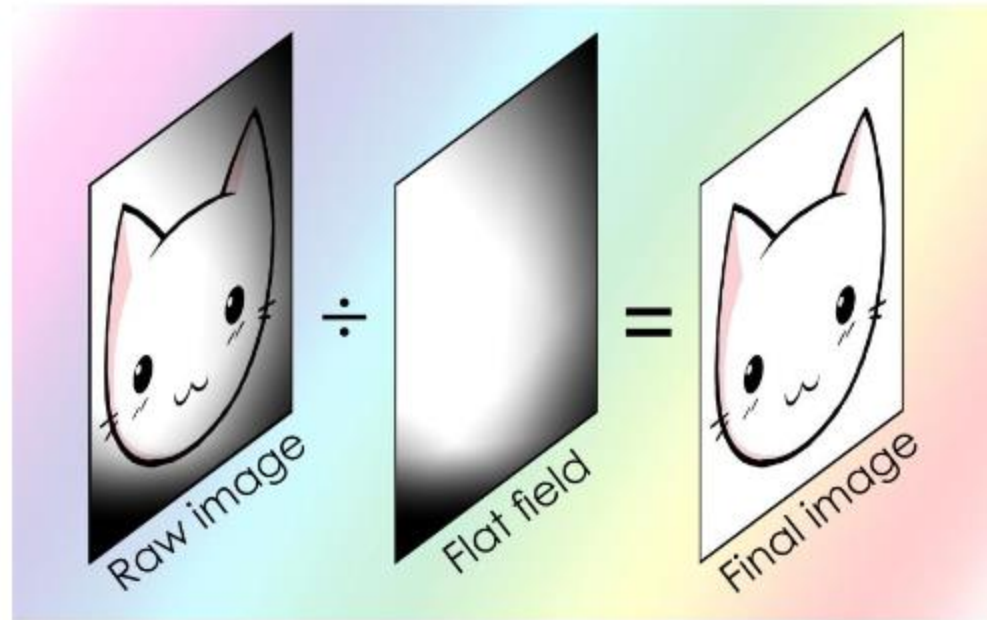
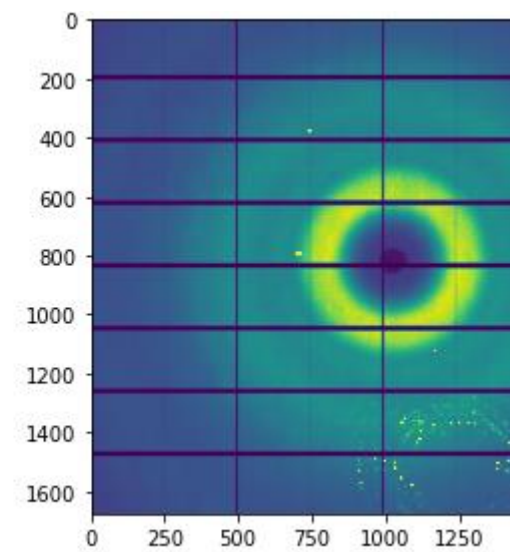
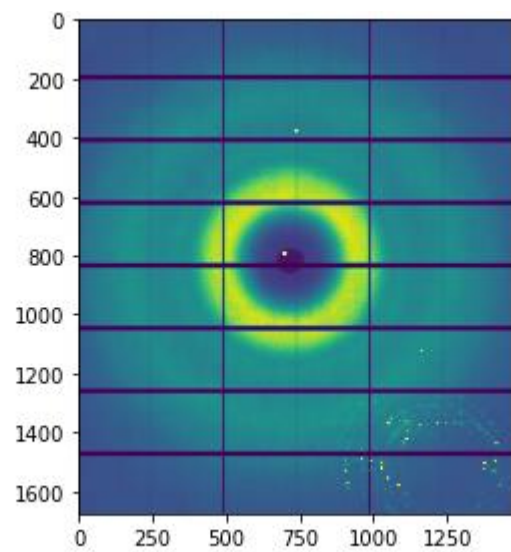
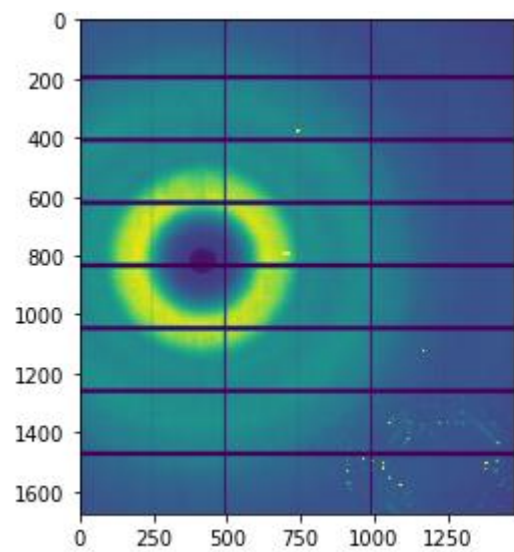
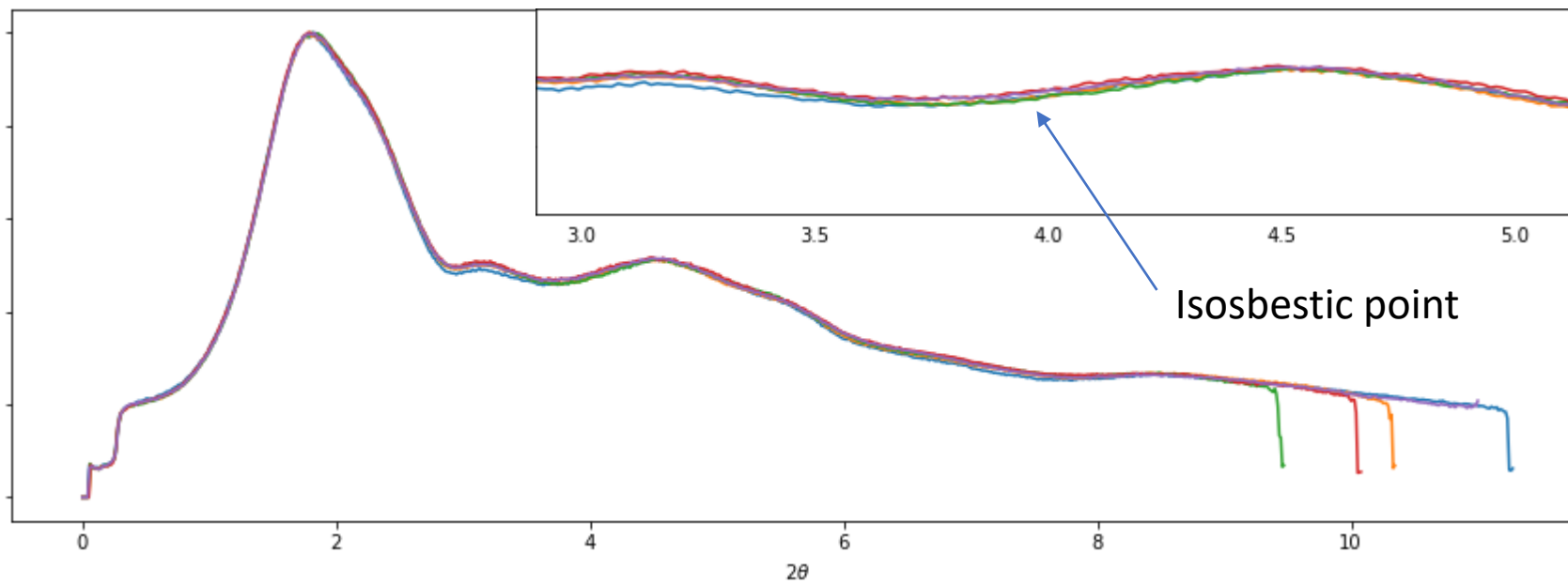
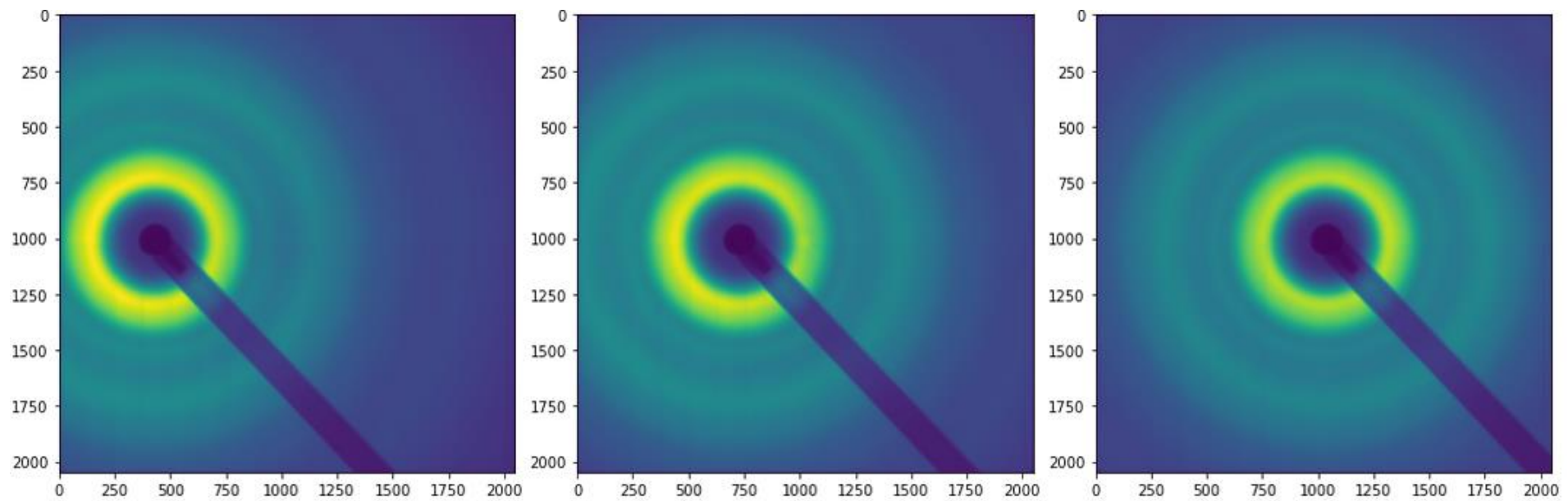
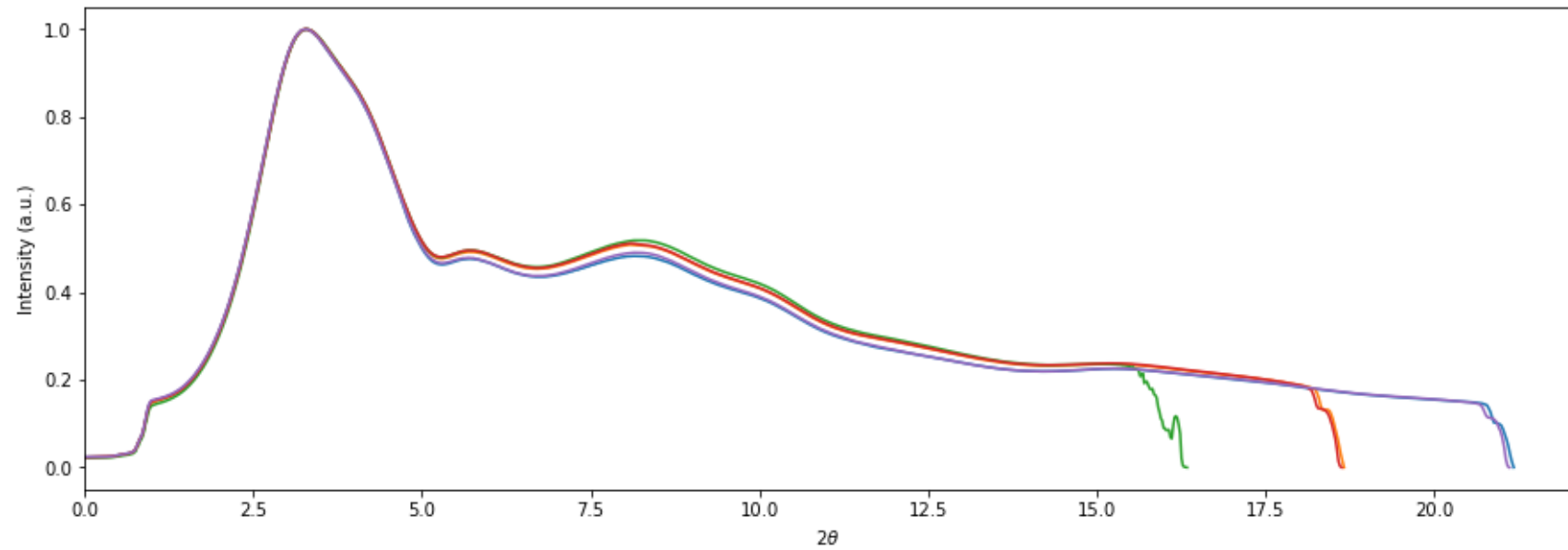


# X-ray detector gain correction without flat fields

James Weng







## Measurements

$$y = A x$$

$y \rightarrow$  measurement

$A \rightarrow$  measurement or sensing matrix

$x \rightarrow$  measured signal

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \longrightarrow \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 0.9 & 0 & 0 \\ 0 & 1.3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Measured Signal

1.1	1.3	3.1	1.8	1.2
0.9	1.5	0.8	1.3	1.2
1	1.3	1.3	1.2	1.3
0.8	1.3	1	1.5	1.5
0.6	1.2	1.6	1.3	1.2



Gain map

1	1.2	3	1.7	1.1
0.8	1.4	0.7	1.2	1.1
0.9	1.2	1.2	1.1	1.3
0.7	1.2	0.9	1.4	1.4
0.5	1.1	1.5	1.2	1.1



Signal being measured

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1



Dark Frame/error

0.1	0.1	0.1	0.1	0.1
0.1	0.1	0.1	0.1	0.1
0.1	0.1	0.1	0.1	0.1
0.1	0.1	0.1	0.1	0.1
0.1	0.1	0.1	0.1	0.1

Flat field

$$I_m(x, y) = G(x, y) \cdot S(x, y) + \epsilon$$

Measured Signal

$I_{00}$	$I_{10}$	$I_{20}$	$I_{30}$	$I_{40}$
$I_{01}$	$I_{11}$	$I_{21}$	$I_{31}$	$I_{41}$
$I_{02}$	$I_{12}$	$I_{22}$	$I_{32}$	$I_{42}$
$I_{03}$	$I_{13}$	$I_{23}$	$I_{33}$	$I_{43}$
$I_{04}$	$I_{14}$	$I_{24}$	$I_{34}$	$I_{44}$

=

Gain map

$G_{00}$	$G_{10}$	$G_{20}$	$G_{30}$	$G_{40}$
$G_{01}$	$G_{11}$	$G_{21}$	$G_{31}$	$G_{41}$
$G_{02}$	$G_{12}$	$G_{22}$	$G_{32}$	$G_{42}$
$G_{03}$	$G_{13}$	$G_{23}$	$G_{33}$	$G_{43}$
$G_{04}$	$G_{14}$	$G_{24}$	$G_{34}$	$G_{44}$

×

Signal being measured

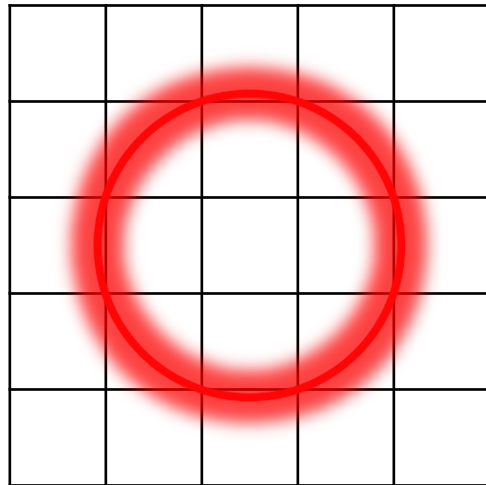
0	0.8	1	0.8	0
0.8	1.4	1	1.4	0.8
1	1	0	1	1
0.8	1.4	1	1.4	0.8
0	0.8	1	0.8	0

$$I_m(x, y) = G(x, y) \cdot S(x, y) + \epsilon$$

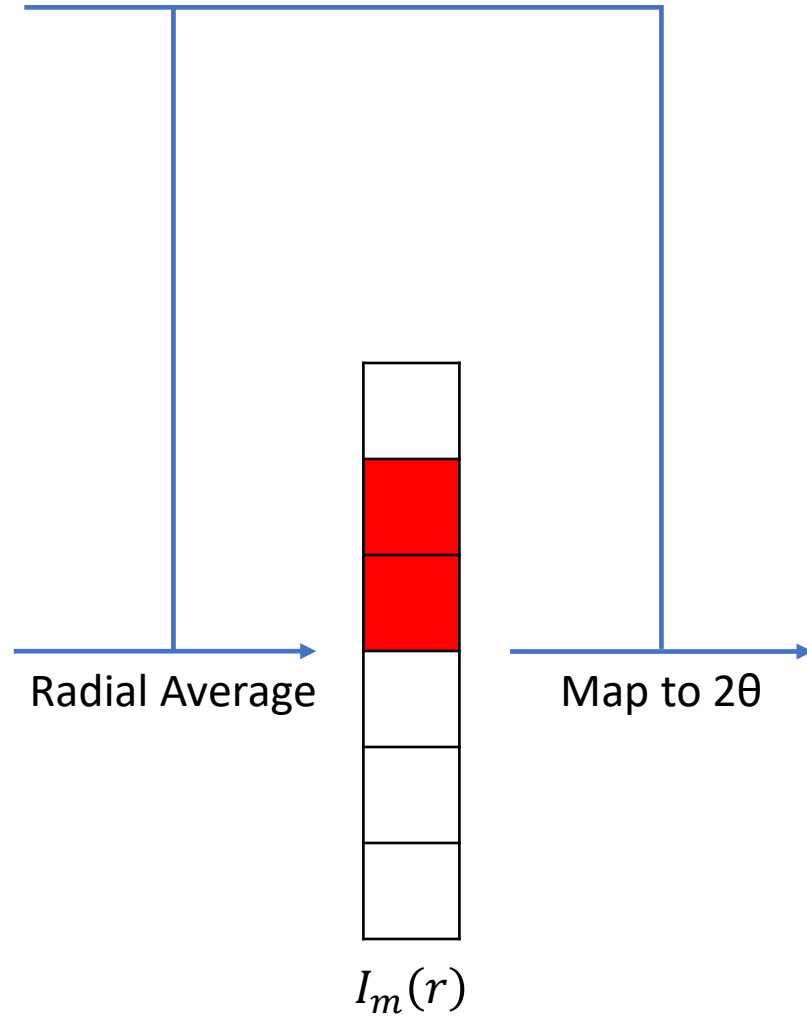
2θ positions

2.8	2.2	2	2.2	2.8
2.2	1.4	1	1.4	2.2
2	1	0	1	2
2.2	1.4	1	1.4	2.2
2.8	2.2	2	2.2	2.8

Measured Signal



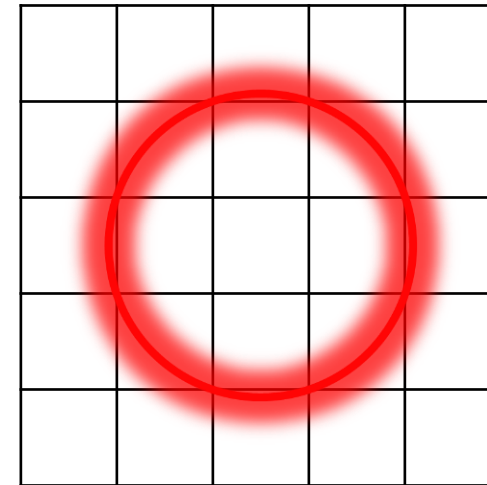
$I_m(x, y)$



$I_m(r)$

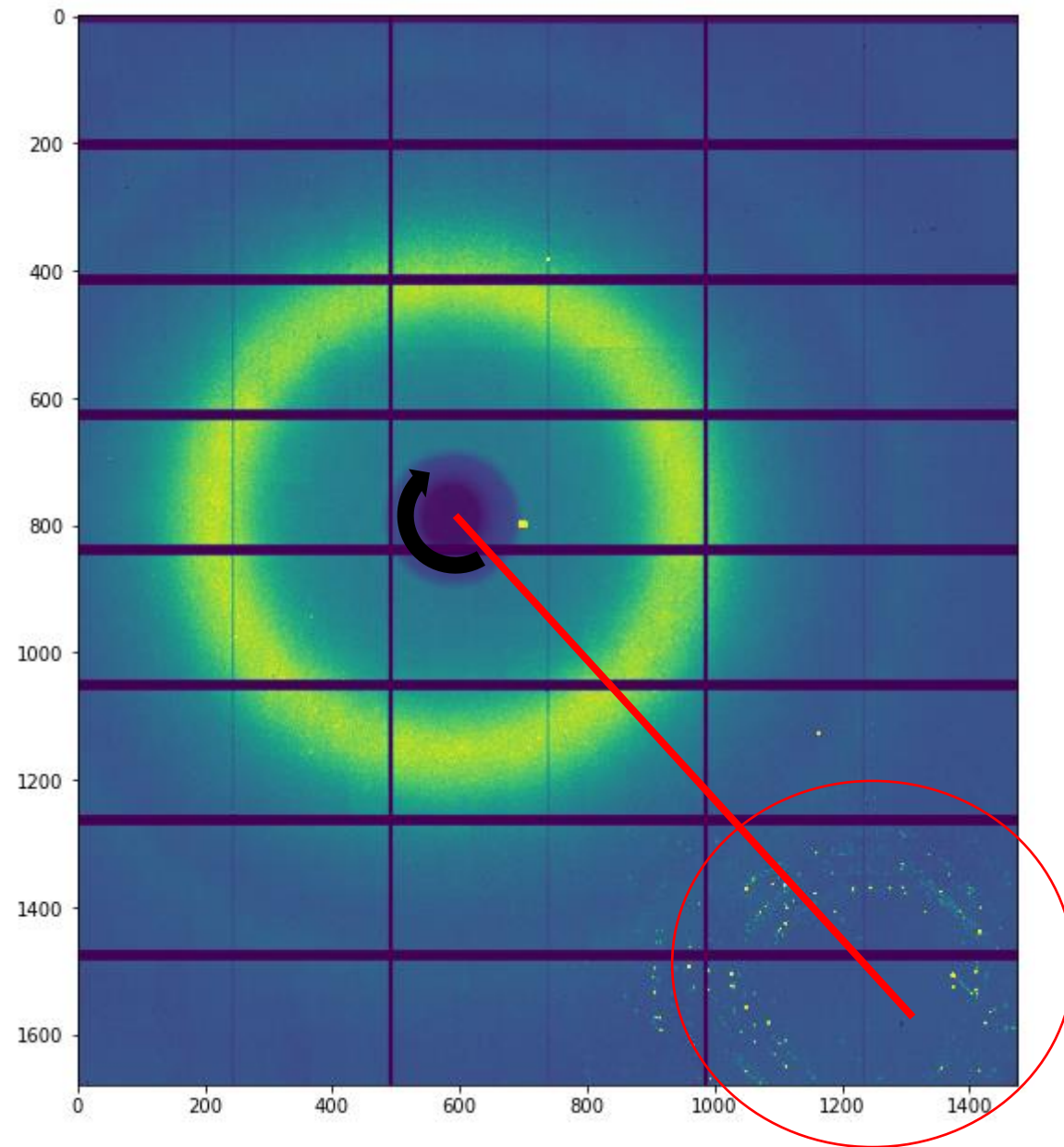
$$I_m(x, y) = G(x, y) \cdot S(x, y) + \epsilon$$

$$\hat{G}(x, y) = \frac{I_m(x, y)}{\hat{S}(x, y)}$$

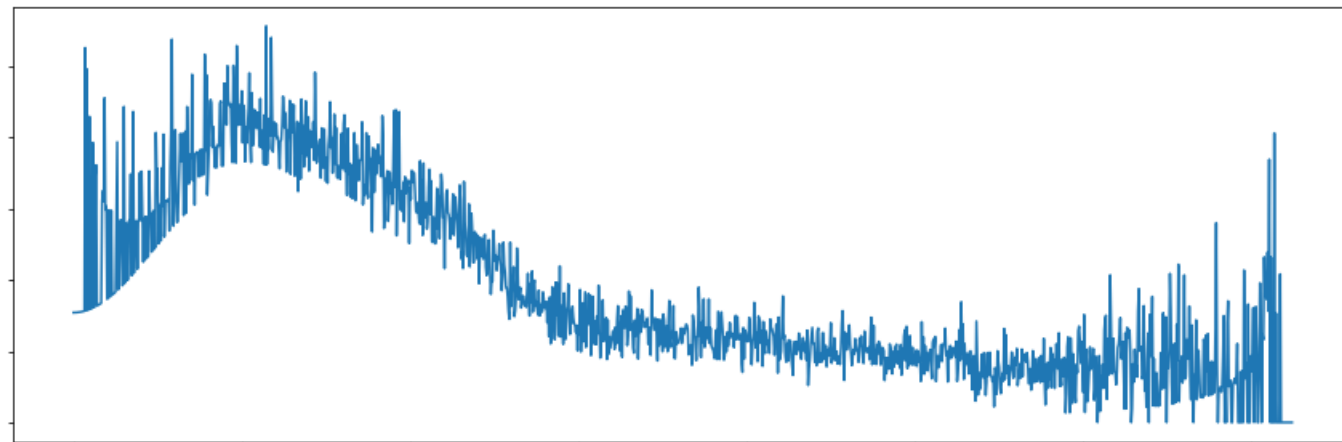
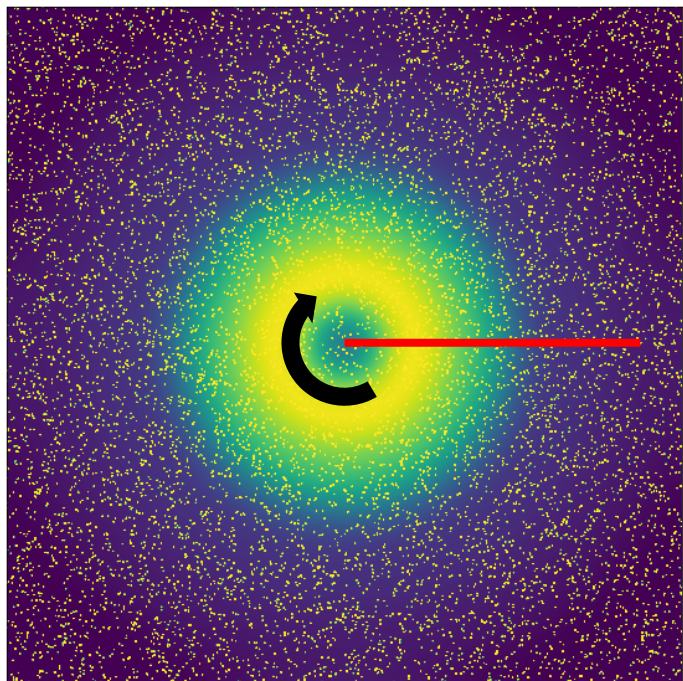
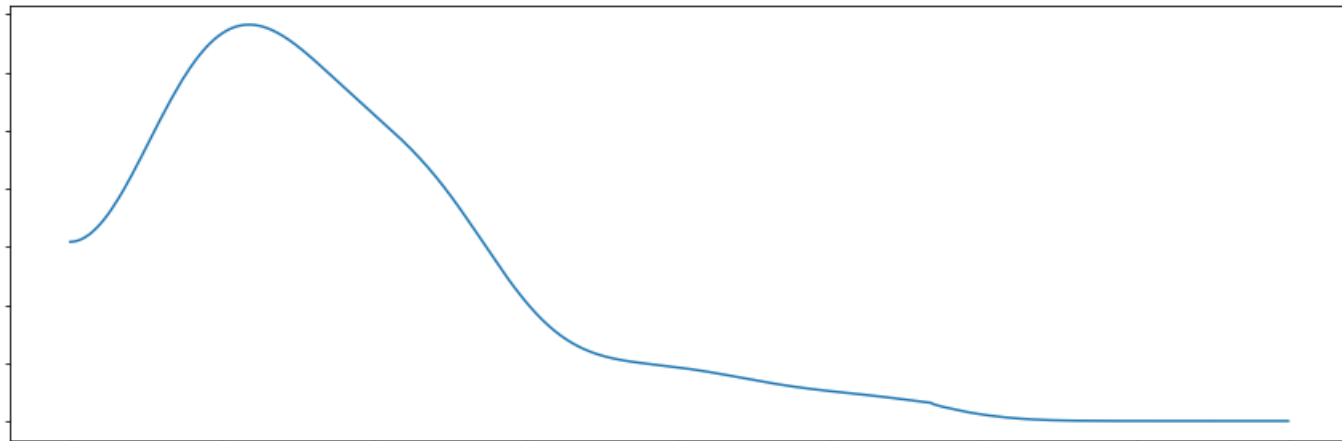
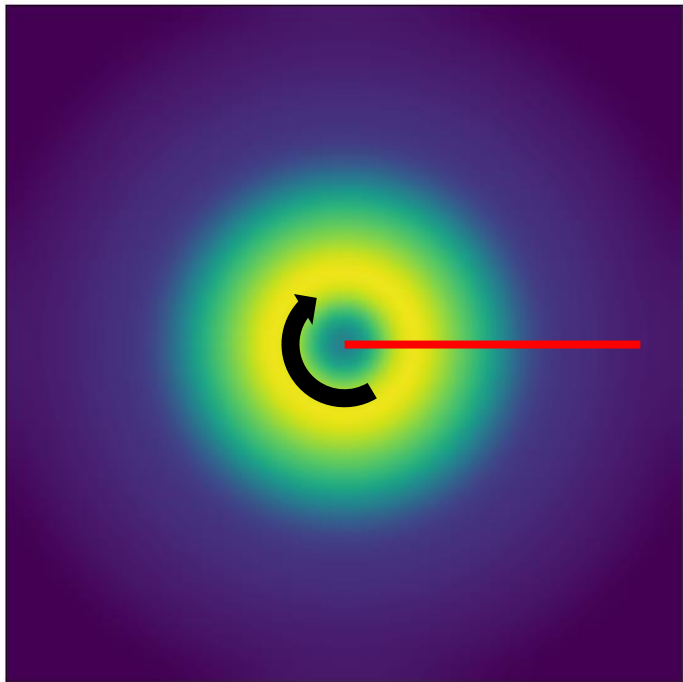


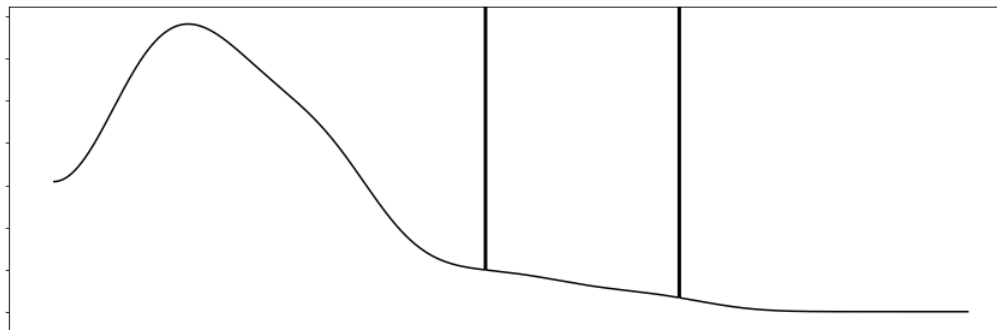
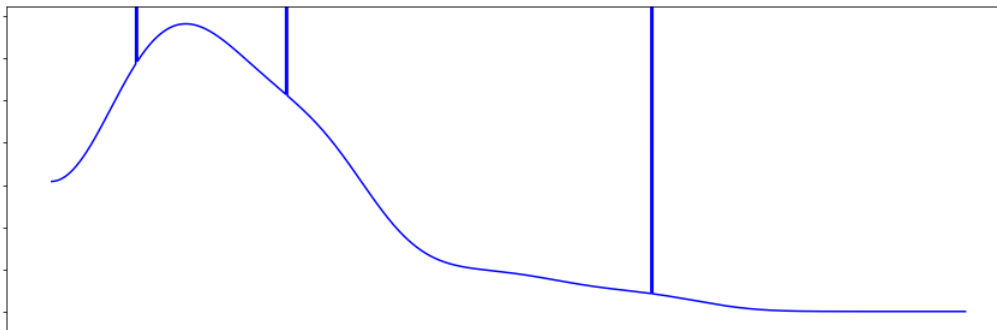
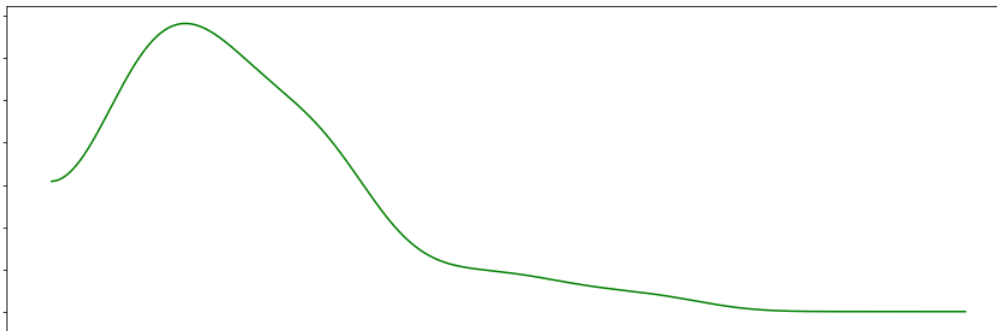
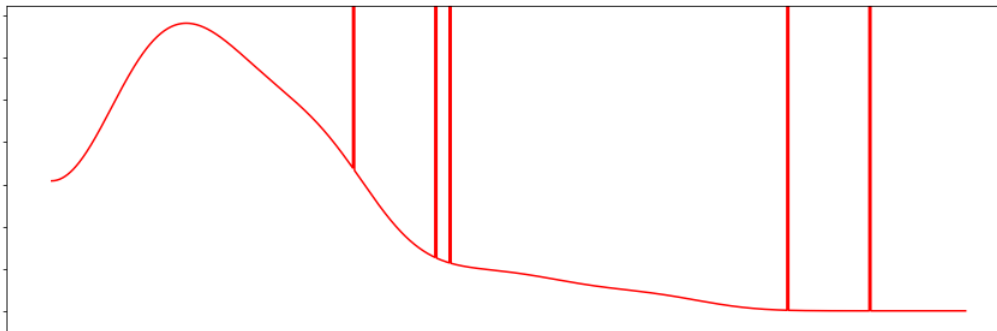
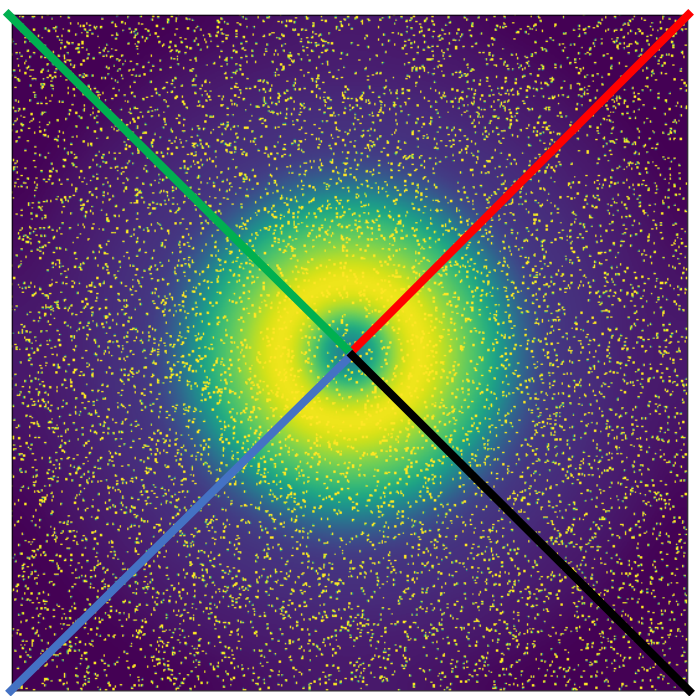
$\hat{S}(x, y)$

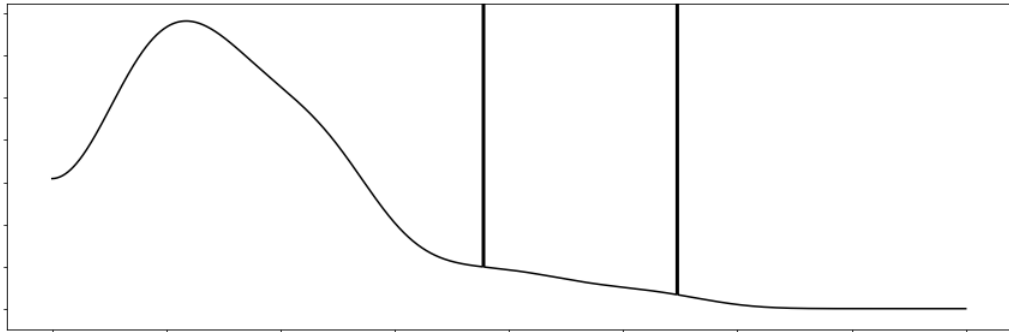
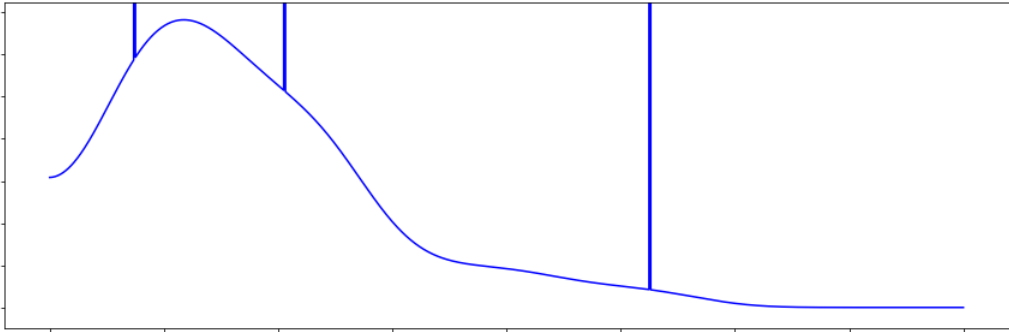
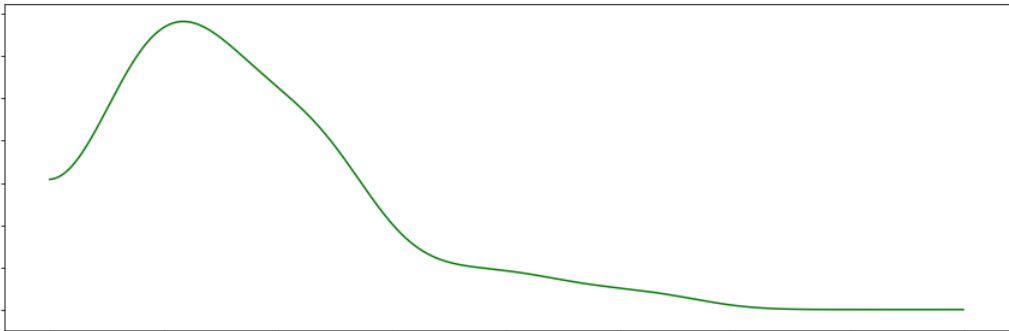
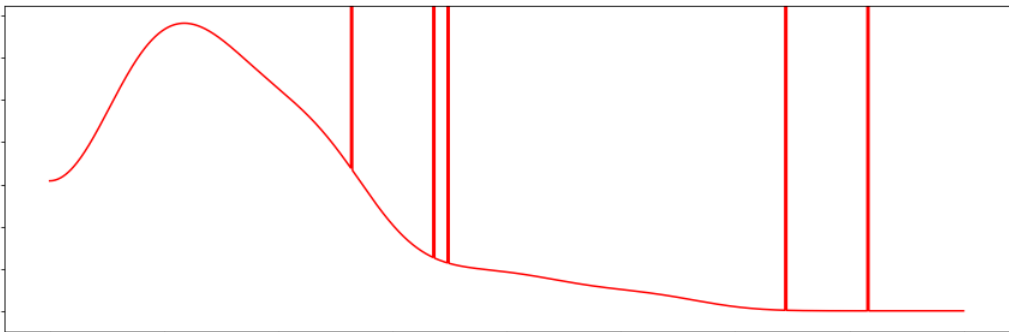




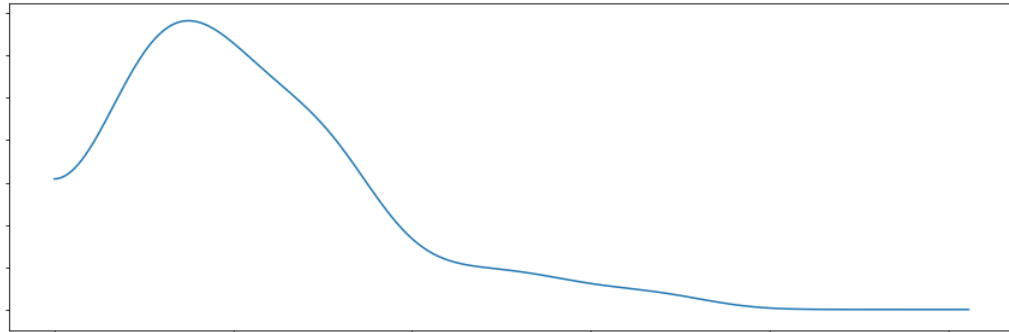
Outliers!

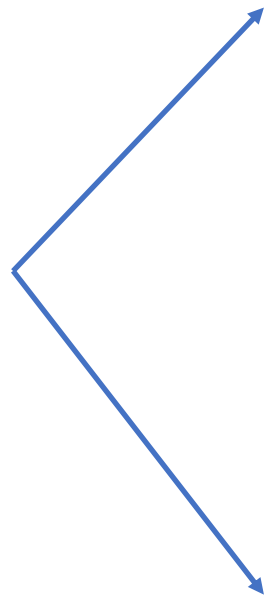
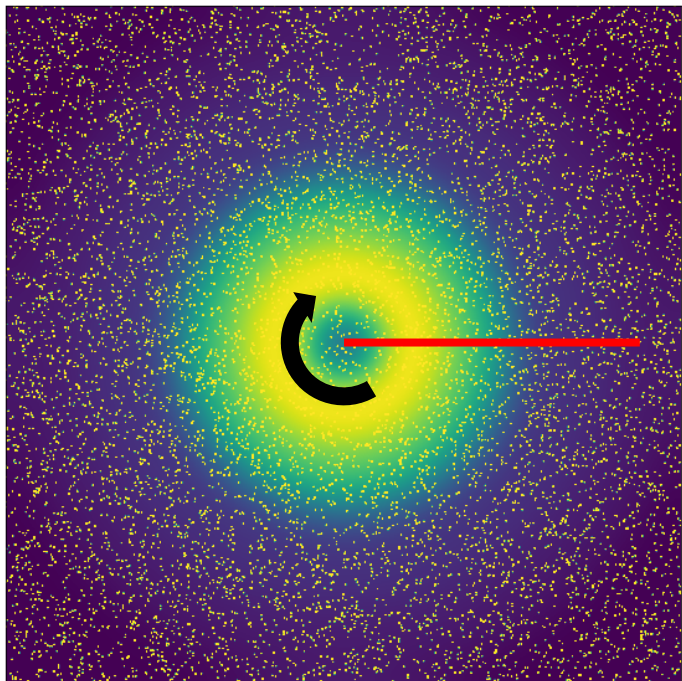




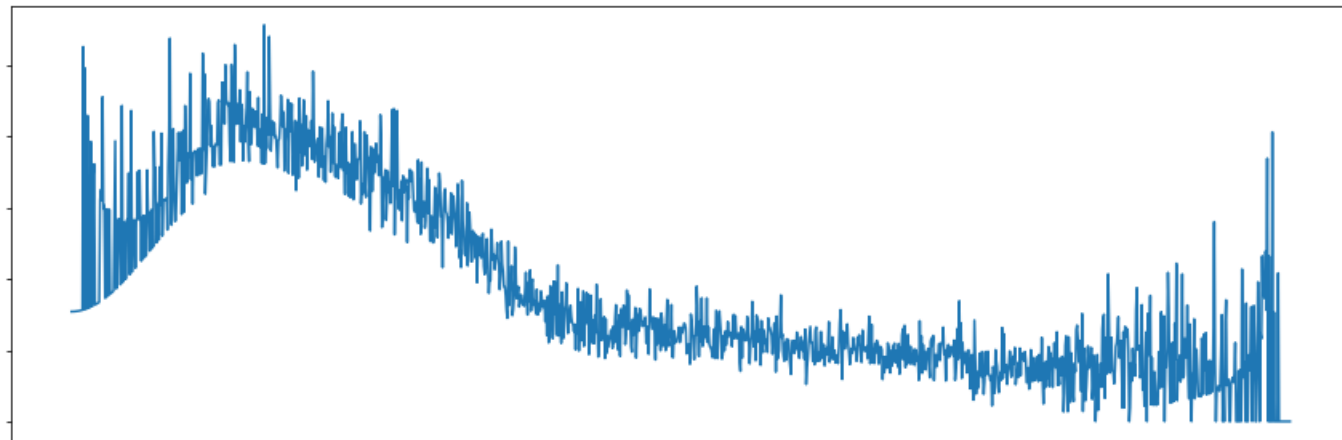


Median

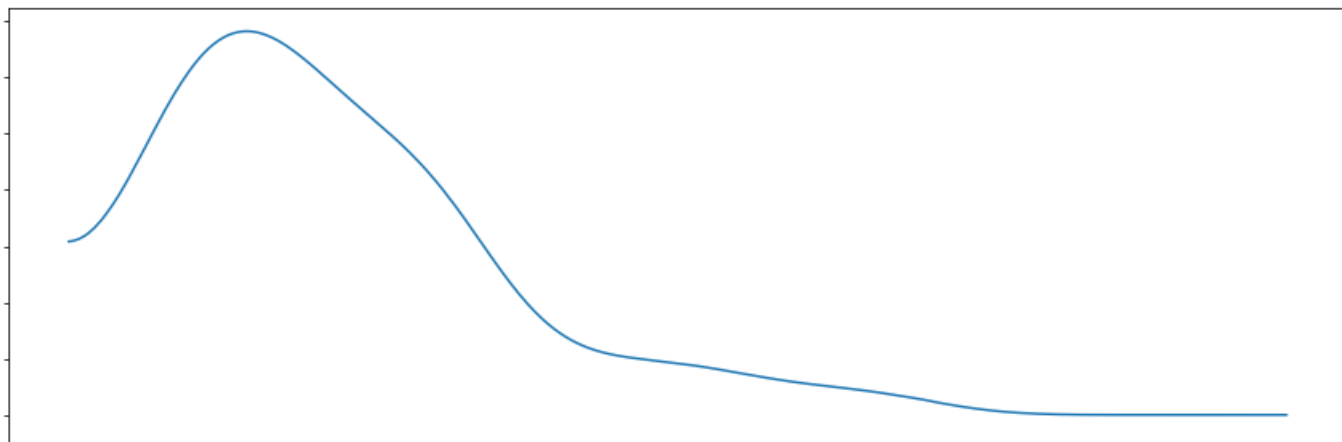




Radial Mean



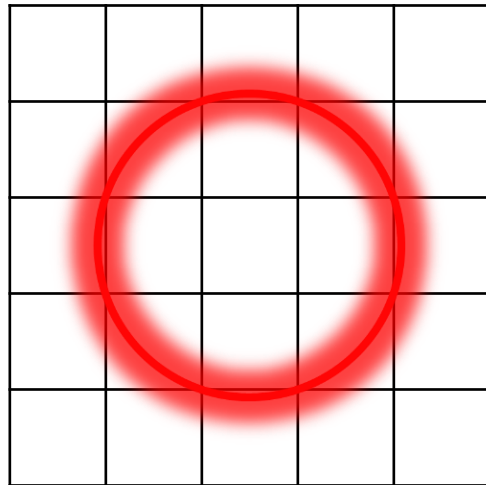
Radial Median



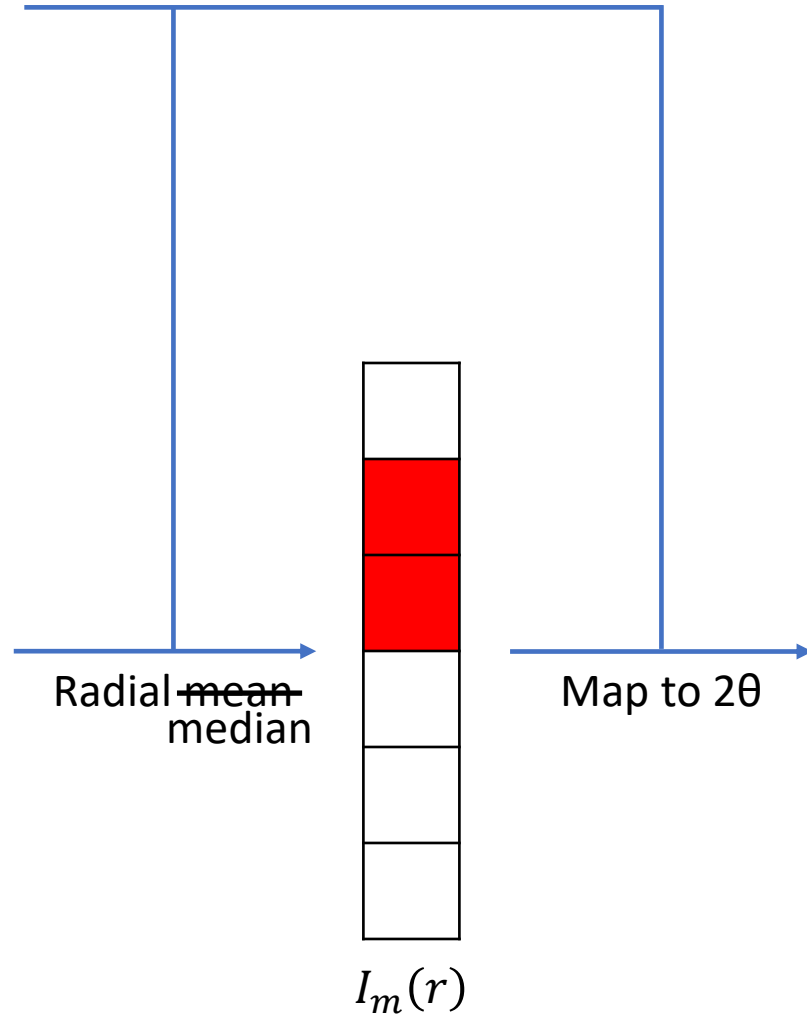
2θ positions

2.8	2.2	2	2.2	2.8
2.2	1.4	1	1.4	2.2
2	1	0	1	2
2.2	1.4	1	1.4	2.2
2.8	2.2	2	2.2	2.8

Measured Signal



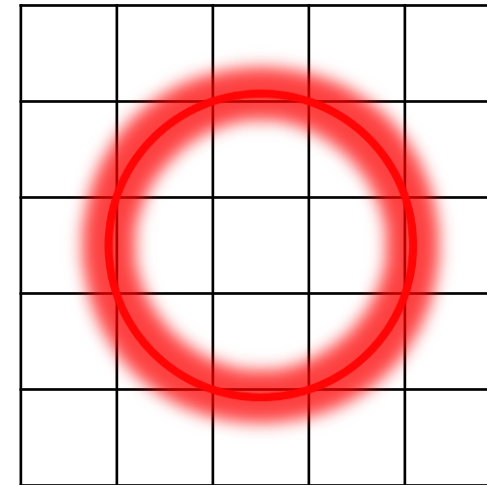
$I_m(x, y)$



$I_m(r)$

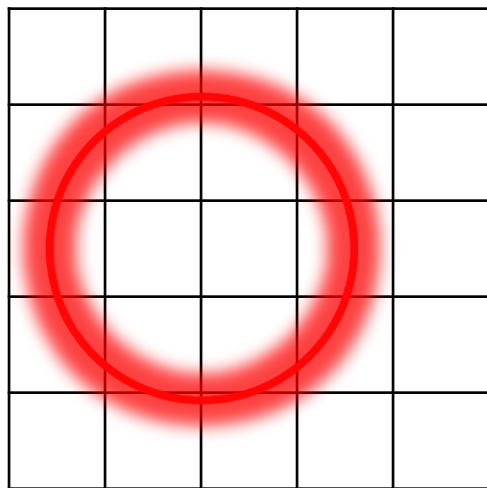
$$I_m(x, y) = G(x, y) \cdot S(x, y) + \epsilon$$

$$\hat{G}(x, y) = \frac{I_m(x, y)}{\hat{S}(x, y)}$$

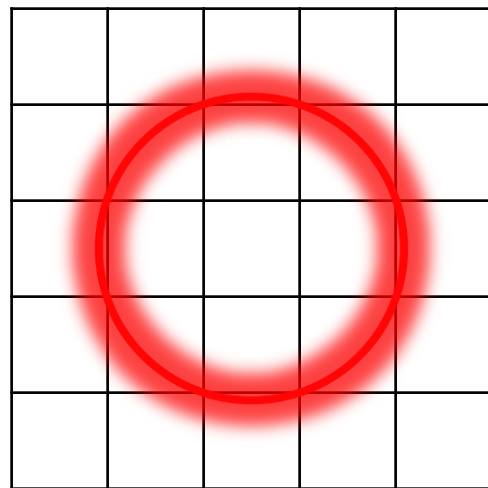


$\hat{S}(x, y)$

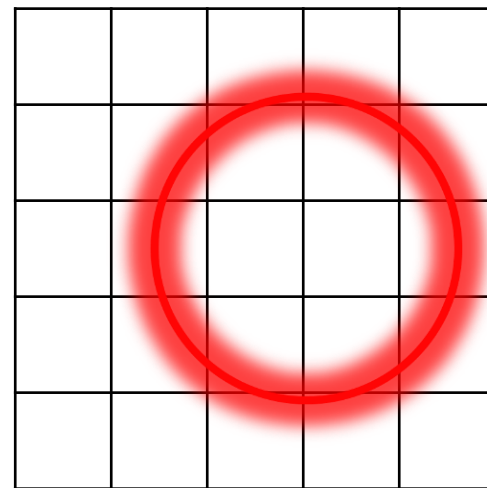
# Measured Signal



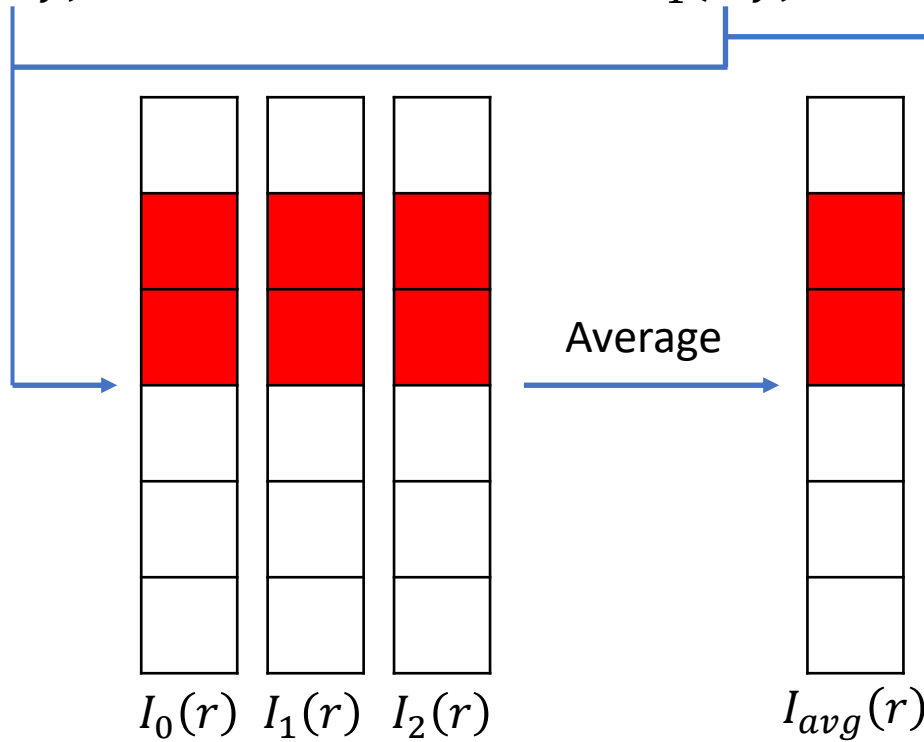
$I_0(x, y)$



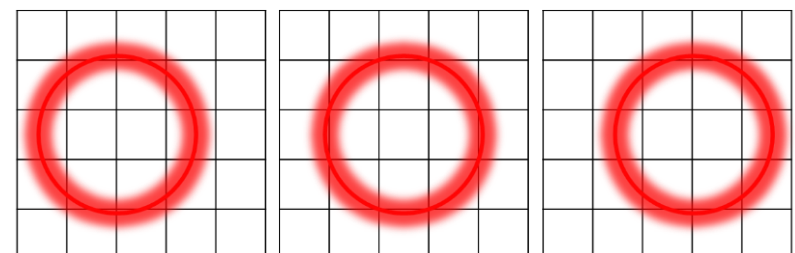
$I_1(x, y)$



$I_2(x, y)$



Map to  $2\theta$



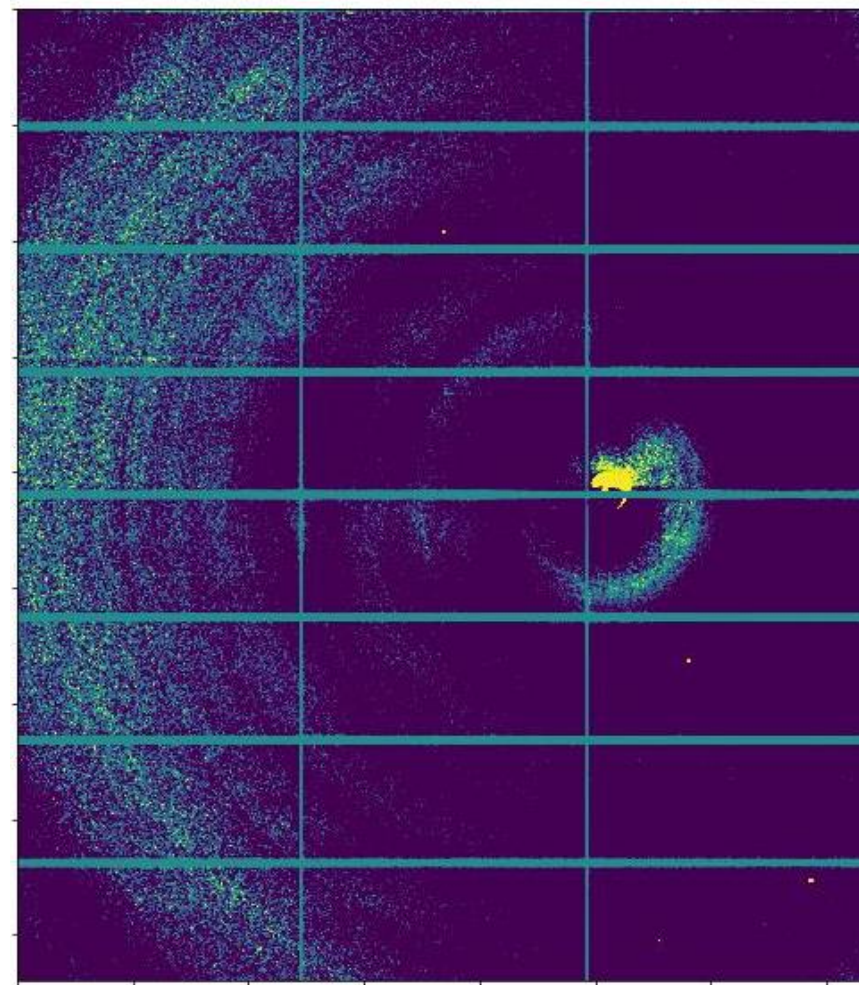
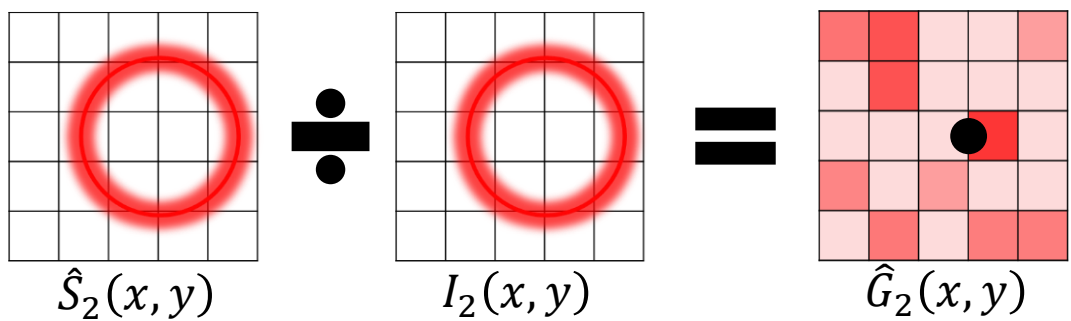
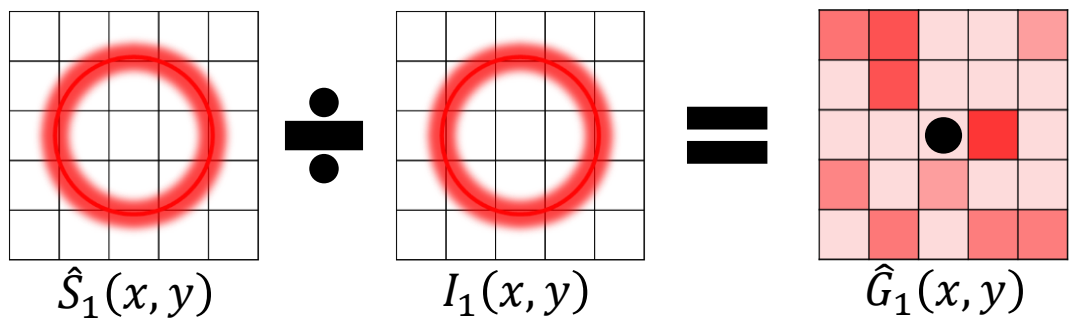
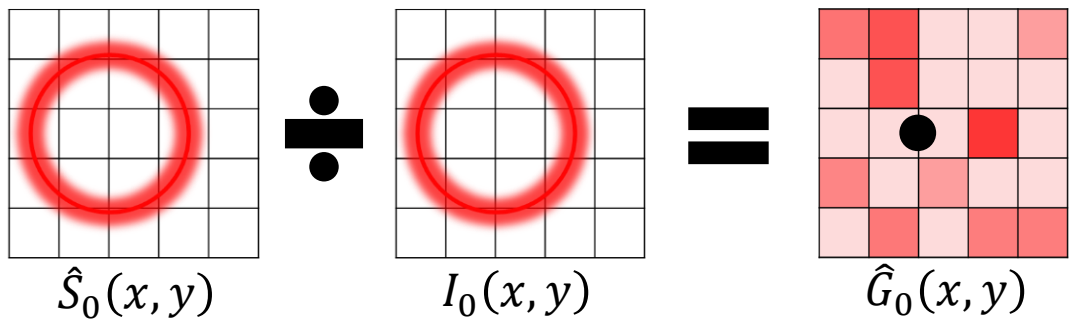
$\hat{S}_0(x, y)$

$\hat{S}_1(x, y)$

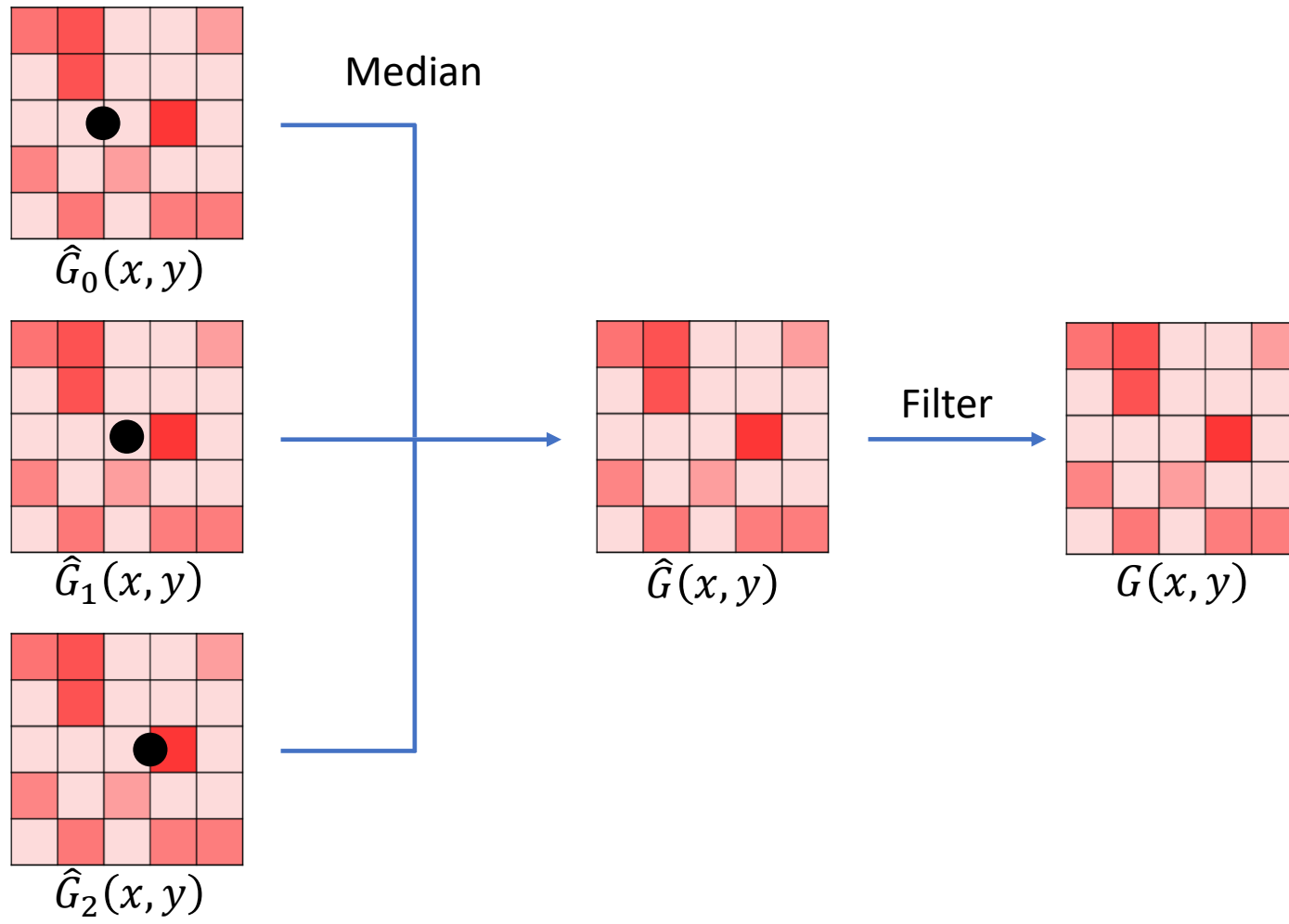
$\hat{S}_2(x, y)$

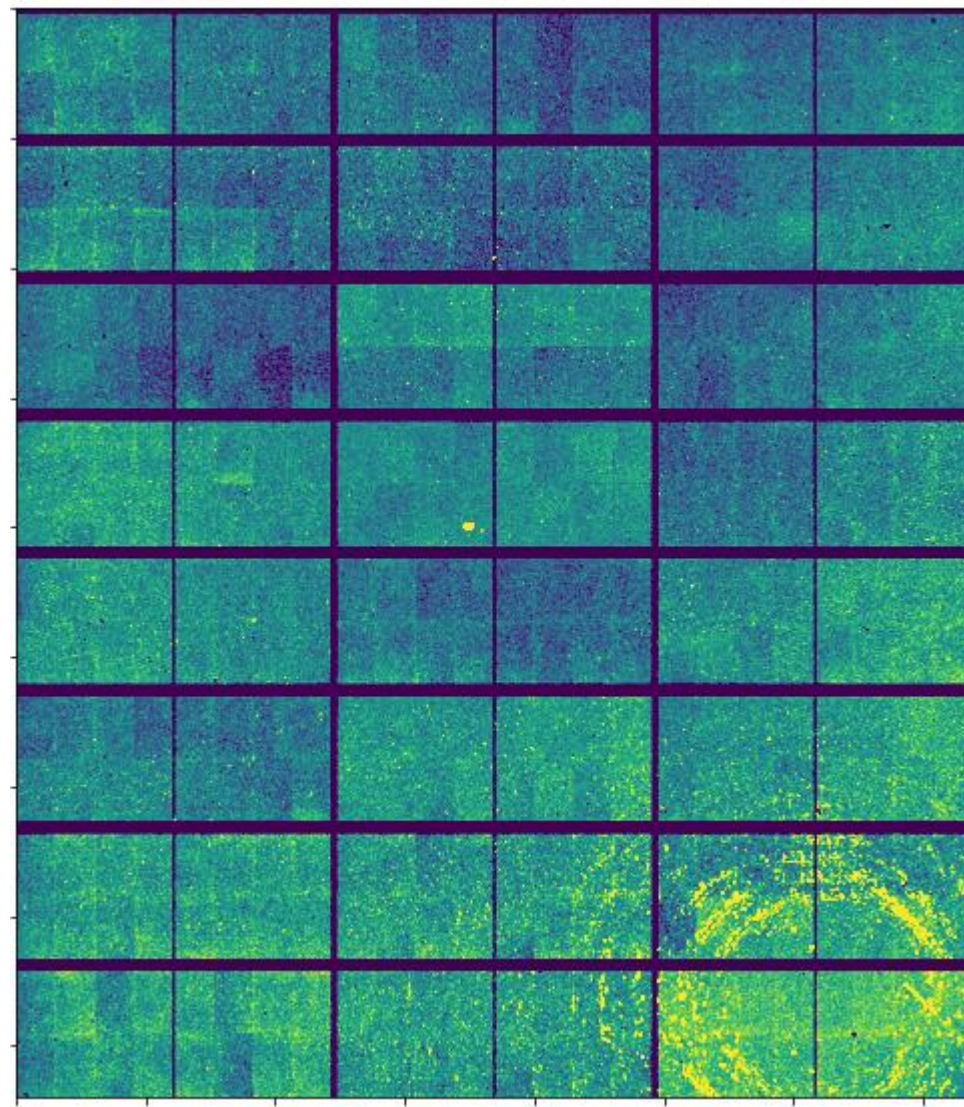
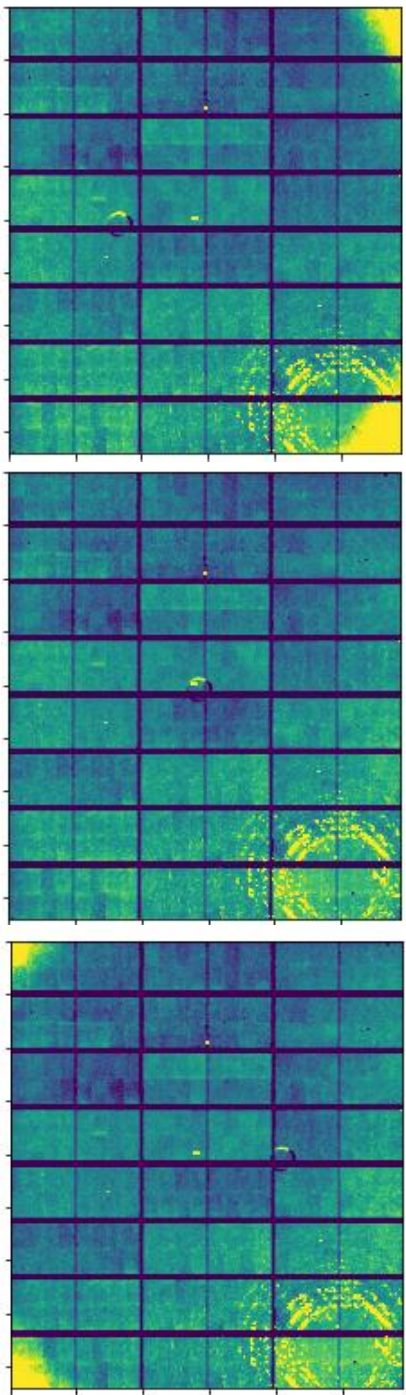


$$\hat{G}(x, y) = \frac{I_m(x, y)}{\hat{S}(x, y)}$$









# Median filter

Input

3	1	1	11	1
1	1	1	1	1
1	1	87	1	1
12	1	1	1	1
1	1	1	39	1

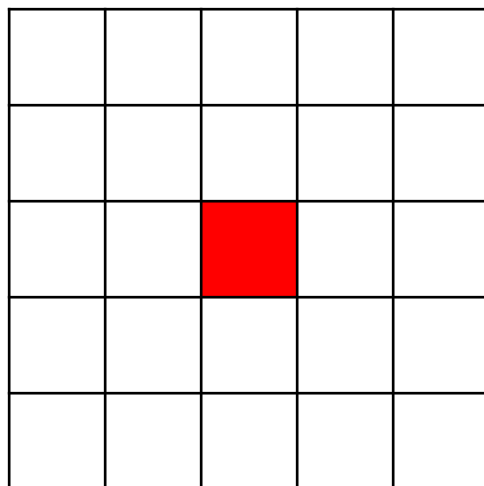


Output

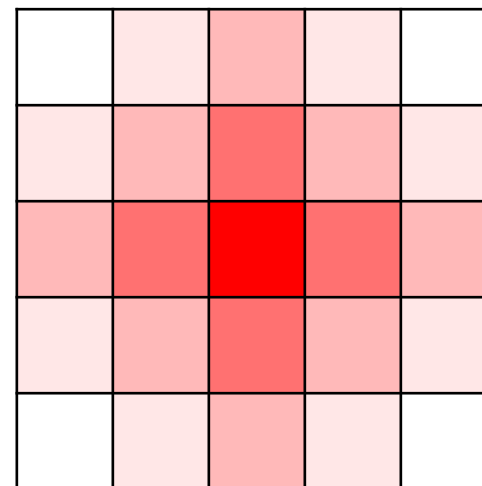
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

## Detector point spread

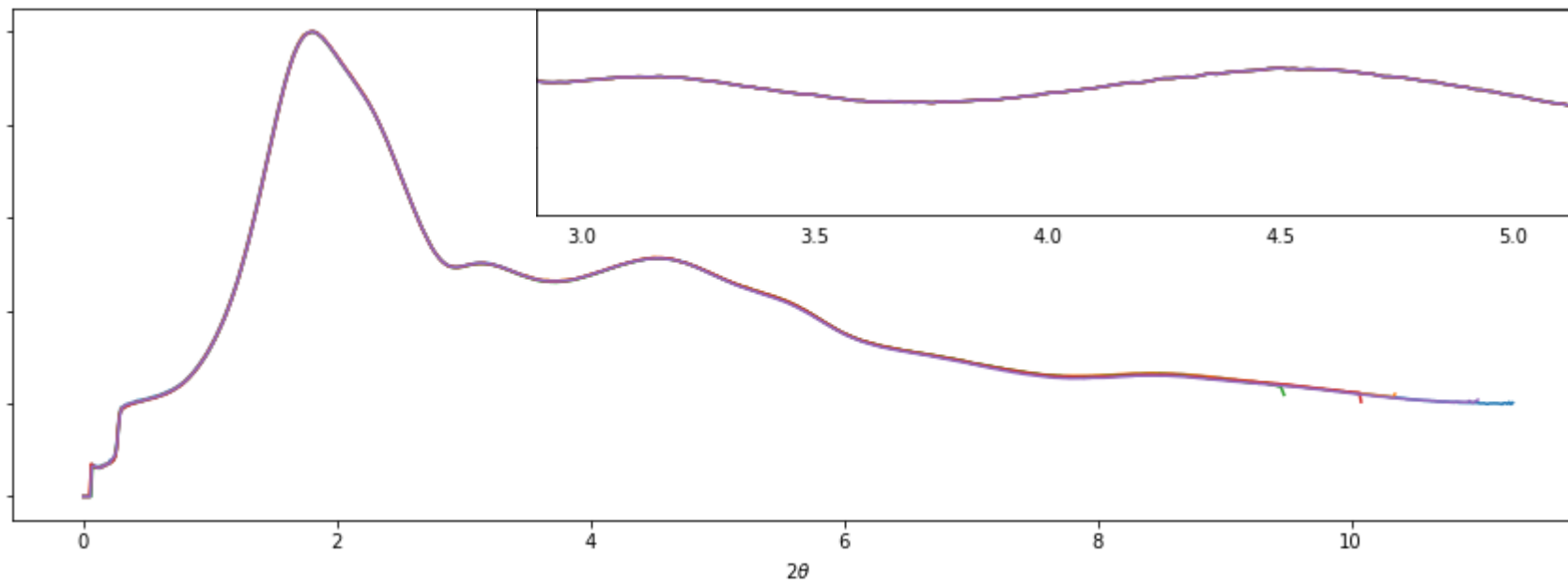
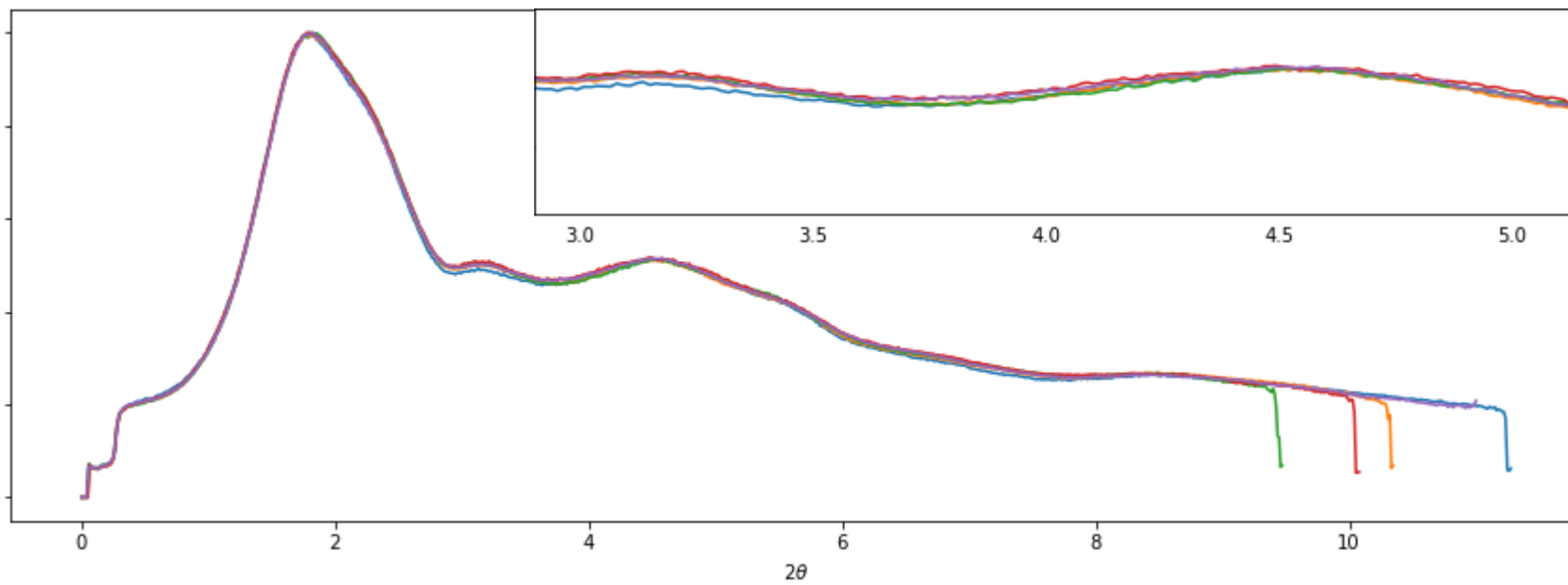
Pixels illuminated



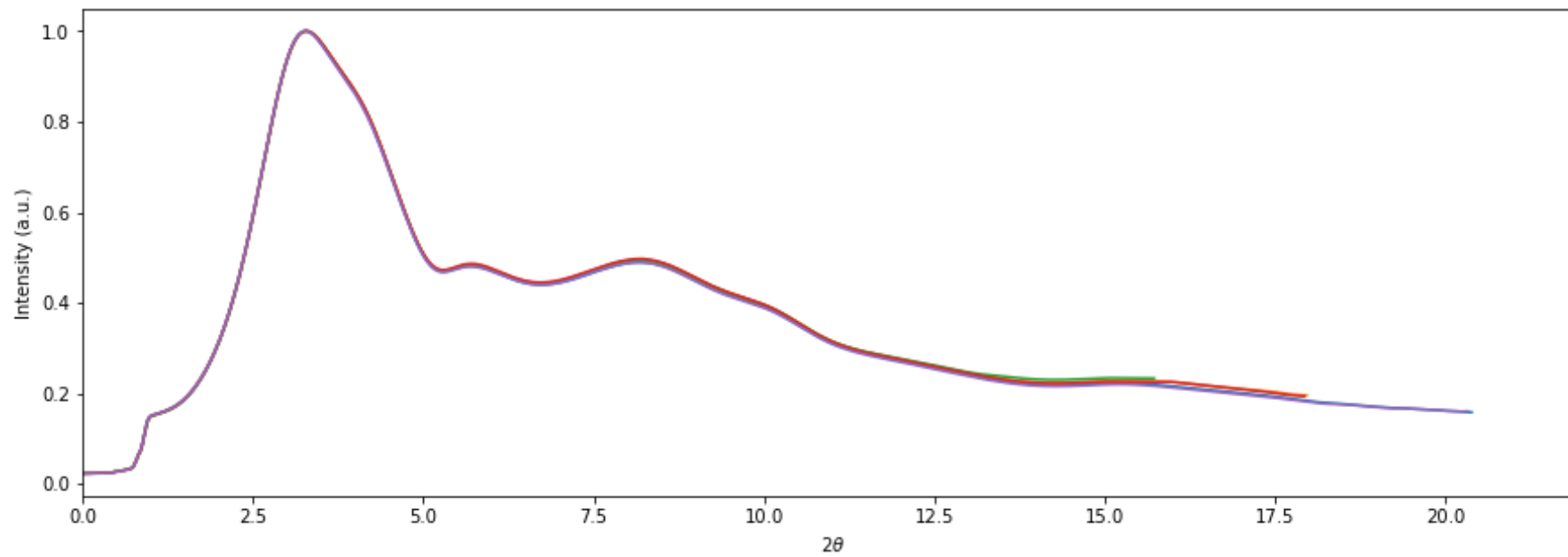
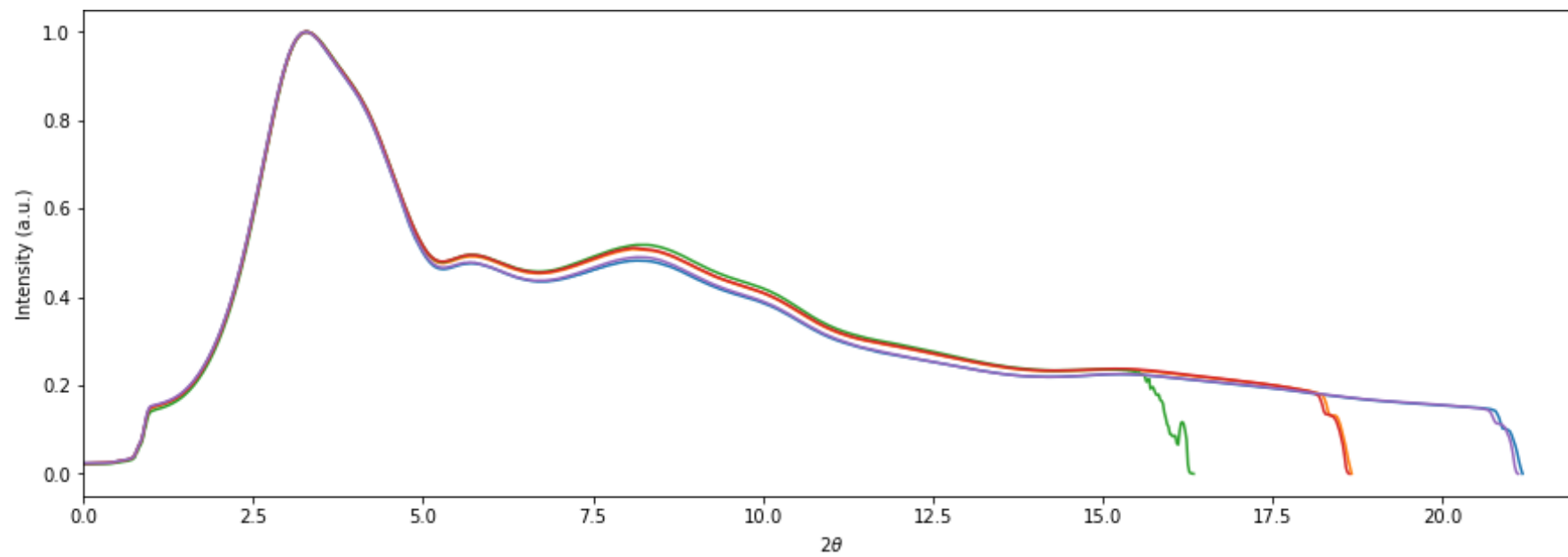
Detector measurement

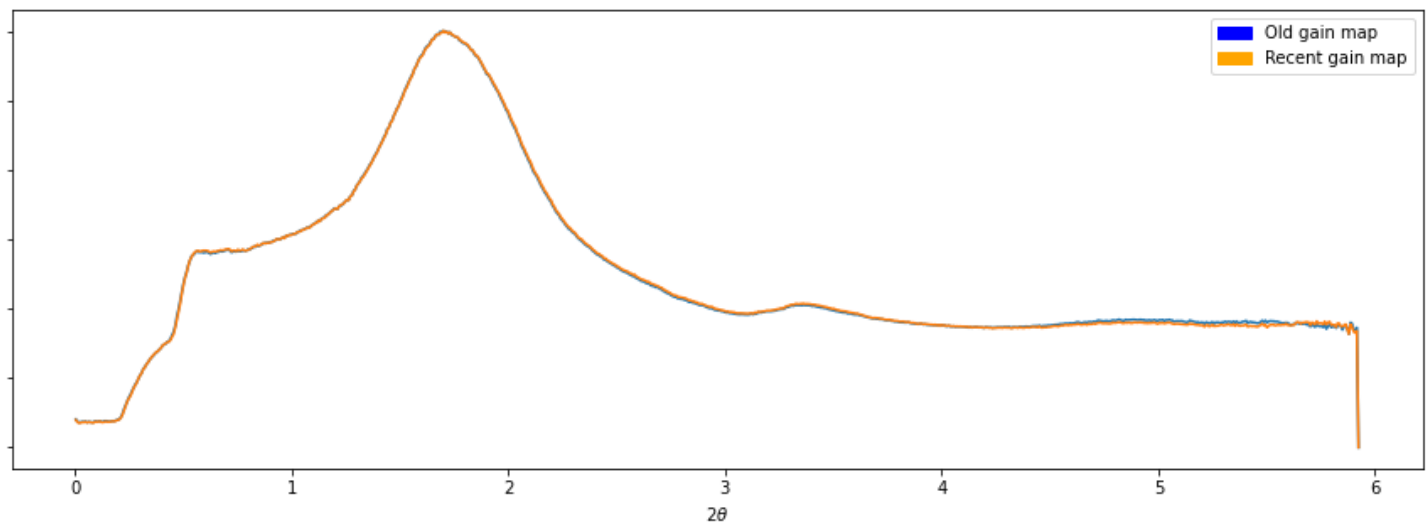
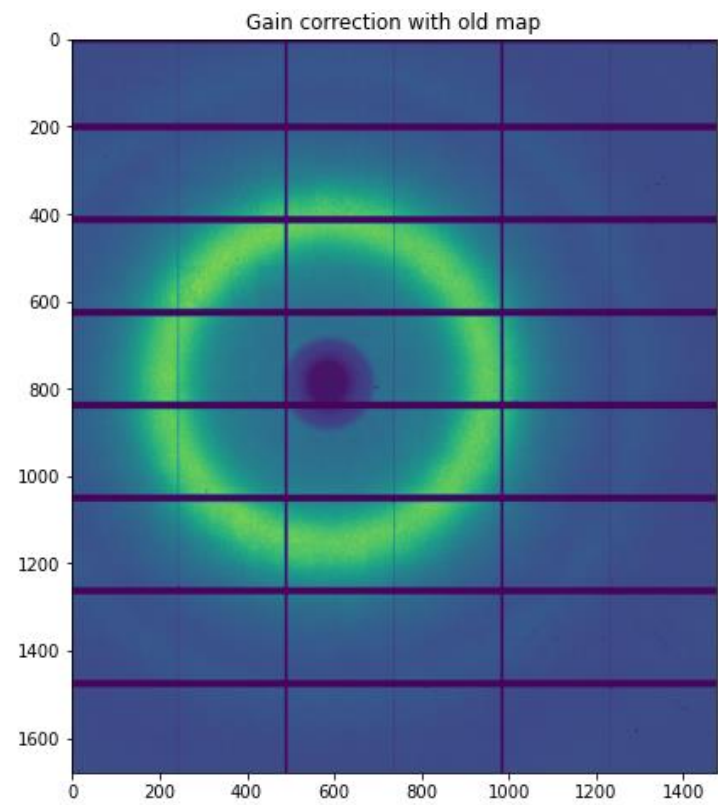
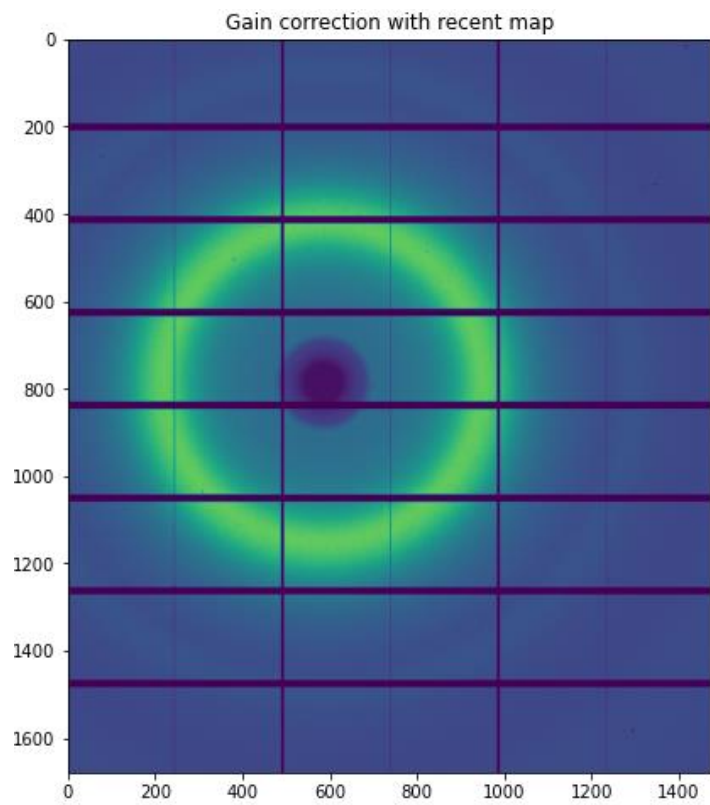


Pilatus

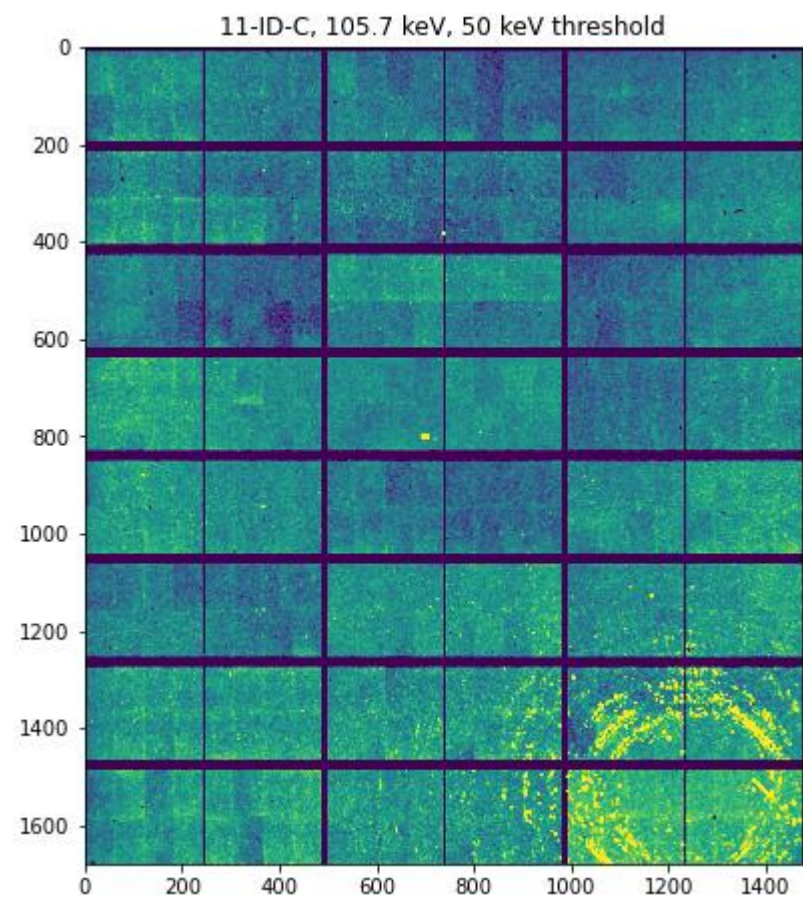
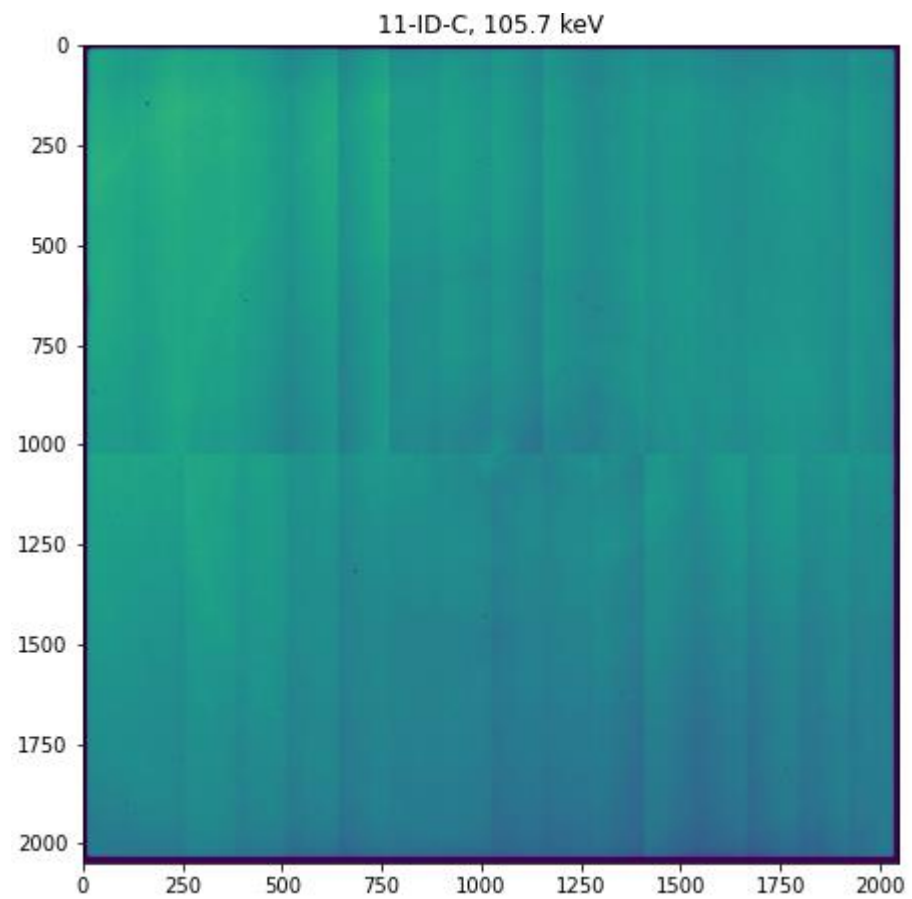


Perkin Elmer

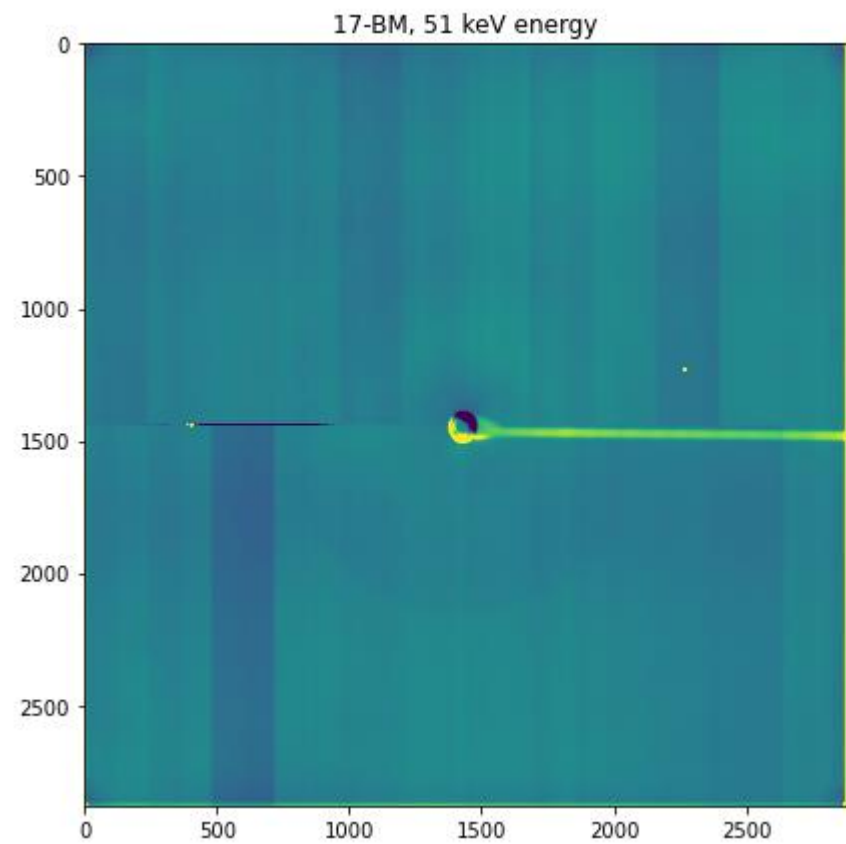
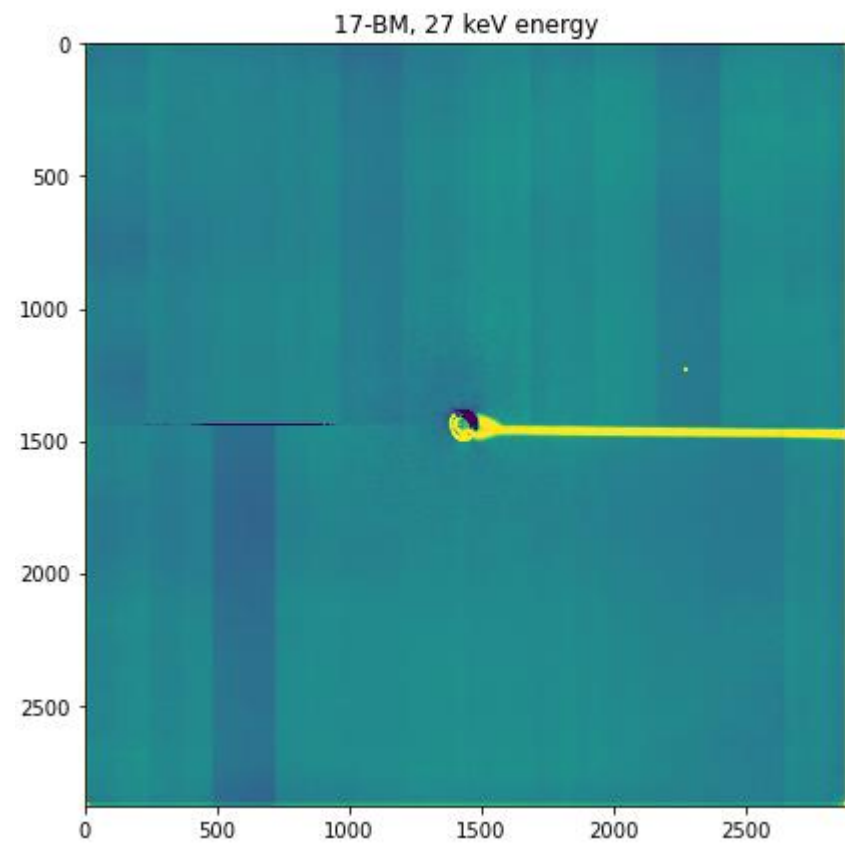


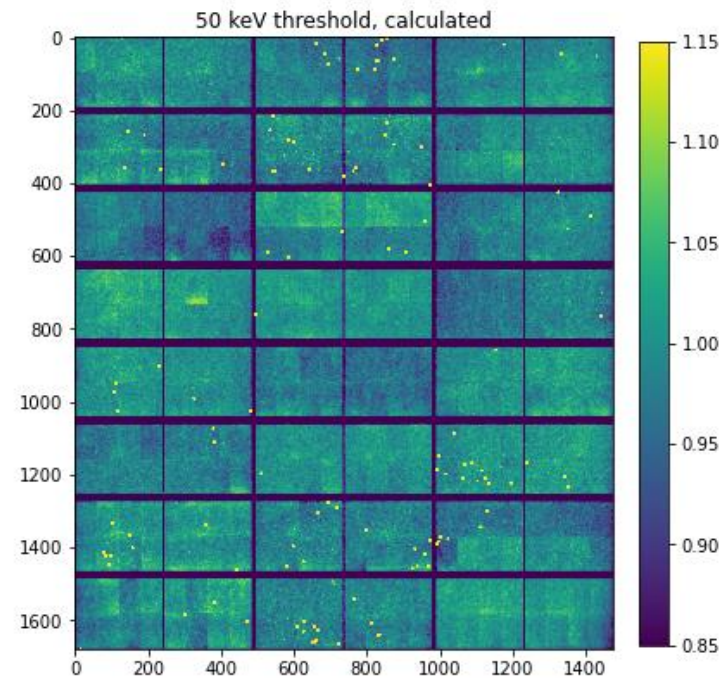
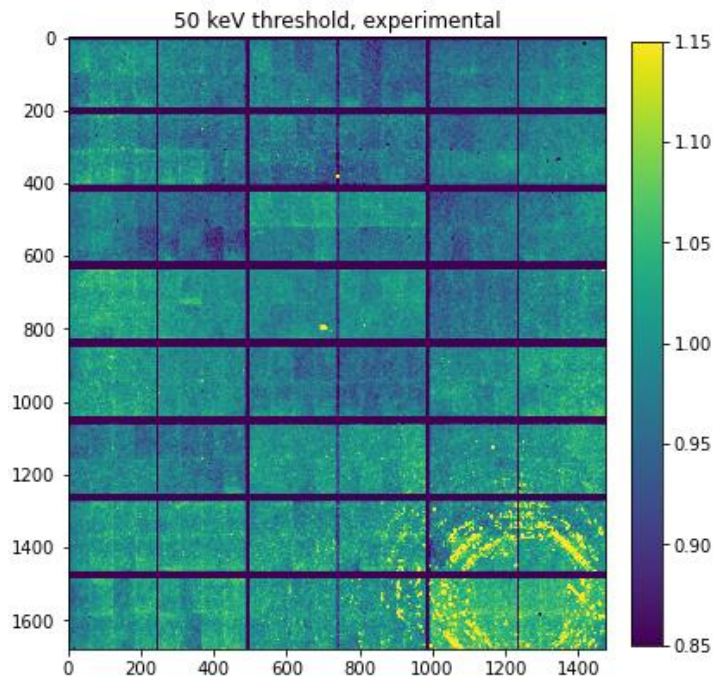
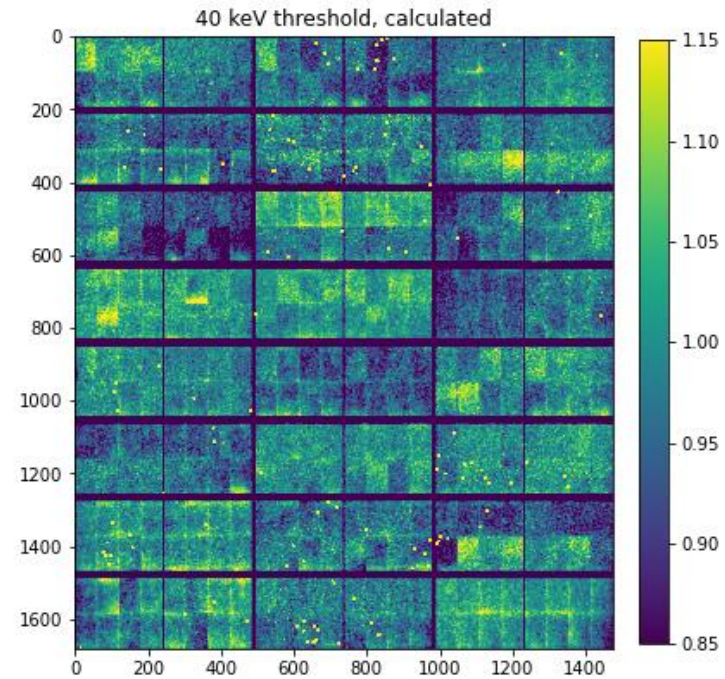
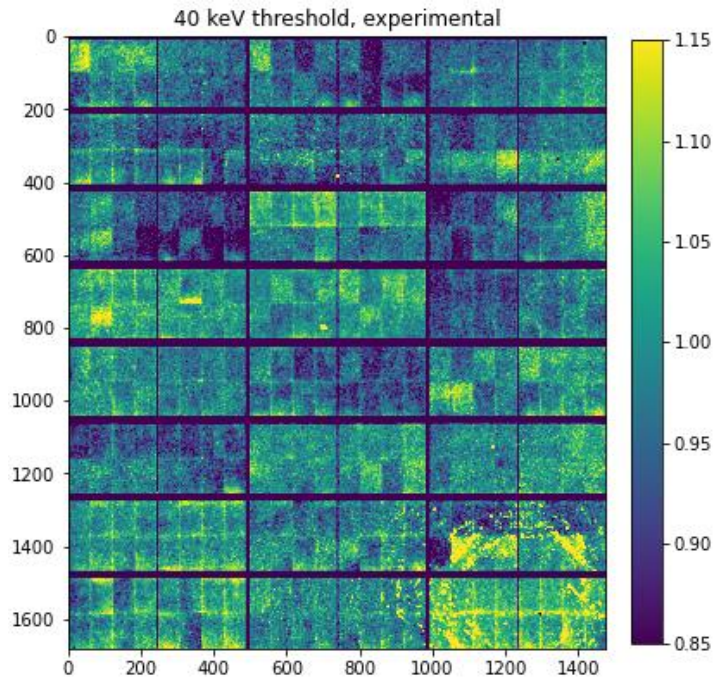




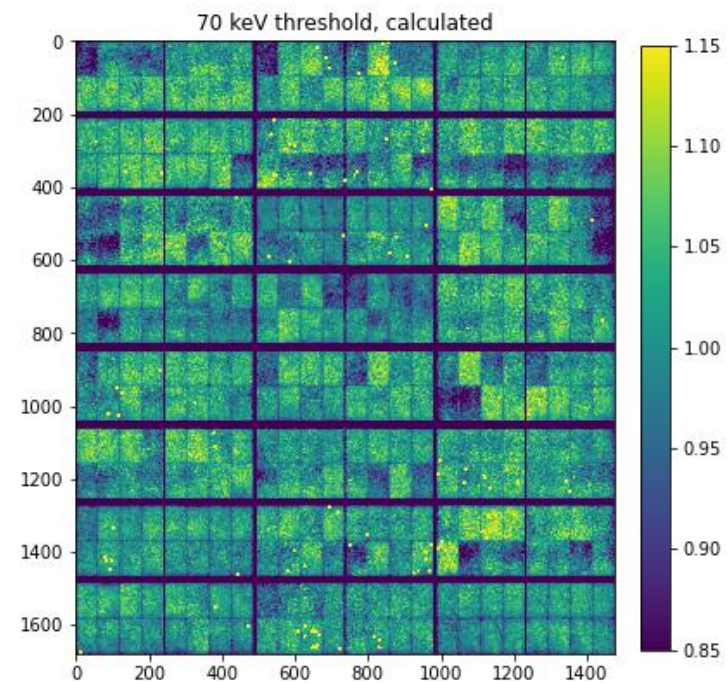
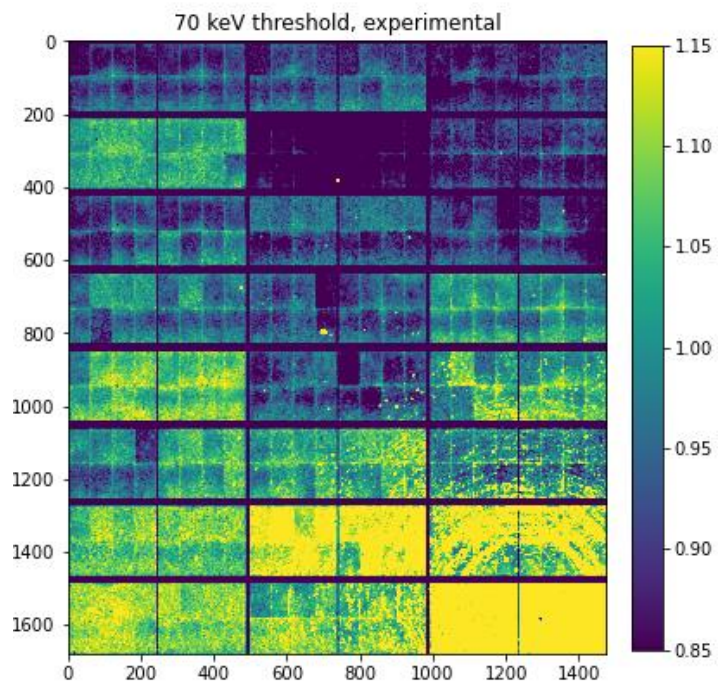
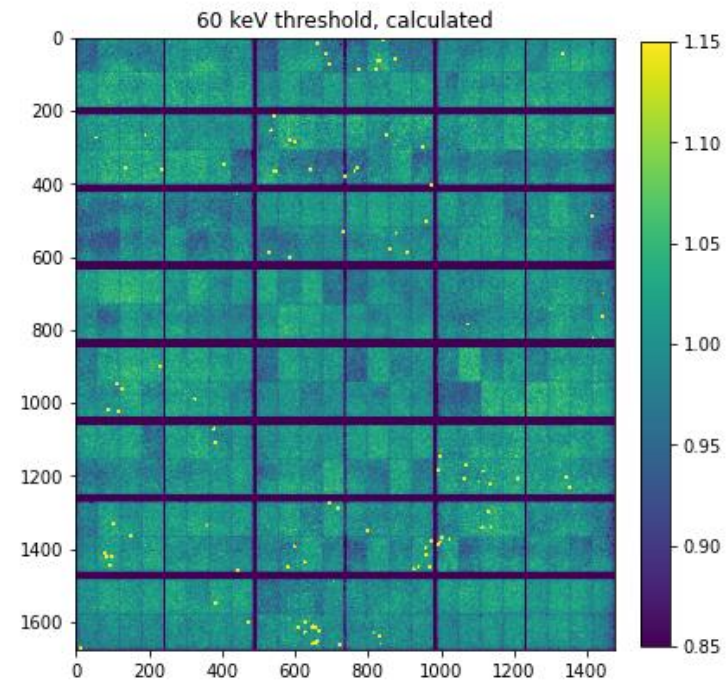
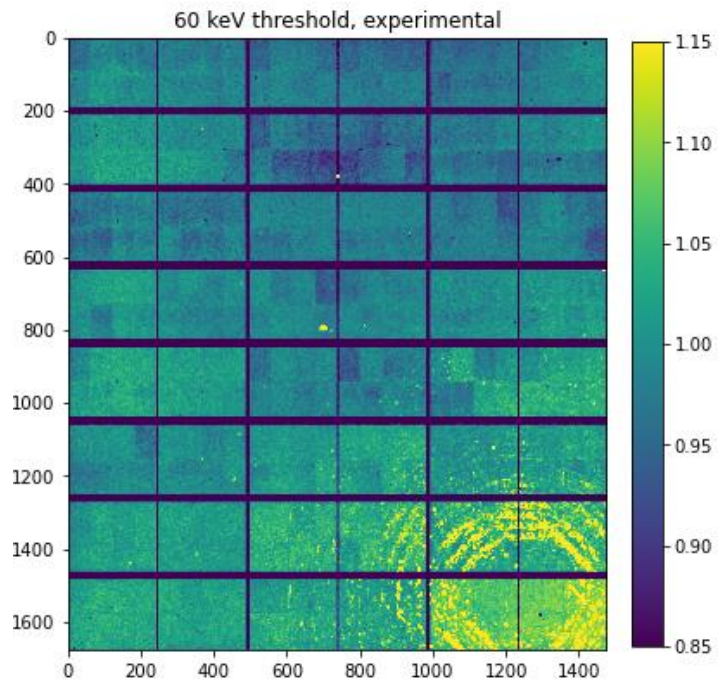


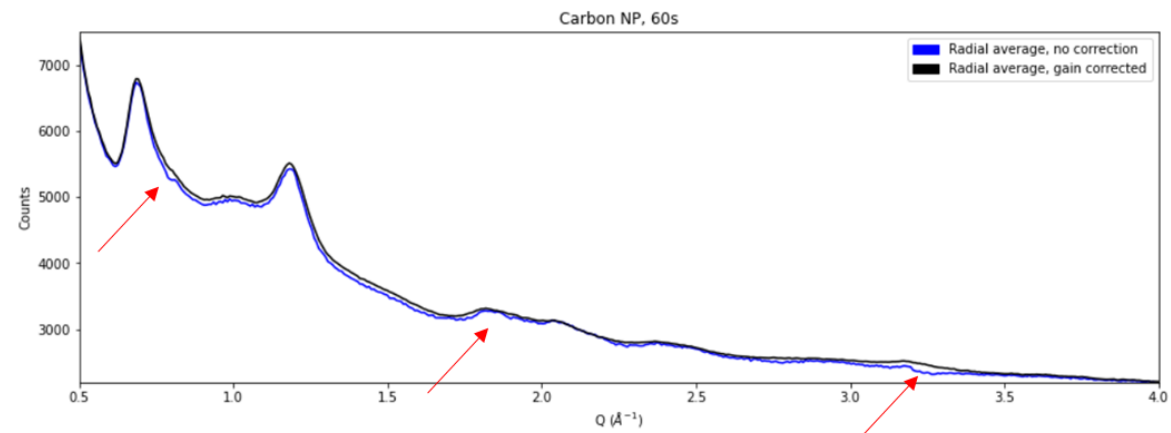
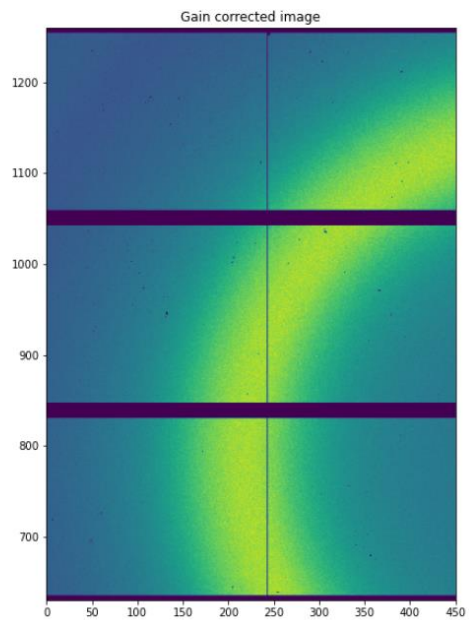
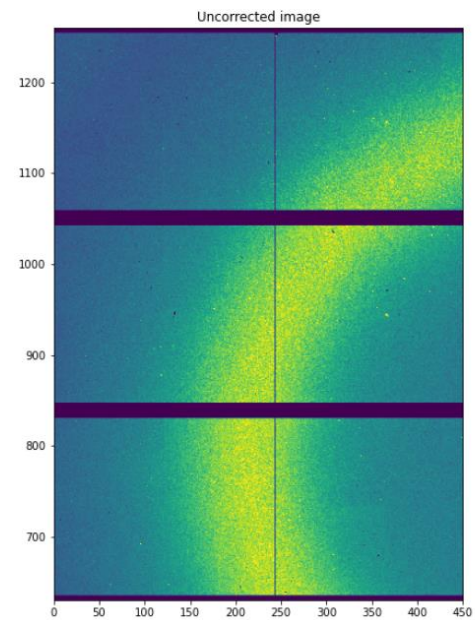
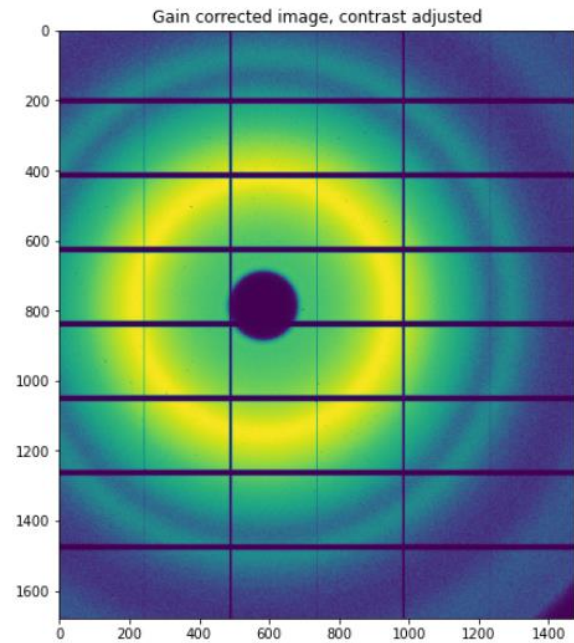
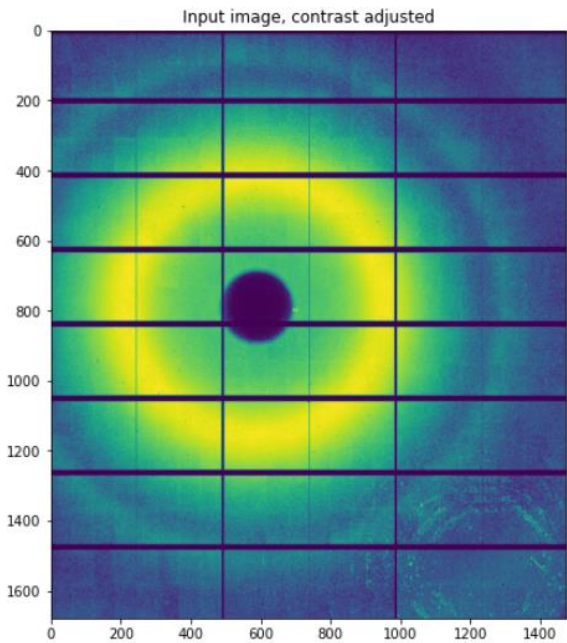




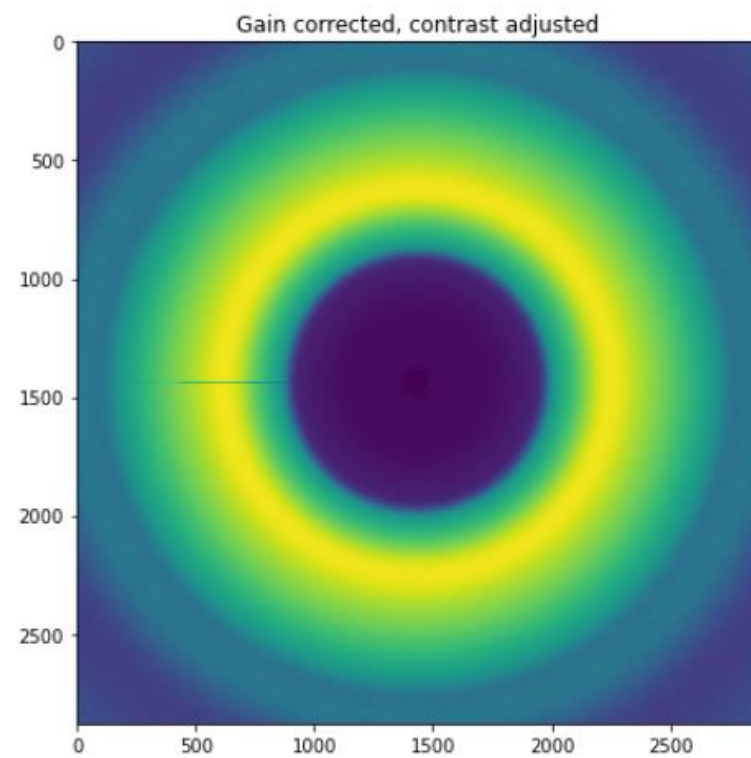
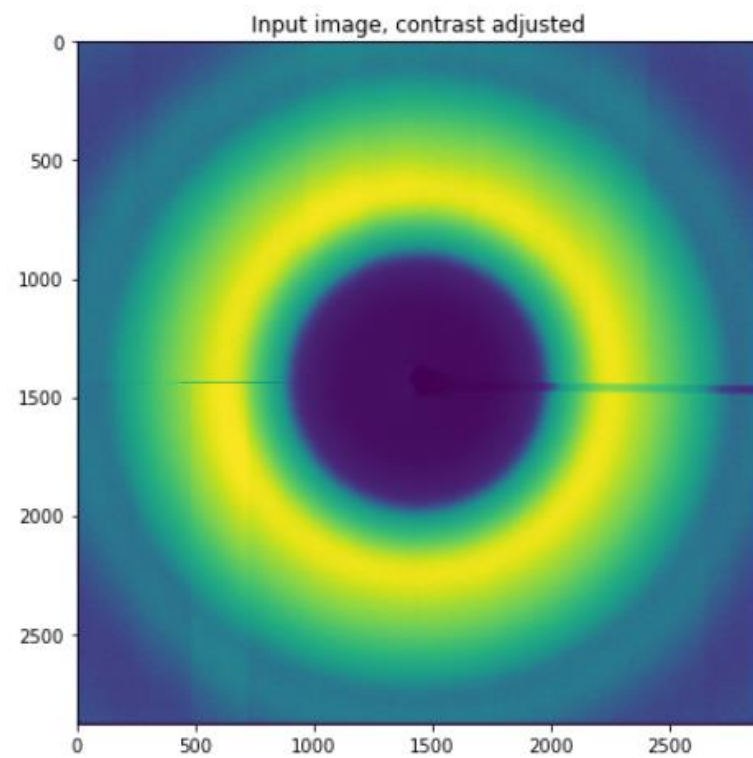




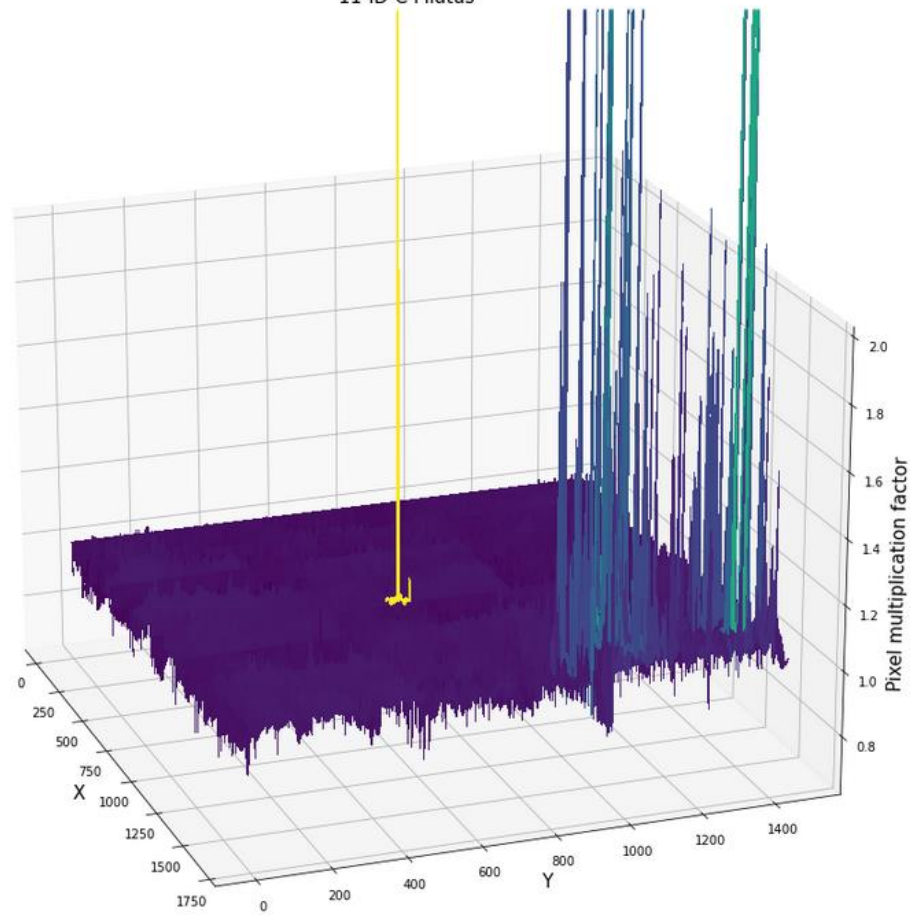




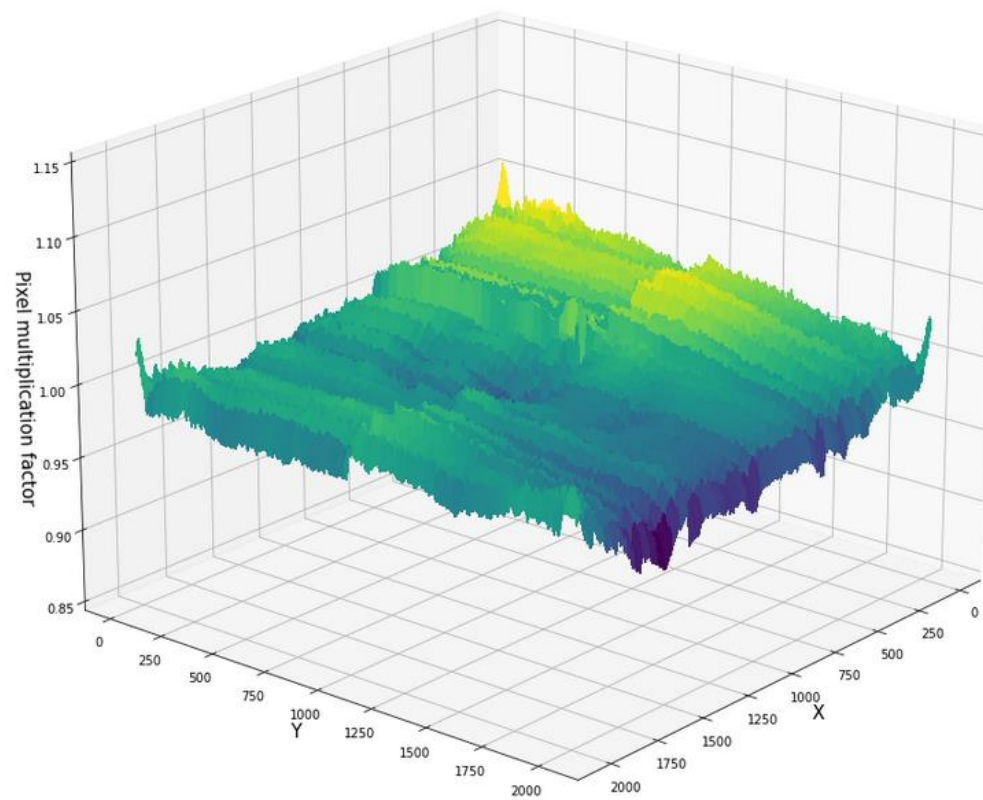




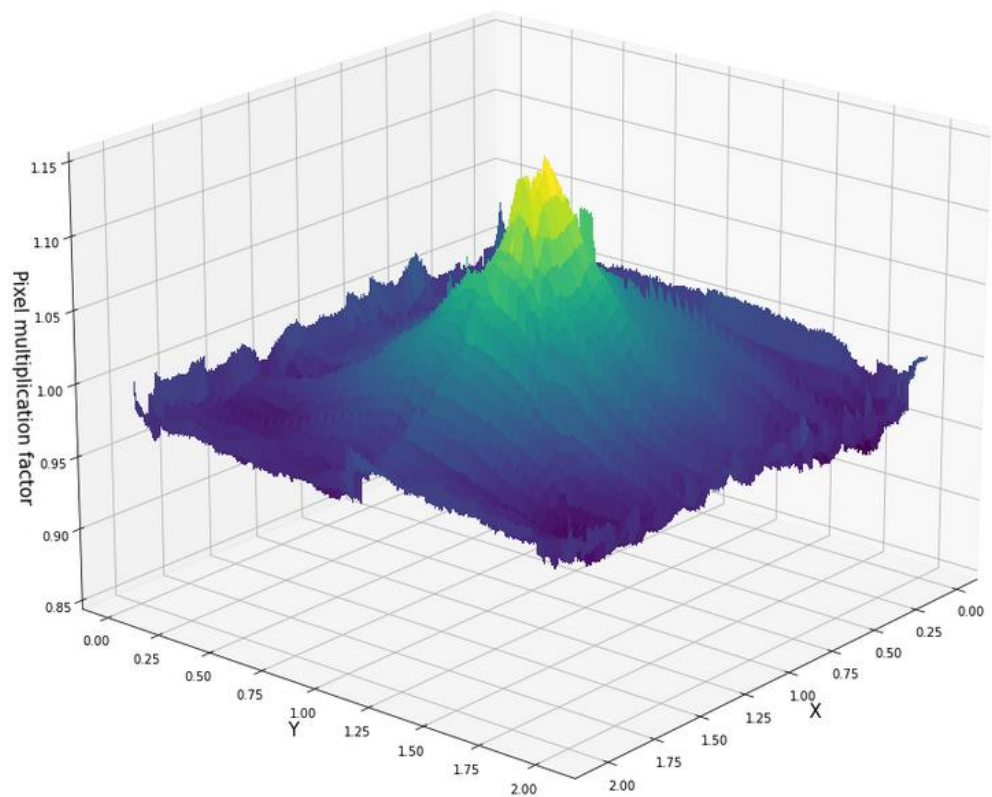
11-ID-C Pilatus



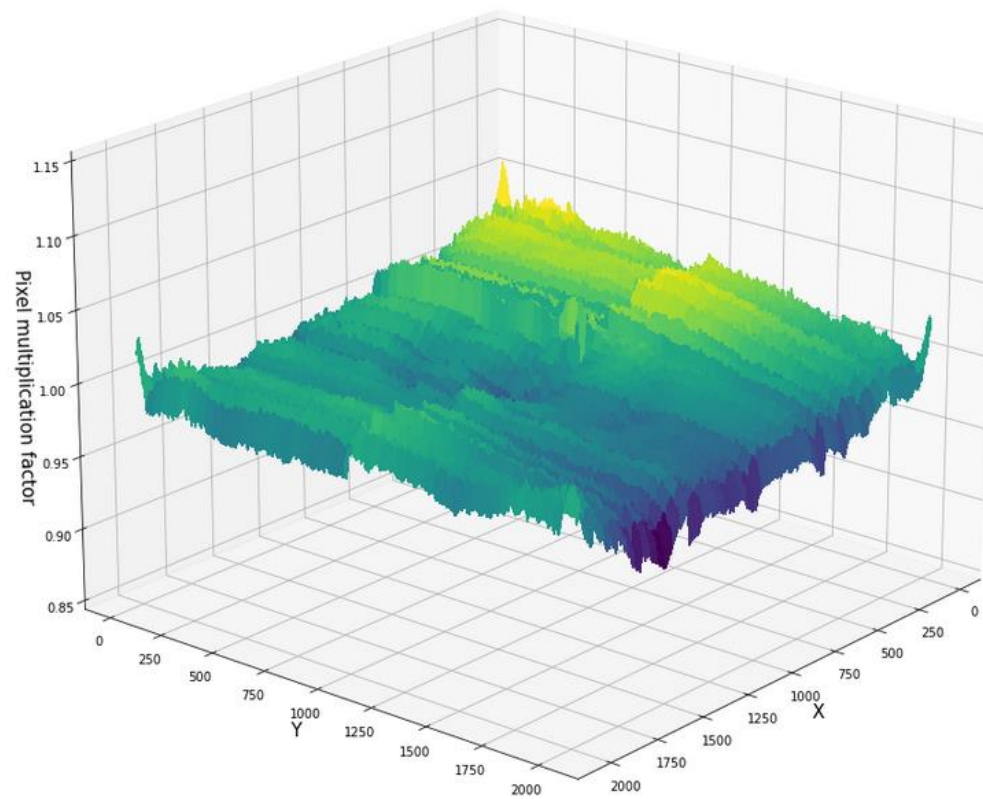
11-ID-C Perkin Elmer

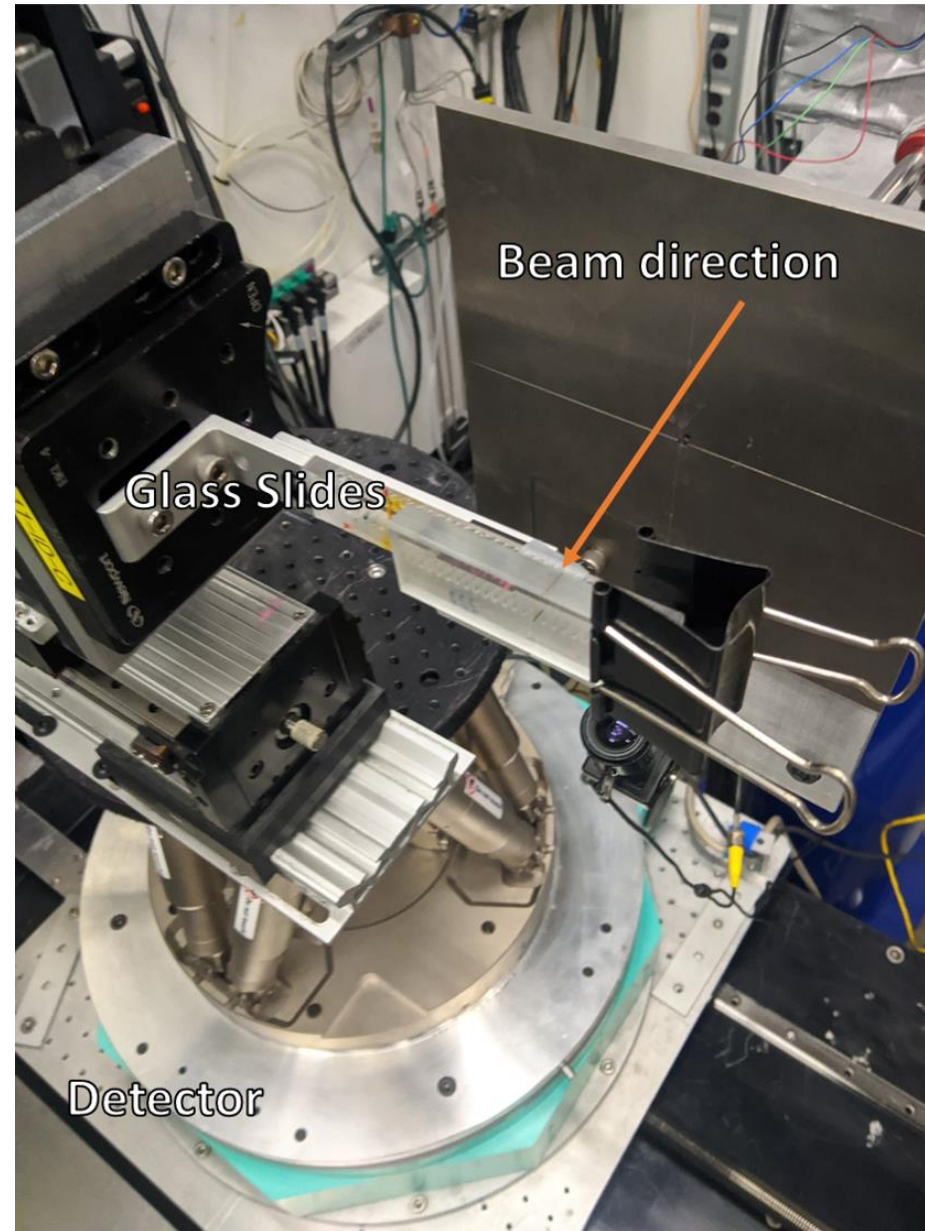


11-ID-B old Perkin Elmer



11-ID-C Perkin Elmer



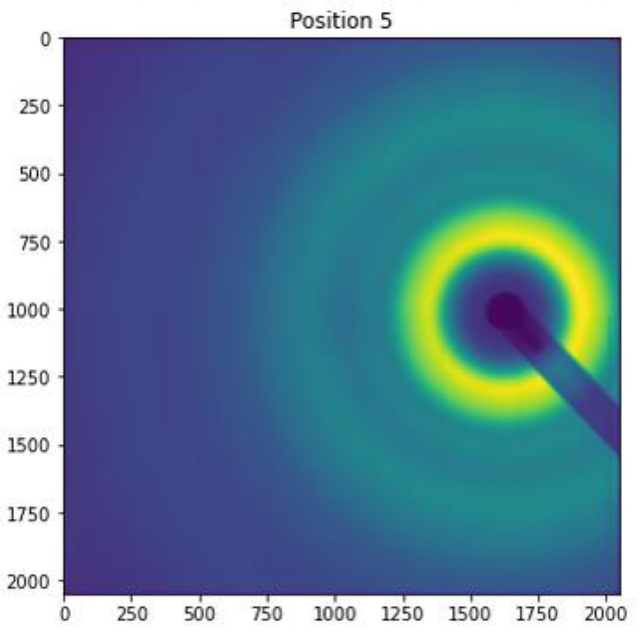
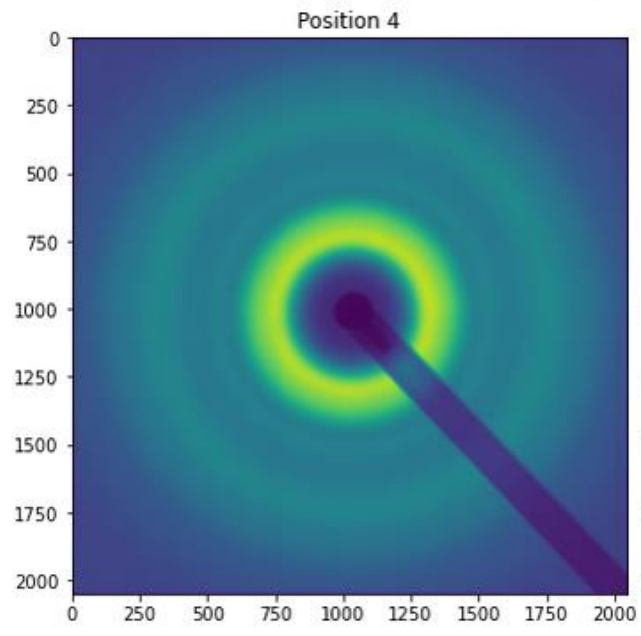
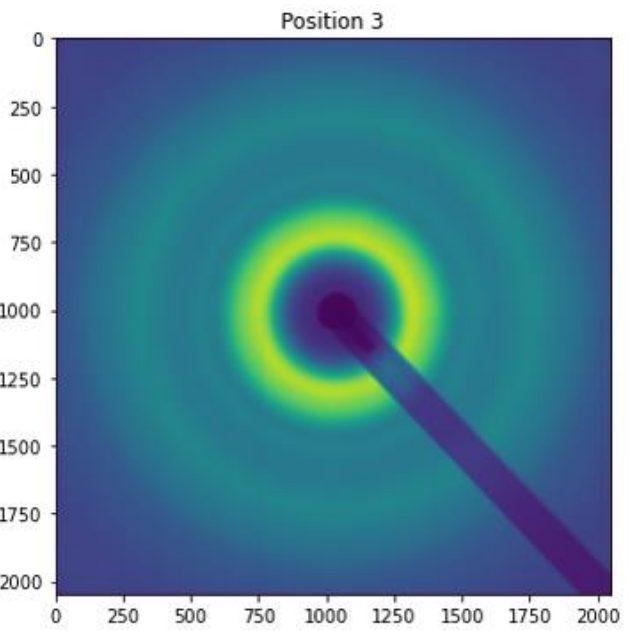
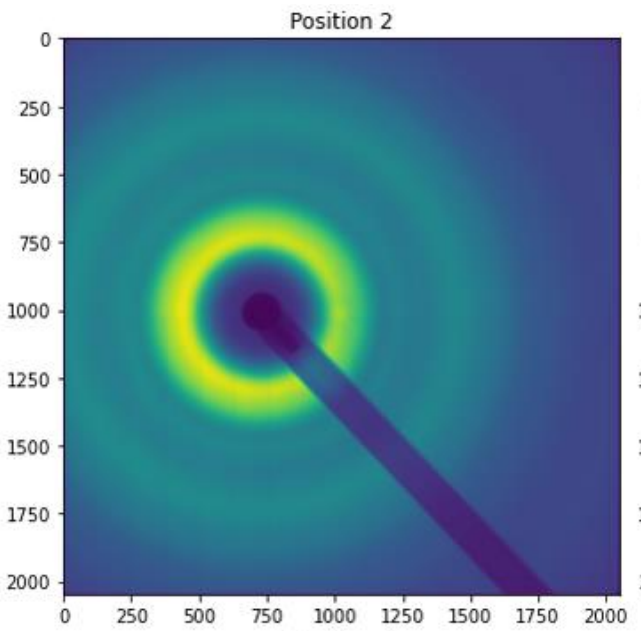
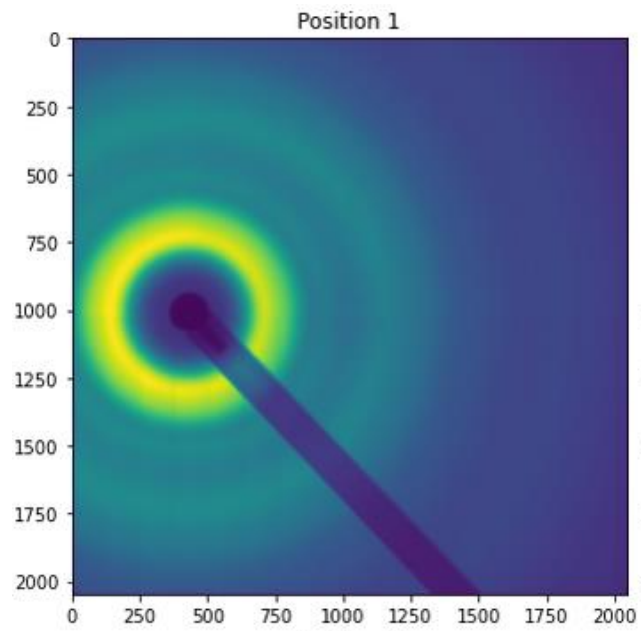


Beam direction

Glass Slides

Detector





https://github.com/jmsweng/X-ray-detector-gain-map

The screenshot shows a web browser displaying the GitHub repository page for 'X-ray-detector-gain-map' by user 'jmsweng'. The repository is public and has 1 branch and 0 tags. The file list includes folders for '11-ID-C', '17-BM', and 'Images', and files for '2 theta map.ipynb', 'Gain map calculation.ipynb', and 'README.md'. The README file is selected and its content is displayed below. The README content includes a title 'X ray detector gain map', a paragraph about the code being provided in Jupyter Notebooks, a bolded statement that documentation is a work in progress, and a paragraph describing the code's purpose in calculating gain maps for x-ray area detectors.

Repository details: jmsweng / X-ray-detector-gain-map (Public). 1 branch, 0 tags. 95 commits.

File/Folder	Action	Time
11-ID-C	Add files via upload	last month
17-BM	Add files via upload	last month
Images	Add files via upload	yesterday
2 theta map.ipynb	Add files via upload	4 days ago
Gain map calculation.ipynb	Add files via upload	4 days ago
README.md	Update README.md	yesterday

### README.md

## X ray detector gain map

Code is provided in the form of Jupyter Notebooks, installation instructions for Jupyter are found [here](#), though it is already included in the default install configuration of [Anaconda Python](#) which is the recommended Python install for this code. Code is written for Python 3 (Python 3.9.7 was used to run the code to generate example images).

**Documentation is currently a work in progress and may (though is not likely to significantly) change without notice**

Code to calculate a gain map for an x-ray area detector at an arbitrary beam energy using measurements of an amorphous scatterer. May be used to correct for non-uniformity in detector response at energies where it is not possible collect a flat field. May also be used to quickly measure a gain map to monitor and correct for detector degradation from exposure to radiation.

Repository statistics: No description, website, or topics provided. 0 stars, 1 watching, 0 forks. No releases published. No packages published. Languages: Jupyter Notebook 100.0%.