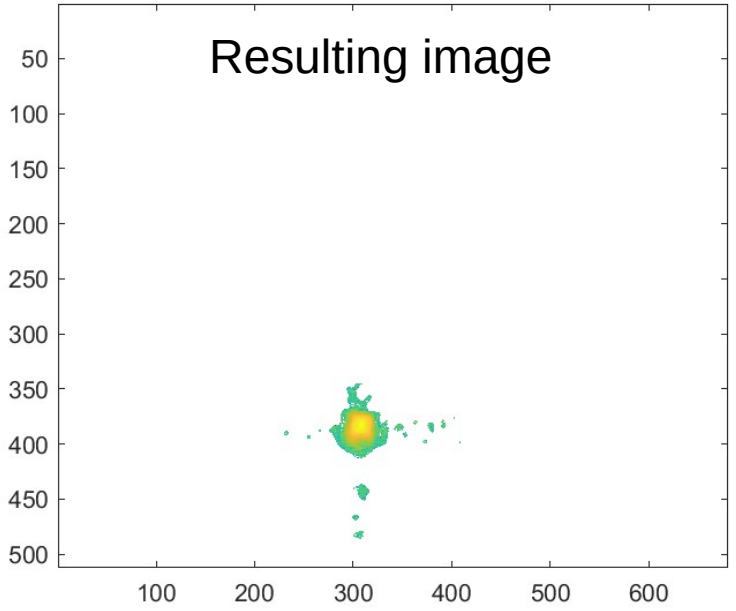


Step 3: Artifact cleaning

- Removes artifacts from reconstructed image.
  - Make binary mask. 1 if pixel intensity > A*peak intensity estimate, 0 else. A can be optimized
    - Define SNR = largest singular value / intercept of noise fit
    - Cut level: A = 0.20*SNR^-0.86
  - Apply median filter to mask multiple times to remove islands and fill gaps
    - Size of the median filter and number of applications can be optimized but results don't change significantly
      - Use 5x5 filter and apply 3 times for all images
  - Apply mask to image

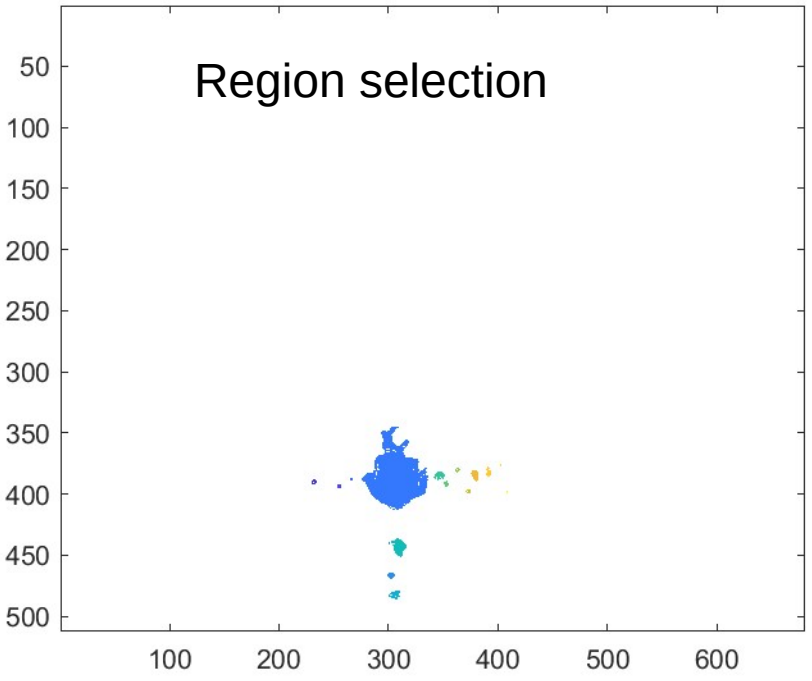


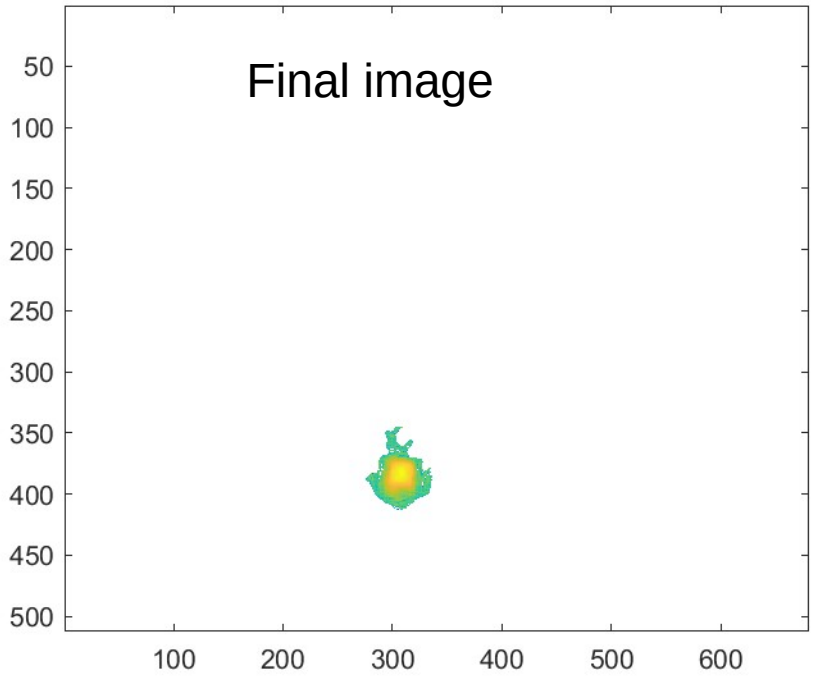




Step 4: Region select

- Identify all separated non-zero regions after cleaning. Keep only largest region
- This removes large artifacts missed by the previous step
- Also removes illuminated non-beam objects that were missed by the ROI filter(e.g. screen holders)

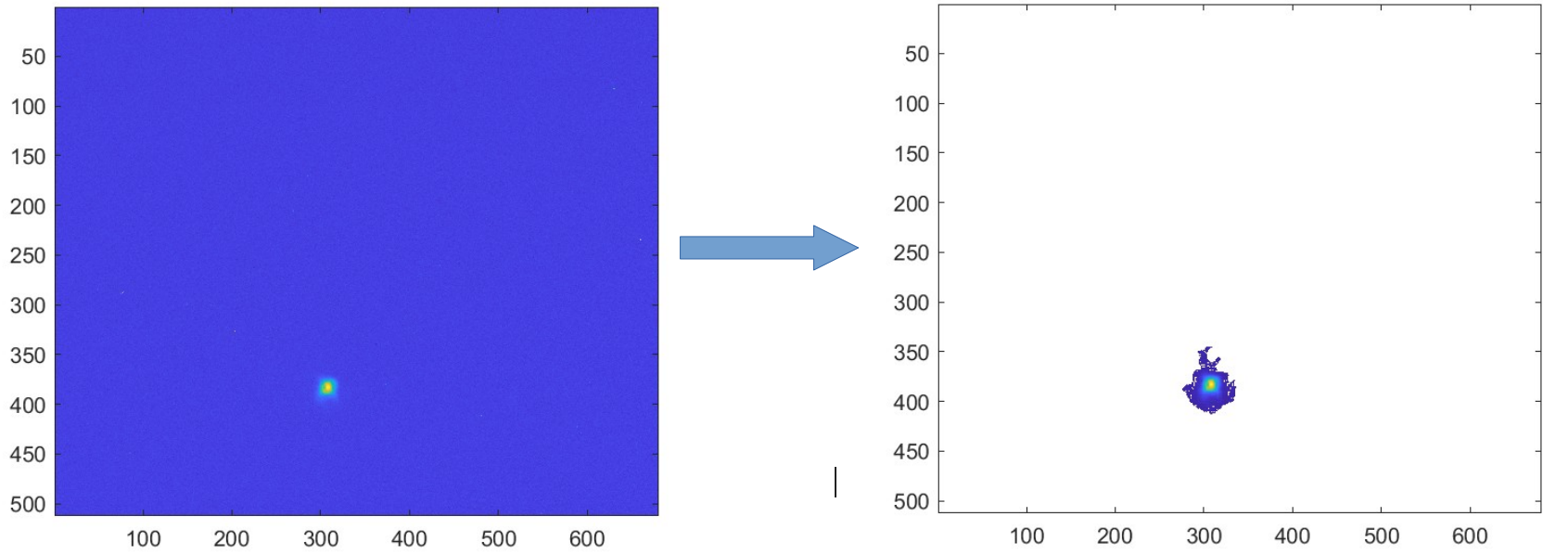


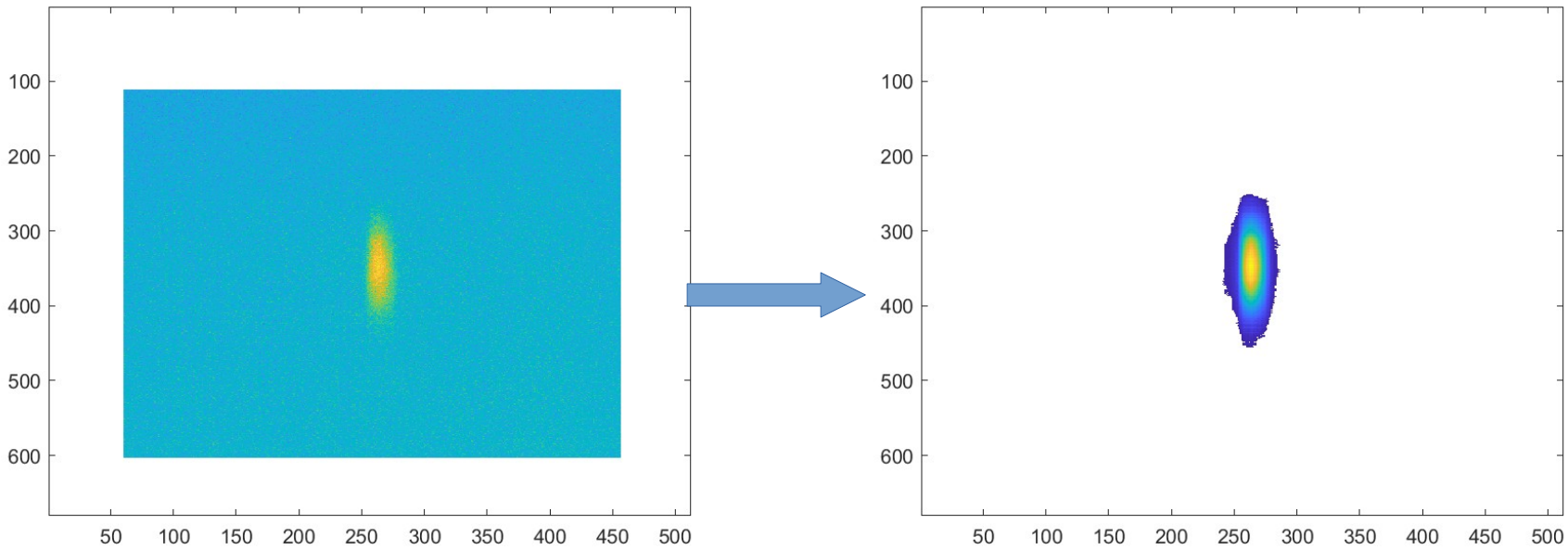


Filter results

Initial images

Cleaned images



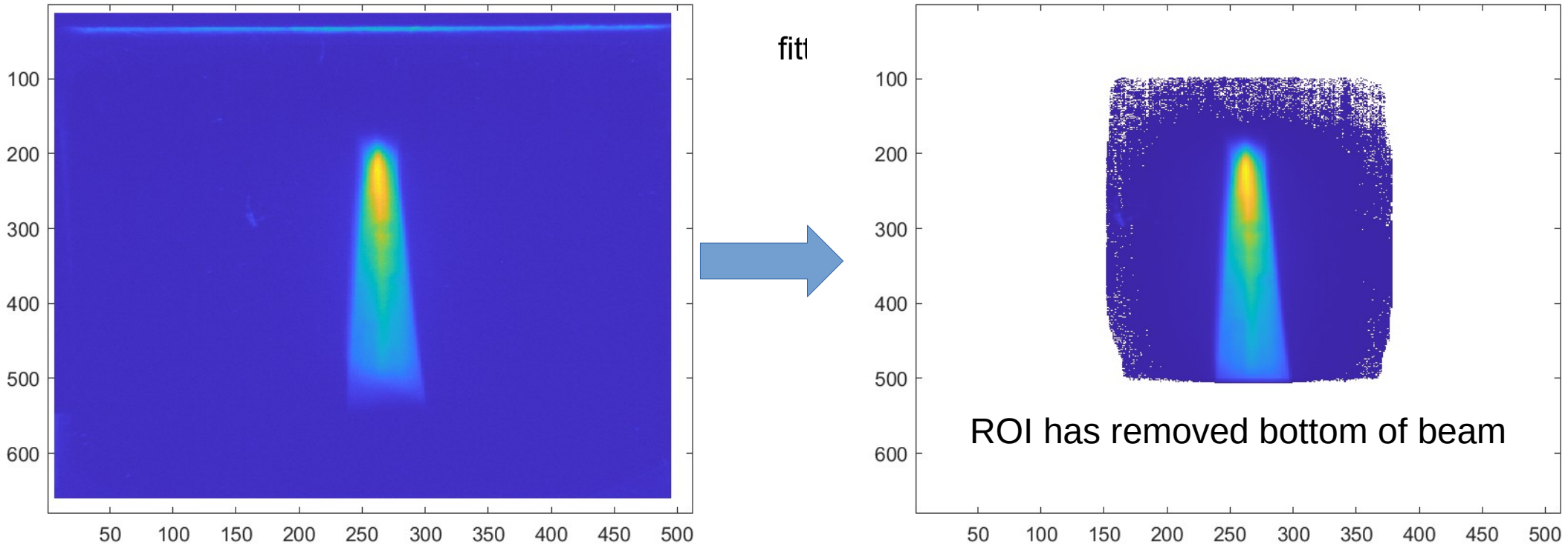


Known issues

- Region select could fail to select the beam if non-beam region is large enough
- Noise fitting is limited for narrow MOI. Number of singular values is defined by smallest dimension
  - If choosing MOI, it will work down to 20px wide regions, but best to keep dimensions >~200 px
- ROI filter was developed for EMSY beamlet images. For large beam images it can cut the edges
- Some images have significant halo after cleaning. Unclear if this is real or an artifact
  - It could be caused by improperly selecting the noise fitting region
  - If it is real, it it part of the beam or the screen near the beam glowing?

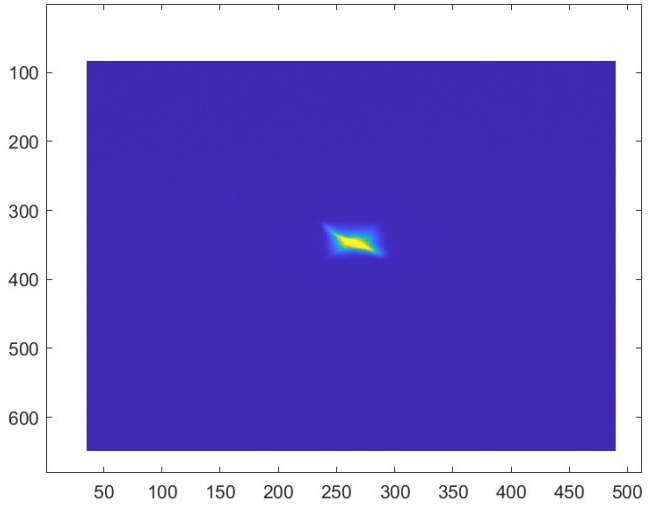
Known issues

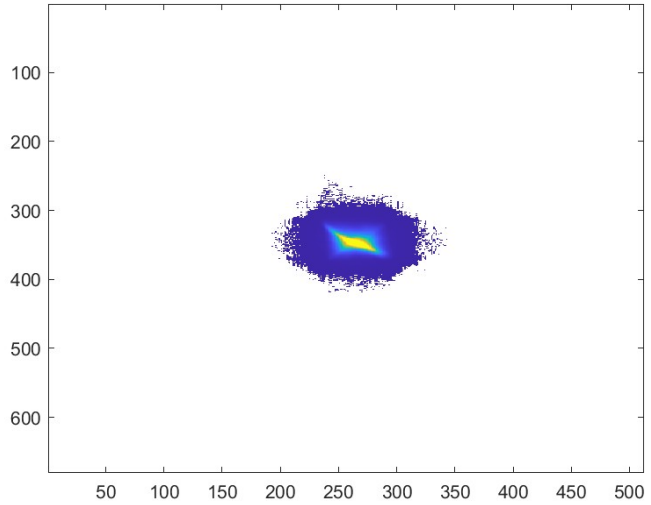
- Region select could fail to select the beam if non-beam region is large enough
- Noise fitting is limited for narrow MOI. Number of singular values is defined by smallest dimension
  - If choosing MOI, it will work down to 20px wide regions, but best to keep dimensions >~200 px
- ROI filter was developed for EMSY beamlet images. For large beam images it can cut the edges
- Ur
  fitt

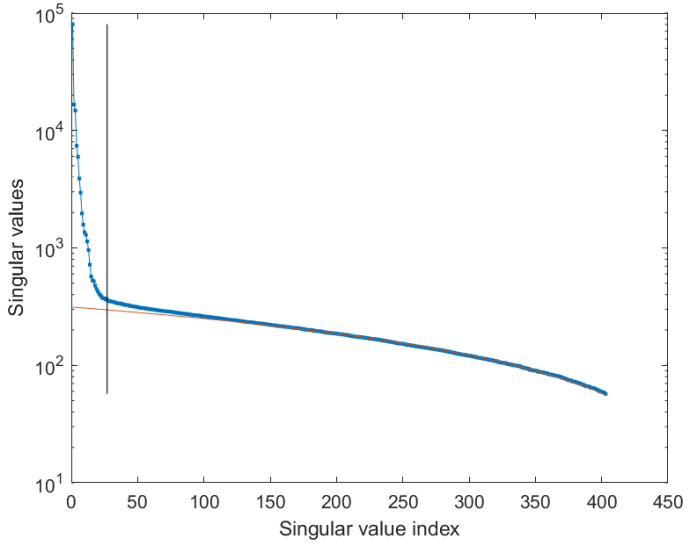


Known issues

- Region select could fail to select the beam if non-beam region is large enough
- Noise fitting is limited for narrow MOI. Number of singular values is defined by smallest dimension
  - If choosing MOI, it will work down to 20px wide regions, but best to keep dimensions >~200 px
- ROI filter was developed for EMSY beamlet images. For large beam images it can cut the edges
- Some images have significant halo after cleaning. Unclear if this is real or an artifact
  - It could be caused by improperly selecting the noise fitting region
  - If it is real, it it part of the beam or the screen glowing?







Summary

- A noise cleaning script using singular value decomposition has been made and optimized
  - Designed to be accurate and robust
- The current version has been tested on beam images, EMSY beamlet images, and TDS images
- Currently implemented in chronos.m
- If you use this script (or part of it), let me know any issues you have