Optical Imaging to Improve Crystal Hit Rates with Serial Crystallography on I24

Laurence Cullen, Supervised by Danny Axford
20th of September 2016

Abstract

Over this 12 weeks project, two offline imaging techniques were tested in an attempt to identify crystal locations on a serial crystallography sample holder. A data collection setup was built in addition to a software pipeline to process optical images and extract an intensity value for each cell in the sample chip. These values were compared to X-Ray diffraction data of the sample chips to see if correlations between optical images and crystal locations could be observed. A similar experiment was run using polarised light to see if the way that crystals altered linearly polarised light could be used as a good guide to their presence. Whilst the fitting and read out tools themselves were effective it was not possible to strongly correlate output from optical imaging and diffraction data both with and without polarising filters.

1 Introduction

This project is looking to improve pre-experiment steps for serial crystallography in which X-Ray diffraction data is collected from thousands of individual crystals. These crystals are at random orientations with the idea being that good coverage of the full rotation space can be obtained. This compares to the classic rotation experiment where a single crystal is exposed to X-Ray beam and is slowly turned on a goniometer, diffraction data being collected at a large number of angles with fine angle resolution.

The classic rotation experiment requires a crystal to be exposed for several seconds meaning any reaction taking place on or below this time scale is out of reach. With serial crystallography and the new FEL, free electron lasers, time resolutions of 10^{-15} seconds are possible. Snapshots can be taken of reactions part way through to greatly increase understanding . The idea is that the reaction is triggered at exactly the same time before each crystal takes beam so that all the X-Ray diffraction data is of the same intermediate species.

Another advantage of serial crystallography with femtosecond pulses is that the problem of radiation damage is removed as the pulse has such a short duration that the crystal has no time in which to get destroyed before the entire pulse has passed through. This is known as diffraction before destruction, the term and technique coined in [1].

Due to the advantages of FEL, techniques to optimise sample delivery for them are currently receiving more attention. The first attempts involved a high-speed jet of fluid with small crystals mixed in. The FEL was repeatedly triggered with the hope of hitting the crystals suspended in the fluid jet. This was not an optimal technique as there was a high degree of waste, lots of crystals were used and there were large numbers of FEL exposures which did not connect with a crystal.

A new technique is being tested on I24 at Diamond Light Source by filling a sample "chip" which has 11,664 holes or "cells" and dimensions of 2cm by 2cm with as many individual crystals as possible. This is then placed into the X-Ray beam and diffraction data is taken for each cell which is later combined in an attempt to solve the molecules structure. The location of every potential crystal is known and the bigger problem this project addressed is that some of the cells are filled and others not. If this can be determined in advance then time will be saved at all stages of the data collection processing steps.

2 Theory

Due to the way that the sample chips are loaded with crystals, the number of cells with good crystal candidates is often well below 100%. To reduce overheads in terms of data processing and collection time it would be helpful if prior to X-Ray exposure the cells on the sample chip which contained good crystal candidates could be distinguished from cells with bad candidates. This is complementary to previous serial crystallography pre-screening work [3] using UV spectrometry to identify crystal locations on the same type of sample chips used in this experiment.

This project tested two additional ideas for detecting the presence of crystals on sample chips. Firstly the idea that light from a background source passing

through the sample chip would be more strongly attenuated when crystals were present and that examining the emission per cell before and after a chip was loaded with crystals could give clues as to whether good crystal samples were present in that cell.

The second idea used a linearly polarised light source with a polarising filter in front of the camera aligned so that all the emission from the background light was blocked. When a sample chip was placed in front of the background light the birefringence of the crystals present in the sample chip would distort the polarisation of the background light so that it would no longer be cancelled out by the filter in front of the camera. This would increase the quantity of light detected by the camera for cells with crystals present. To test the efficacy of these methods predictions of crystal locations were compared to X-Ray diffraction data for each cell to see if significant correlations existed.

The motivation for investigating these techniques is that they are very quick and images can be taken in a minute or two compared to 10+ minutes for other scanning techniques.

3 Method

3.1 Data Acquisition Setup

Figure 1 shows the setup used for collecting images of sample chips, a magnetic sample chip holder was chosen to ensure that captured photos are all closely aligned. The light source has a linearly polarising filter in front of it with another polarising filter before the Mako G Camera, this filter can be rotated so that the effect of crystals on the purely polarised background light can be observed. For most of the project, images were taken without polarising filters and just the effect on the overall light level was observed.

Unfortunately due to beam time scheduling it was only possible to collect data during the first and second last week of the project so there were limited opportunities to iterate on the data collection techniques.

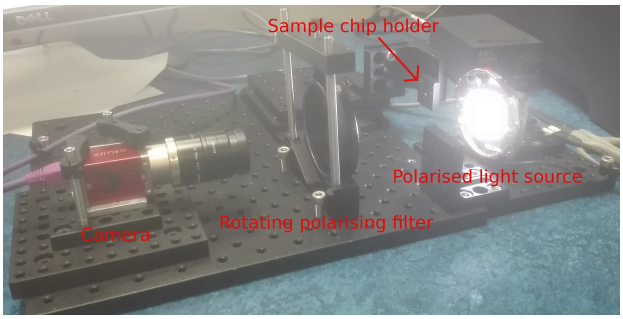

Figure 1: Image collection setup, sample chip holder is magnetic and clips onto chip holder

3.2 Fitting Problem Setup

The problem of extracting how much light was coming through each cell in the sample chips and how this varied depending on the presence of crystals was complicated by several factors. Firstly due to the manufacturing process the holes in the sample chip are of different sizes so that even with no crystals present there is a large difference in the amount of background light coming through. To defeat this effect images were captured before and after the chips were loaded with crystals and the differences were divided out so that the effects of the size of each cell could be removed.

Another problem was the misalignment of the chips, random translations in x and y of a certain amount were observed in photos depending on how the chips were mounted in the sample holder attached the magnetic clamp. Some variation in the rotation was also present, usually between 0 and 10 degrees but in some cases, the chips were flipped by 90 or 180 degrees complicating how the cells were indexed. Many of the chips also had significant gaps due to the fragility of their design and general wear and tear.

To account for the randomised nature of the chips within the captured photos a theoretical layout of the chip detailing the exact x,y position of each cell (figure 2) was taken and modified and masked over the photo. The parameters defining how the theoretical mask was applied included the pixel to physical size scaling, x and y translation and the angle of rotation. When all these parameters can be determined

to small errors the mask has been well fit and the summed pixels in a small radius around each theoretical cell location are returned in an indexed list giving the brightness value for each cell.

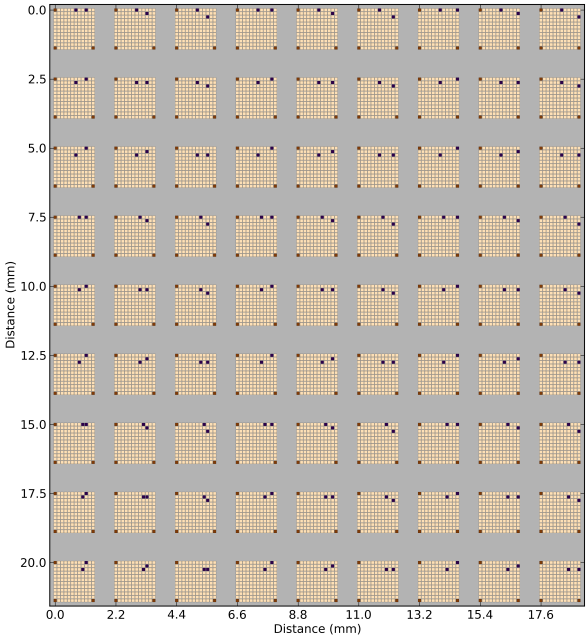

Figure 2: A schematic of the sample chip showing corner gaps in red and fiducial cells in blue, during manufacturing cells are not created here to allow identification of city blocks.

3.3 Rotation and Pixel Distance Ratio Extraction

Unfortunately finding a good mask fit is challenging as simultaneously fitting 4 parameters is computationally taxing and filled with local optimums and all attempts at pure brute forcing a solution ended up far too slow and ineffective to be useful.

Instead, a method was developed to extract the angle of rotation and pixel to real size scaling from features of the image. To do this the locations of the "city block" collections of cells as shown in the figure –x– were identified in a multi-stage computer vision algorithm. Firstly the image was converted to black and white then a Gauss blur was applied on the scale of the cells making up the city blocks so that the interior of the city blocks were close to homogeneous. An adaptive threshold binarization (using equation 1) was applied to the image which converted each pixel into a 1 if it was a certain fraction of the local mean of pixel values and to a 0 if below, the result of this process is figure 3.

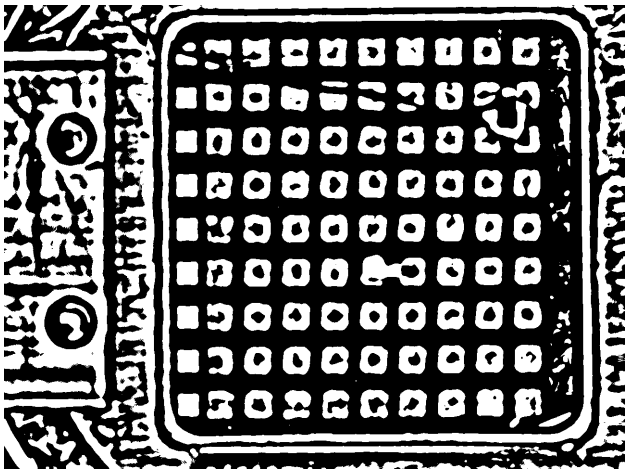

Figure 3: This image shows what the computer "sees" after the adaptive threshold has been taken, rectangles are fitted to the shapes derived in this step.

$$dst(x,y) = \begin{cases} 1\, if\ src(x,y) > T(x,y) \\ 0\, otherwise \end{cases} \tag{1}$$

Where dst is the destination image, src is the source image and $T(x,y)$ is the threshold value which is the mean of the pixels in the $blocksizexblocksize$ neighbourhood of (x,y) minus C, a modifier to the adaptive threshold value which can be tuned.

From this binarized image contours were calculated for each distinct object and as the city block are square, rectangles were fitted around the list of contours calculated. As the rough size of the city blocks in pixels are known, rectangles with areas too small and too large are discarded as they are unlikely to represent a clean city block identification. Also, rectangles with too great a ratio between their side lengths (strongly squashed or extended) were dis-

carded. This leaves a set of rectangles bounding all good rectangle determinations on the photo as shown in figure 4.

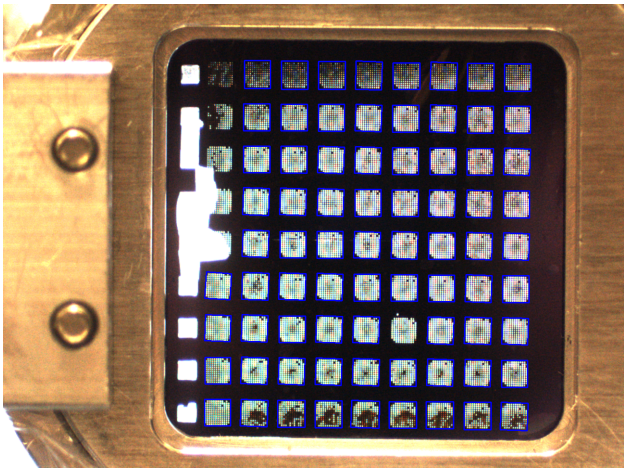

Figure 4: A sample chip example image with rectangles drawn around city blocks which have been confirmed by the feature extraction code.

From this collection of solved city block rectangles, central coordinates for each rectangle were determined. Each central coordinate point was compared to every other central point and an array was created which contained the 4 closest neighbours to each pair of coordinates, discarding values outside of a reasonable range of possible values for the close neighbours (horizontal) and far neighbours (vertical). This corresponds to the 4 closest city blocks, as their physical distance separation is known from the theoretical chip layout. To derive the best value of the pixel to distance ratio statistics of the neighbour population in the vertical and horizontal were taken. These yielded a mean and standard deviation which along with the population size gives the standard error, to get the best values possible the thresholding and feature recognition code was run in a loop with varying threshold parameters which caused different populations of city blocks to be identified.

Some thresholding passes dug deeper into the noise in the image and were able to identify more city blocks definitively but these were also more prone to

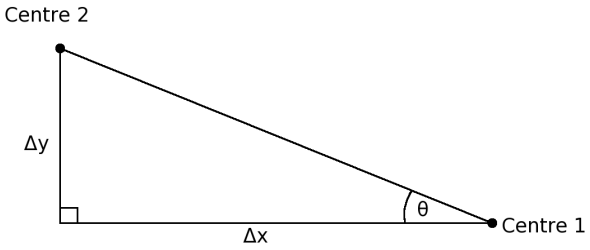

Figure 5: Diagram showing two adjacent proven rectangle centres next to each other and how the overall orientation θ is obtained.

giving slightly distorted values for the central position, this compared to the more conservative threshold passes which identified fewer city blocks but with a lower degree of error. The optimum thresholding values varied on each image depending on the light levels and the amount of excess protein floating around between city blocks which could make it harder to separate or "watershed" them. After each pass statistics for the pixel to distance ratio was recorded and the threshold pass which yielded the lowest standard error was taken.

A similar process was taken when calculating the rotation of the image, the vectors connecting city block centres to their neighbours had their angles calculated with simple trigonometry as shown in figure 5. The thresholding was evaluated to take into account both the rotation and pixel to distance ratio and simultaneously minimising the standard error in both.

After this information had been extracted from a photo the theoretical mask was overlaid over the photo and a range of x and y offset values were tested to find a tight match. The attempted fits were evaluated by trying to find the mask position that when convolved with the original image yielded the highest value.

3.4 Spurious Emission Removal

A problem encountered when cycling over x and y offset values was that the edges of the chip holder and gaps in the chip were illuminated more brightly

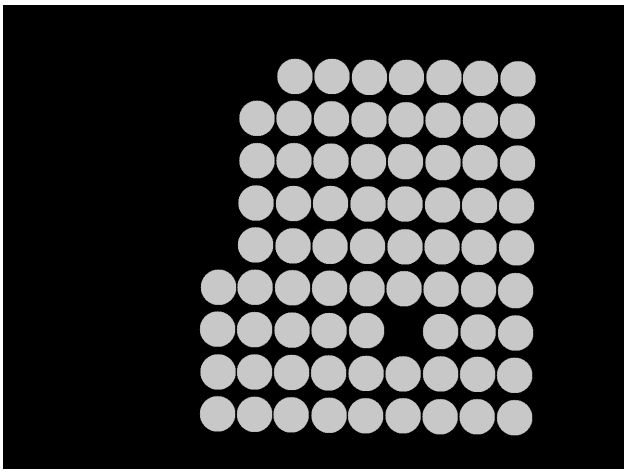

Figure 6: White areas are from around confirmed rectangles and will not have saturated pixels removed within them.

than the city blocks of cells that the fitting program was attempting a match with. Several approaches were used to tackle this problem.

Firstly all maxed out pixels (value of 255) were set to 0 as they were almost universally from gaps in the sample chip where light was shining directly through from the backlight. However, some of the cells also showed maxed out pixel values which we did not want to erode. To counteract this a mask was created which was positive in the area around each proven city block as shown in figure 6, if a pixel was equal to 255 whilst being in the vicinity of a proven city block its value would not be set to 0 whereas anything off a city block would be removed as before.

Another problem was reflected light from the sample holder where the data was collected which was not so easy to remove as it often did not saturate pixel values in the camera. To deal with this a new image was derived which was the local standard deviation of the original image. The rationale behind this process was that the emission from the sample chip holder had very small amounts of local variation in brightness, thus the local standard deviation of this area would be very low compared to on city blocks where the variation varies very strongly locally moving on and off cells in close proximity to each other.

The local standard deviation map was calculated by evaluating a pixel in the new image as the value of the standard deviation of pixels in the local area (5x5 box of pixels). This standard deviation map was binarized such that every pixel with a value equal to or above the 78th percentile of the overall pixel value population was set to 1 and everything else 0. The 78th percentile was optimised by extensive testing, it was found to yield the best balance of signal and noise.

The standard deviation map was effective at picking out city blocks compared to reflected light from flat areas of the sample holder, however, it also picked up significant amounts of noise from free floating pieces of loaded sample on the sample chip and showed large values at the transition zones between the sample chip and the sample holder. These features were fairly constant so they were possible to remove largely, contours were taken off the binarized standard deviation map and any contour containing a very small area or very large perimeter was removed. This was effective at removing the border with the sample holder (very large perimeter and low area contour) and the small flecks of excess sample (contours with a very low area). An example of how effective this contour cleanup was can be seen in figures 7 and 8.

3.5 Sweep Generations

The strategy when searching in the x,y translation space was to fit chip mask so that each cell in the model mask is aligned to the cells in the image in addition to having each city block matched. Thus sweeps were set up to sweep at a small fraction of each of these space scales. First, the cells were aligned, then the city blocks and finally the mask was moved by the distance between the city blocks. During this stage instead of masking the original image, the mask was convolved with the binarized standard deviation maps as the fitting routine had a tendency to move the mask towards bright parts of the sample holder. However when iterating over an entire city block some misalignment's tended to be introduced so that further fine fitting was required after the city block by

city block fitting on subcell dimensions.

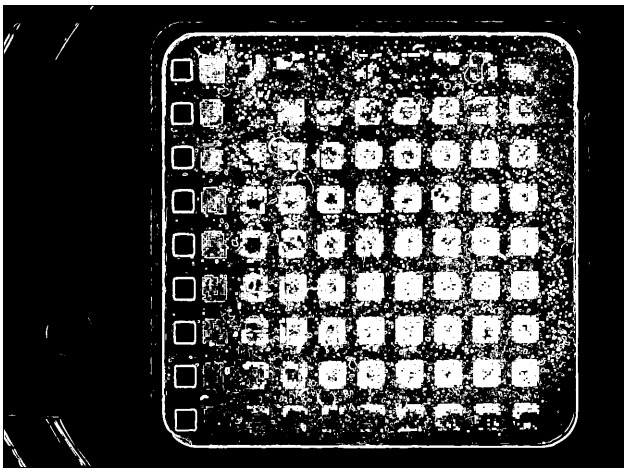

Figure 7: Standard deviation map derived from original image.

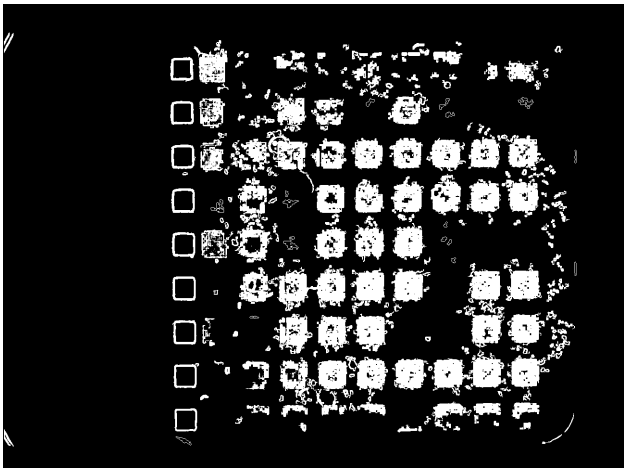

Figure 8: The same standard deviation map as figure 6 after spurious objects have been cleaned.

3.6 Read Out

When the final mask fit parameters had been determined a routine went through cell by cell summing the values of the pixels in a small radius around each defined cell location from the chip mask. This yields a list of values in order of cell id with the integrated pixel values.

4 Results

4.1 Fitting

The final fitting algorithm was highly effective at matching sample chips even for badly lit samples and those with large amounts of excess sample. Time limitations meant that only chips within around 40 degrees rotation of the chip mask as my fitting program would have required significant reconstruction to be able to handle these situations. As some chips were loaded at 90 and 180-degree rotations some data could not be processed.

(fit_plot.py) Code linking pipeline elements together when comparing chips before and after loading to see relative changes in light transmission per cell. This is setup for imaging without polarising filters

(polar_fit_plot.py) Code linking pipeline for chips imaged with the polarising filter setup, fitting is performed on chips where the background light is allowed through so the mask can be correctly located. The mask fit parameters are saved and applied to the frame where the background light has been largely removed by the polarising filter for readout purposes.

An example of a well-fitted mask after read out is shown in figure 9 and how it compares to the original image in figure 10. Note that the mask has managed to determine effectively the rotation of the sample chip in the original image in addition to identifying and removing spurious emission from light coming through damaged parts of the ship.

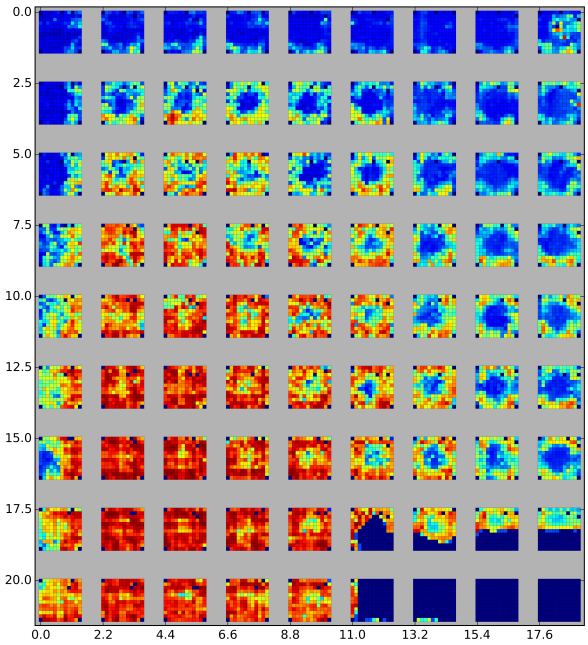

Figure 9: The read out of summed local pixel values around each fitted cell position, chip called Zurich.

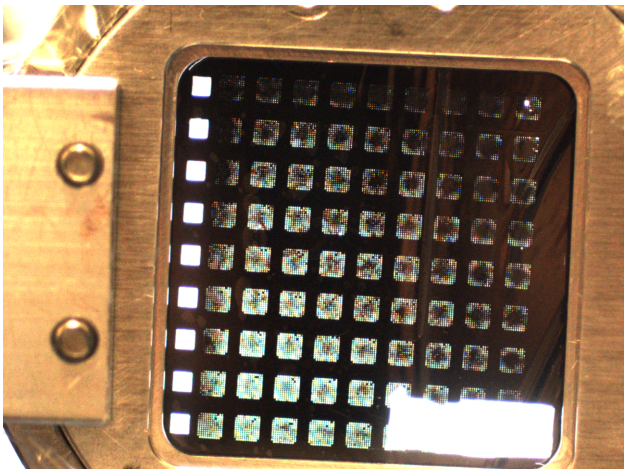

Figure 10: Original sample chip image mask was fitted to, Zurich.

4.2 Comparison with Diffraction data

4.2.1 Non Polarised Imaging

Shown in figures 11 and 12 are X-Ray diffraction data showing good crystal candidates and the extracted per cell brightness computed using my fitting program respectively.

4.2.2 Polarised Imaging

Figures 13 and 14 show examples of images taken with a polarising filter and the good crystal candidates from diffraction data on the same sample chip.

5 Conclusions

This project has demonstrated the feasibility of fitting and reading out brightness information from photographs of sample chips being tested for use in serial crystallography on the I24 beamline at Diamond Light Source and the XFEL facility at SACLA in Japan. Correlations between brightness and proven crystal locations from X-Ray diffraction data were not clearly observed for images taken with a Mako G camera and a simple imaging setup but images taken with optical microscopes showed much greater promise. Time constraints meant that microscope imaging could not be well explored as it required a new imaging setup and a process to stitch multiple images together as the field of view of the microscope was too small to image the entire sample chip at once. This would be worth further attention in the future as it offered a far higher level of image quality and spatial resolution.

In the future trying to find another type of transparent film that had less of an effect on the polarisation of the background light than Mylar whilst still holding in the protein sample on the sample chip would be useful. The effect of Mylar added significant noise to the images taken with polarising filters.

Whilst my fitting tool was effective at angles of orientation between 30 and -30 degrees for sample chips it was not built to handle orientations outside of this. To make the system more flexible in the future a feature could be added that would attempt to apply the fitting mask at 90, 180 and 270-degree rotations to attempt to correctly handle extreme rotations. This

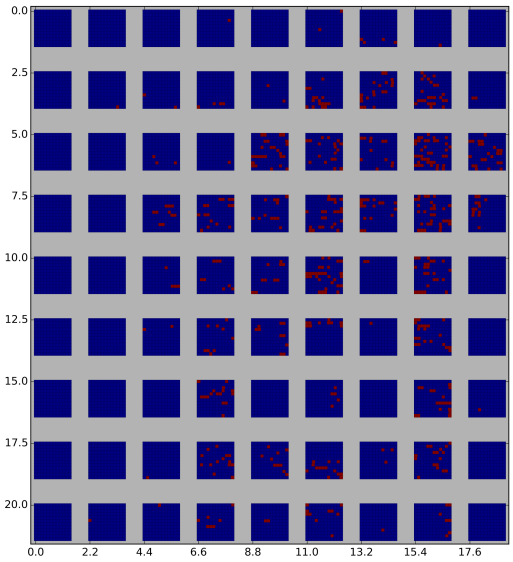

Figure 11: Good crystal candidates determined from X-Ray diffraction data taken for each sample chip cell, sample chip name Yamoto.

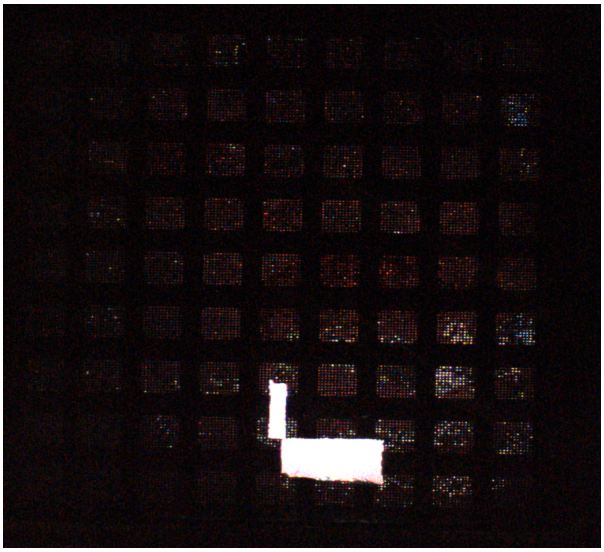

Figure 13: An optical image taken with a polarising filter in front of the camera and with purely polarised back lighting of the sample chip Dallas.

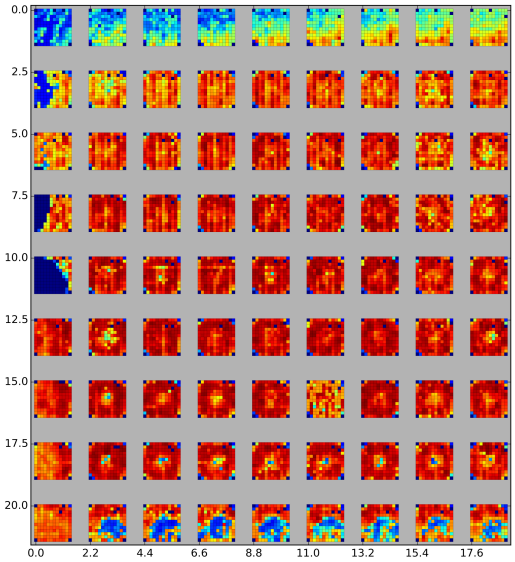

Figure 12: The plotted readout values from Yamoto.

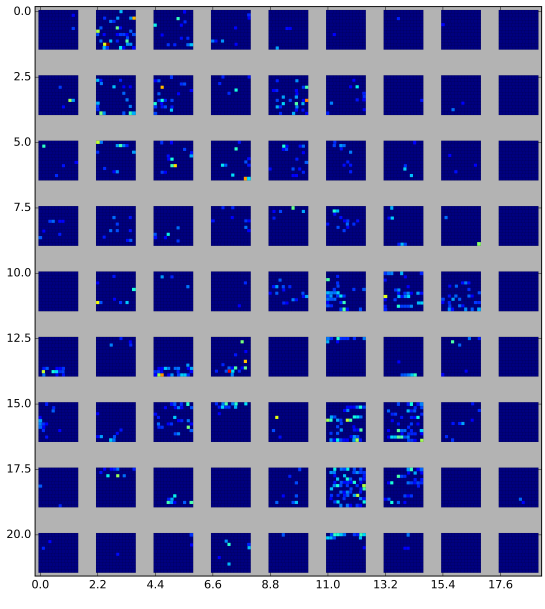

Figure 14: Good crystal candidates on the sample chip Dallas from diffraction data.

was not initially a development priority as most of the data initially collected only had modest rotations from 0 degrees.

References

[1] Henry N. Chapman, Carl Caleman, and Nicusor Timneanu, Diffraction before destruction. *Philosophical Transaction of the Royal society B* (2014)

[2] Mark S. Hunter, Brent Segelke, Marc Messerschmidt, Garth J. Williams, Nadia A. Zatsepin, Anton Barty, W. Henry Benner, David B. Carlson, Matthew Coleman, Alexander Graf, Stefan P. Hau-Riege, Tommaso Pardini, M. Marvin Seibert, James Evans, Sébastien Boutet , and Matthias Frank, Fixed-target protein serial microcrystallography with an x-ray free electron laser. *Nature Scientific Reports* (2014)

[3] Oghbaey S1, Sarracini A1, Ginn HM2, Pare-Labrosse O1, Kuo A3, Marx A4, Epp SW4, Sherrell DA5, Eger BT3, Zhong Y4, Loch R4, Mariani V6, Alonso-Mori R7, Nelson S7, Lemke HT7, Owen RL5, Pearson AR8, Stuart DI2, Ernst OP3, Mueller-Werkmeister HM1, Miller RJ1, Fixed target combined with spectral mapping: approaching 100% hit rates for serial crystallography. *Acta Crystallographica Section D* (2016)

6 Appendix

6.1 Code Guide

Here follows a brief run down of the purpose of the different scripts written as part of this project. Most are in Python however for some tasks Bash was more appropriate.

6.1.1 reduc

Main analysis code which does the mask creation and fitting operations, includes a deprecated top section (commented out) which employs an alternative method to determine the orientation of the sample tray using 2d Fourier transforms. The top level function is meta_sweep() which pulls together the other lower level functions to execute the multiple sweep generations and call the scripts which create all the image derivatives necessary to get a good fit.

6.1.2 watershed

Named after the watershed problem where a series of objects are separated from each other, this script identifies the city blocks and their central locations. From this, it derived the pixel to mm ratio of the image and determines the orientation from how the city block centres are positioned compared to their neighbours.

6.1.3 variance_map

Computes the local standard deviation map of the of an image highlighting features which change on a small scale and wiping out those constant on large scales. Used to counter the reflection of light from the metal edges of the sample holder which could often outshine the light coming through the sample chip from the backlight. The output of this code was used in one of the sweep generations to ensure spurious emission was not being fitted for.

6.1.4 visual_map

This code (Credit Darren Sherrell) marks the locations of each of the cells on the schematic mask in physical space, this was taken by reduc in order to generate the mask which was applied to the captured photos.

6.1.5 spot_map

Called from ./grid_plot.sh this took the diffraction data information for a chip and plotted as shown in figure 11.

6.1.6 polar_plot

Used to plot fittings of images taken with a polarising filter (warning incomplete).

6.1.7 polar_fit_plot

Brings together all the listed core analysis scripts to fit a well lit polarising filter image and then uses that fitting information for the badly illuminated frame to maximise the chance of an effective fit.

6.1.8 intensity_plot

Plots the summed local pixel intensities per cell calculated from reduc as seen in figure 12 for sample chips before and after loading to create a difference map.

6.1.9 fit_plot

Pulls together the other lower level analysis routines to generate a difference map directly from two input photos assuming that it is the same sample chip before and after loading with crystals.