Stony Brook University

OFFICIAL COPY

The official electronic file of this thesis or dissertation is maintained by the University Libraries on behalf of The Graduate School at Stony Brook University.

© All Rights Reserved by Author.

Beyond FIT2D:
Calculating Intensity Errors for Data Analysis of X-Ray Synchrotron Powder Diffraction Data

A Thesis presented

by

Melissa Sims

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Master of Science

in

Geosciences

Stony Brook University

May 2014

Copyright by
Melissa Sims
2014

Stony Brook University

The Graduate School

Melissa Sims

We, the thesis committee for the above candidate for the

Master of Science degree, hereby recommend

acceptance of this thesis

Dr. Lars Ehm - Thesis Advisor
Research Associate Professor, Mineral Physics Institute

Dr. Robert C. Liebermann - Chairperson of Defense
Distinguished Service Professor, Department of Geosciences

Dr. Richard J. Reeder - Second Reader
Professor, Department of Geosciences

This thesis is accepted by the Graduate School

Charles Taber
Dean of the Graduate School

Abstract of the Thesis

Beyond FIT2D

by

Melissa Sims

Master of Science
in
Geosciences
Stony Brook University

Synchrotron x-ray diffraction is used to investigate mineral structures. Accurate analysis of the results is limited by intensity errors caused by the experimental setup, sample, and detector. These include high intensity spikes, dead pixels, beam stop shadow, single crystal spots, and poor powder statistics, and they all contribute to the background uncertainty within powder diffraction datasets. The errors must be either quantified or removed. Due to the intrinsically low peak-to-background ratios of measured intensities in experiments at non-ambient conditions, the intensity errors are usually overestimated in subsequent Rietveld refinement; leading to unrealistically low uncertainties in the refined structural parameters. FIT2D, the commonly used tool for synchrotron x-ray diffraction data processing, does not have methodology to solve some of these problems, and masking data is ordinarily manual and tedious. Our program automatically masks and addresses the data analysis problems by examining intensity uncertainties. In the program, data is sorted by two-theta value into bins during the integration process. The contents of individual bins are then statistically analyzed. The data is assumed to be Poisson distributed, and intensity values outside a user-specified interval of the standard deviation are rejected. The data is then normalized to account for variance in the number of pixels contributing to a particular measurement. The final software modules will be collected in a code repository and used for the data acquisition and analysis software package at the X-ray Powder Diffraction (XPD) beamline at NSLS-II.

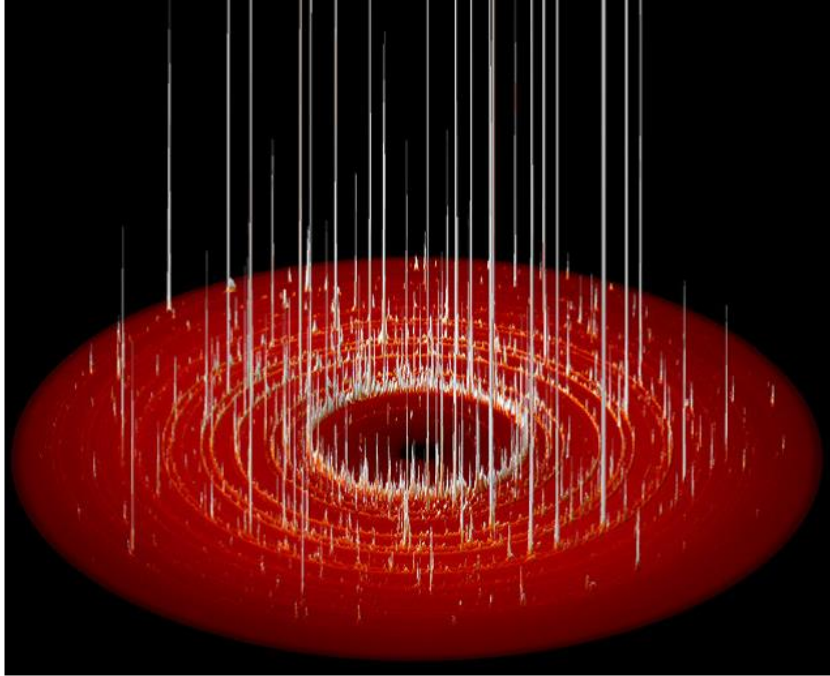

Figure 1: Rendered image of a 2-d round detector, with the intensities of individual pixels represented by height. The figure is from Hinrichsen [10]

Table of Contents

Contents

1 Introduction **1**

1.1 Synchrotron Powder Diffraction Data 1

1.2 Structural Refinement 2

1.3 Error Estimation 9

1.4 FIT2D 15

2 Methods **16**

2.1 Experimental Methodology 16

2.1.1 Components 16

2.1.2 Experimental Details 17

2.1.3 Sources of Error 19

2.2 Data Reduction Software 19

2.2.1 Read in Data and Background Correction 20

2.2.2 Geometry Calculation and Correction 21

2.2.3 Polarization Correction 22

2.2.4 Integration 23

2.2.5 Bin Analysis and Automatic Masking 24

2.2.6 Output 26

3 Results and Discussion **27**

3.1 Data Import 27

3.2 Data Integration 29

3.3 Automatic Masking 32

4 Conclusions **35**

5 Outlook **36**

5.1 Geometry Correction 36

5.2 Normal-Pareto 36

5.3 Fractile Filters 36

6 BeyondFIT2D Code **39**

List of Figures/Tables/Illustrations

List of Figures

1 Intensities iv
2 Structural Determination Techniques 3
3 Incident Radiation Sources 4
4 Methods of Calculating Residuals 8
5 A Plot of the Average Bin Value with the Individual Data it Derives from 12
6 Differences in Weighting Schemes 13
7 Comparison of the Fit of a Normal Distribution to a Gaussian 14
8 Comparison of the Fit of Normal-Pareto, Log Normal, and Normal Distribution 15
9 Image of Diffraction Setup with Path of the X-Ray Beam Indicated 16
10 A Raw Data Image 18
11 A Raw Data Image with Texture 18
12 Flow Chart for Image Processing 20
13 A Diagram of the Setup Geometry 21
14 Polarization Correction, Plotted 23
15 Change in Goodness of Fit with Bin Size 24
16 Comparison of Normal-Pareto and Normal Distribution 25
17 An Image of the Program BeyondFIT2D 26
18 Summed Histogram 27
19 Data Histogram 28
20 Background Histogram 28
21 Data Average Compared to Data Variance 30
22 Histogram for Two Bins 31
23 A Gaussian Distribution Fitted to the Bin Data 31
24 A Poisson Distribution Fitted to the Bin Data 32
25 Integrated Pattern After Two Types of Mask 33
26 Poisson Mask at Various Sigma Values 34

List of Abbreviations

DAC - Diamond Anvil Cell
R - Residuals
XRD - X-Ray Diffraction
XPD - X-ray Powder Diffraction

Acknowledgements

I would like to express my sincere gratitude to Dr. Lars Ehm for all your patience, encouragement, invaluable assistance, and unwavering support. I'd like to thank Dr. Robert Liebermann for your guidance, for sharing your vast experience, and teaching me your passion for the geosciences. I'd like to thank both of you for starting a program for minorities in mineral physics.

I'd like to thank my lab mates Adaire Heady and Ashley Thompson for your help over the years. The research would not have been possible without financial assistance from the Bridge to the doctorate program.

To those I love, you know who you are, and thanks for everything.

1 Introduction

1.1 Synchrotron Powder Diffraction Data

Synchrotron x-ray powder diffraction experiments at extreme conditions have intrinsically low peak to background ratios because of the nature of the experimentation. Specific components are required to achieve the conditions necessary for a successful experiment and produce useful data. These components create problematic data statistics. The sample, diamond anvil cell, sample holder and the X-ray source can all contribute to the low signal to noise ratio. In these experiments, an x-ray beam is diffracted through a powder sample, consisting of hundreds of crystallites with random orientations, contained by a sample holder. The diffracted beam is then captured by a 2-d detector. In a normal single crystal sample, diffraction occurs where a set of crystal planes have the correct spacing to fulfill the condition for diffraction, Bragg's Law (Equation 1).

$$n\lambda = 2dsin\Theta \tag{1}$$

where n is a value, lambda (λ) is the wavelength of the light, d is the spacing between planes and theta (θ) is the angle of the diffracted light. In powder diffraction experiments, few crystallites diffract at any given angle because diffraction occurs only where planes in the crystallites are correctly aligned. The problem is exacerbated because the sample sizes are minute. While the peak to background ratio would benefit from larger samples, size is limited due to the experimental setup. Also, sample thickness contributes to sample adsorption.

For high pressure diamond anvil cell experiments, intensity counts can be as low as a few tens or hundreds of counts. For powder diffraction data from a one-dimensional tube type detector, up to 10 percent errors may be possible because of low intensities for experiments with short observing times. The experimental setup utilizes diamond anvil cells to create high pressure conditions for experimentation. When the beam passes through the diamonds, some adsorption occurs. Diffraction from the diamonds also creates single intense bright reflections. These reflections can dominate the diffraction pattern and combined with the low peak-to-background ratio create a pattern

that does not follow common crystallographic distributions.

Diffraction experiments using synchrotron radiation have several advantages. The source is capable of a pulsed time structure, allows the user to select the wavelength and has high polarization. It also has a very collimated, high brightness beam with a small spot size that allows for better resolution. The beam intensity is many orders of magnitude higher than X-rays produced in conventional X-ray tube sources, which is advantageous for weakly scattering crystals. Synchrotron intensities can reduce the error to around 3 percent. However, synchrotron data is generally captured on a two dimensional detector, and so has very different statistics compared to one-dimensional powder data and single crystal data. Each angle for powder diffraction datasets is measured by multiple pixels. Therefore, data must be integrated to produce a single value for each angle. This is advantageous because multiple measurements allow for more accurate intensity measurement. Better statistical analysis of datasets can be produced, provided the number of pixels contributing to a particular measurement is accounted for [4].

1.2 Structural Refinement

X-ray powder diffraction is applied in order to produce data for determination and refinement of the atomic structure of crystalline materials. There are several available methods. Different procedures may be followed depending on the purpose of the experiment. Some of these procedures are seen in Figure 2.

Structure determination consists of finding a periodic atomic arrangement that would produce intensities that fit the intensities of the experimental diffraction pattern. In powder diffraction, many factors affect peak intensities, which include the structure factor, multiplicities, lorentz factor, polarization factor, temperature factor, absorption by the sample, preferred orientation and extinction coefficients. Equation 2 is the calculation intensity at a particular step. The I_{calc} is the calculated intensity at a step, s is scale factor, hkl are Miller indices, L is the multiplicity, Lorentz, and polarization factor. F is the structure factor, φ is the reflection profile funtion, P is the preferred orientation, A is absorption factor. I_{back} is the background function. The diffraction pattern is deconstructed in terms of these effects.

$$I_{calc} = s \sum L_k |F_k|^2 \varphi(2\theta_i - \theta_k) PA + I_{back} \tag{2}$$

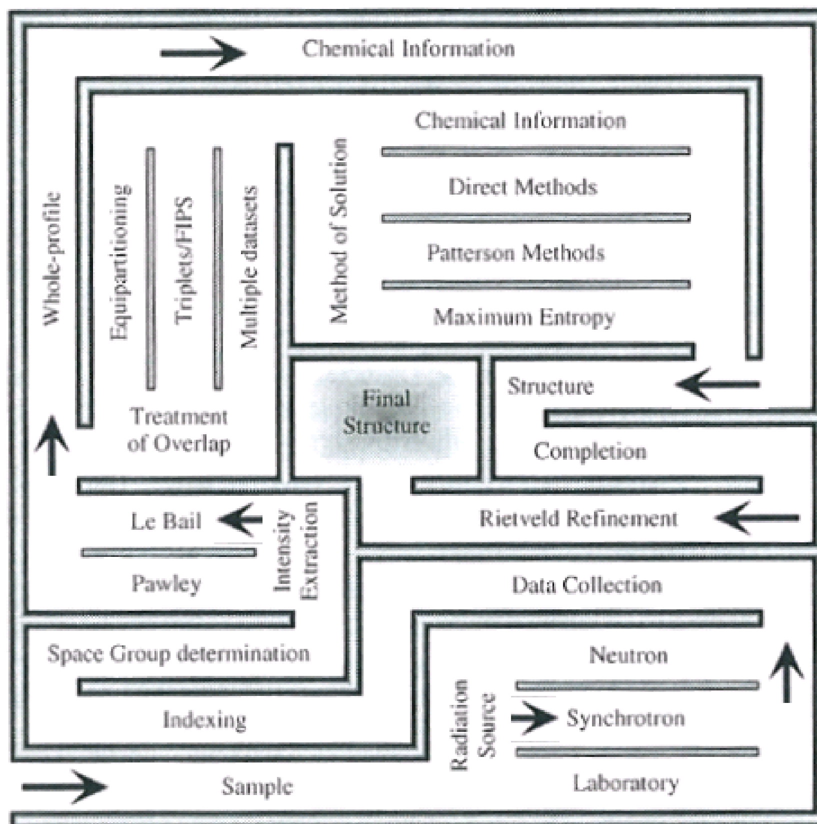

Figure 2: A figure modified from David [5] describing different methods of data collection and analysis to determine structural information from diffraction data. The method used for this paper is outlined using black arrows. The ideal methods to follow are sample dependent.

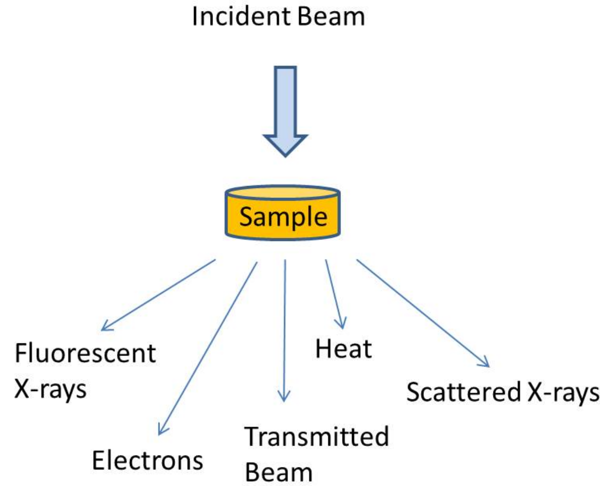

Figure 3: During the process of passing through a sample, an incident beam is diffracted. In addition to useful diffracted x-rays and the unaffected transmitted beam, electrons and other radiation forms are produced. These include heat, florescent x-rays, and incoherent scattered x-rays, and electrons. Some of these forms contribute to the noise present in the image due to capture by detectors. This figure is adapted from Henry [9].

The most important effect for structural determination is the structure factor, which defines how the atomic structure affects an incident beam. It is defined as Equation 3.

$$F(hkl) = \sum_{i=1} f_i[cos2\pi(hx_i + ky_i + lz_i) + isin2\pi(hx_i + ky_i + lz_i)] \quad (3)$$

In this formulation, F is the amplitude of the beam scattered by all atoms in the unit cell. h, k, and l are Miller indices. x, y, and z are the fractional coordinates for a particular atom. See Young [18] for further details. Individual intensities result from the diffraction and interference of the beam with different individual and planes of atoms. In powder diffraction measurements, individual intensities consist of the superimposition of reflections that have the same d-spacing. The multiplicity, the number of symmetry equivalent reflections contributing to an intensity, must also be accounted for in order

to model the pattern. In addition, several thermal properties can have an effect. Temperature vibrations, accounted for by the Debye-Waller factor, cause changes in the unit cell which increase background scattering and decrease diffraction intensities. The structure itself can produce extinction, which reduces or eliminates intensities due to destructive interference within crystals.

Initially, the pattern must be indexed in order to find the unit cell. The unit cell is refined using the LeBail method. It is a whole pattern profile fitting method used to extract intensities (I_{hkl}) suitable for determination of the atomic structure of a crystalline material and for refinement of the unit cell from powder diffraction data. In general, the extraction of intensities from powder diffraction is complicated by overlapping diffraction peaks with similar d-spacings. Knowledge of the unit cell and the approximate space group of the sample are required to successfully utilize the Le Bail method. The algorithm refines the unit cell, the profile parameters, and the peak intensities to match the measured powder diffraction pattern. The structural factor and its associated structural parameters are not considered in this type of analysis. It can be used to find phase transitions in high pressure and temperature experiments. It is generally used to provide a quick method to refine the unit cell, which allows better experimental planning. Le Bail analysis gives a more reliable estimate for the intensities of allowed reflections. Le Bail analysis fits parameters using a steepest descent minimization process, least squares analysis. This is an iterative process. The parameters being fitted include unit-cell parameters, the instrumental zero error, peak width parameters, and peak shape parameters. First, the Le Bail method defines an arbitrary starting value for the intensities (I_{obs}). This value is ordinarily set to one, but other values may be used. While peak positions are constrained by the unit cell parameters, intensities are unconstrained. To calculate intensities Equation 4 is used.

$$I_{obs}(a) = \sum \frac{(y_i(obs) \times y_i(a))}{y_i(calc)} \tag{4}$$

In Equation 4, $y_i(obs)$ is an observed profile point, $y_i(a)$ is a profile point on a particular peak, and $y_i(calc)$ is the calculated peak profile point. A single intensity value may contain more than one peak. Other peaks may be calculated using the same formula. The final intensity for a peak is calculated as $y_i(calc) = y_i(1) + y_i(2)$. The summation is carried out over all contributing profile points. The summation is known as profile intensity partitioning,

and it works over any number of peaks. Le Bail technique works especially well with overlapping intensities since in this method the intensity is allotted based on the multiplicity of the intensities that contribute to a particular peak. The calculated values are biased due to the choice of starting values. The process continues by setting the new calculated structure factor to the observed structure factor value. The process is then repeated with the new structure factor estimate. The unit cell, background, peak widths, peak shape, and resolution function are refined. This improves these parameters. The structure factor is then reset to the new structure factor value. The Rietveld refinement process follows.

Rietveld refinement fits the whole pattern by refining a structural model. The atomic positions, disorder or mixing between atomic sites, lattice parameters, profile parameters such as peak shape, and background parameters are fitted. Like LeBail and Pawley methods, it is a pattern decomposition technique. A plausible background is generally added as a function to the calculated result. During Rietveld analysis, first all intensities are assigned to particular Bragg reflections. Individual intensities are the sum of the contributions of many reflections. So, the diffraction pattern may be considered a collection of individual peak profiles, which consist of a height, peak position, and breadth. Each peak also has a tail, which tapers off away from the peak centroid. The integrated peak area is proportional to the square of the structure factor. Determining a peak shape requires care, since the profile contains contributions from the instrument and from the sample. Many functions have been used to find the reflection profile, for more detail see Young [18]. The structure refinement that follows is a separate process. According to Young [18], ordinary modeling for scattering patterns can be accomplished by deconvolving instrumental effects and the background contribution resulting from incoherent scattering, air scattering, and thermal diffuse scattering from the pattern. A background function is added to reduce errors in the model. It may be phenomenological or based on more sophisticated refinable models for actual features. The phenomenological model can consist of operator supplied linear interpolation based on data points selected by the user. This would be produce a high order polynomial. The background may be removed through Fourier filtering or direct modeling with a sine series. However, it is usually simply fitted as a parameter of the model. Amorphous scattering from the sample container can be encompassed in the background function. However, background functions can fail to encompass some crystalline scattering components, such as thermal diffuse scattering. They would

appear in the form of broad oscillations on the Bragg pattern. Poor crystallization in samples and separate phases that appear in the sample also cannot be calculated as a background function [18].

Rietveld analysis uses least squares analysis to find a best fit to intensities at each step by minimizing the residuals. The Residual can be quantified as:

$$R_p = \frac{\sum |(y_i(obs) - y_i(calc))|}{\sum y_i(obs)} \tag{5}$$

where $y_i(obs)$ is the observed gross intensity at i_{th} step. $y_i(calc)$ is the calculated gross intensity at the i_{th} step. There are other meaningful residual calculations. Figure 4 contains a summary.

Goodness of fit: R's and χ^2

$R_F = \frac{\sum \left| I_K('obs')^{1/2} - I_K('calc')^{1/2} \right|}{\sum I_K('obs')^{1/2}}$	R-structure factor
$R_B = \frac{\sum \left| I_K('obs') - I_K('calc') \right|}{\sum I_K('obs')}$	R-Bragg factor
$R_p = \frac{\sum \left| y_i(obs) - y_i(calc) \right|}{\sum y_i(obs)}$	R-pattern
$R_{wp} = \left\{ \frac{\sum w_i (y_i(obs) - y_i(calc))^2}{\sum w_i (y_i(obs))^2} \right\}^{1/2}$	R-weighted pattern
$\chi^2 = \left(\frac{R_{wp}}{R_{exp}} \right)^2$	Reduced Chi-squared

Figure 4: A table of the equations for different methods of calculating the residuals. Each calculation has a different meaning and bias. I_k*(obs)* and I_k*(calc)* are the observed and calculated intensities, respectively. w_i is the weighting scheme. R-Bragg and R-structure factor are biased in favor of the model, and the numerator of R-weighted pattern contains the actual residual being reduced. The square root of the reduced chi-squared function is also known as the goodness of fit. The table is from Young [18].

Least squares minimization produces a set of normal equations that include intensity derivatives based on adjustable parameters. The set of equations can be solved by inverting the matrix of normal equations. This is an iterative procedure consisting of applying small shifts to starting parameters, because the solution is non-linear. The initial model must be close to the correct solution, since otherwise solutions that are local minima instead of the global one may be found. Local minima may be eliminated by combining data from different sources, such as x-ray and neutron, by using multiple least squares algorithms, or by applying an established best practice refine-

ment strategy. To judge the accuracy of the fit, residuals may be used. The most direct one is the R_{wp}, the function in which the numerator actually contains the residual being minimized. This is seen in the equation for R_{wp} in Figure 4. R_b is the Bragg residual and is based on intensities predicted only from the Bragg reflections from the model. The R-expected, or R_e, (Equation 6) is based on the ideal weighted sum of squared residuals, so it defines a limit for the fit due to variance of individual measurement's underlying populations. In this case, O is the number of bins in the interval used and P is the number of parameters that are being fitted.

$$R_e = \left[\frac{(O-P)^2}{\sum w_i y(obs)^2} \right]^{1/2} \tag{6}$$

The goodness of fit function is another measure of the success of the fit. Young [18] calculates goodness of fit as R_{wp}/R_e. This is equivalent to Equation 7. In Equation 7, O is the number of bins in the interval used and P is the number of parameters that are being fitted. An ideal goodness of fit is between 1.0 and 1.5. The higher value may indicate false minima, a poor model, preferred orientation problems, or sample recrystallization. The lower value indicates a model with too many parameters than is reasonable based on data quality.

$$GoF = \left[\frac{\sum w_i((I_i(obs) - I_i(calc))^2}{O-P} \right]^2 \tag{7}$$

1.3 Error Estimation

Rietveld analysis requires the input of intensity error information from datasets to calculate an accurate weighting scheme, which is important in order calculate residuals and later the goodness of fit reported by most authors. Since intensity uncertainties increase the minimum parameter uncertainties, some uncertainty must be removed or at minimum determined and errors must be quantified. After finding a goodness of fit value, some authors multiply it by the estimated standard deviation (E.S.D) [2]. The E.S.D. of a particular parameter estimate is a measure of the irreducible minimum uncertainty in the values of the parameters for an accurate model [18]. The multiplication is executed in order to artificially correct an assumed overweighting of all data points, by adding an additional weighting scheme at

the end. However, the assumption is only an approximation and the multiplication is not statistically meaningful. Model inaccuracies can have a correlation with model parameters. Additionally estimated model parameter errors can also have a systemic bias based on the particular model used. In spite of these problems, they are a reasonable estimate of the true value [18].

The method by which errors are calculated has altered over the years for synchrotron powder diffraction. Hammersleys method [8] is used in FIT2D; the most commonly used program for data pre-processing. Hammersley [8] calculates the standard deviation of an averaged intensity from the variation in values that produced that intensity. Hammersley uses the weighted average variance, Equation 8.

$$\sigma_i^2 = \frac{\sum_{j=1}^{N} w_{ij}(I_j - O_i)^2}{\sum_{j=1}^{N} w_{ij}} \times \frac{N}{N-1} \tag{8}$$

where N is the sum of the weights. The *N /(N - 1)* term accounts for the fact that the mean is derived from data. The standard deviation is calculated from Equation 8; the difference between the weighted average from individual pixel values squared and the overall weighted average squared [8] (Equation 9).

$$\sigma_i^2 = \left(\frac{\sum_{j=1}^{N} w_{ij}(I_j^2)}{\sum_{j=1}^{N} w_{ij}} - O_i^2 \right) \times \frac{N}{N-1} \tag{9}$$

Hammersley points out that measurements are assumed to be independent, and that this is untrue. The width of the point spread function for two dimensional detectors creates a correlation in adjacent pixel values which would cause underestimation of the standard deviation. This problem is limited for narrow peaks.

To solve the problem of determining correct error statistics, it became important to find the statistical distribution of the data. Most approaches are general and phenomenological because factors contributing to the noise in a pattern are complex and specific to the sample and experimental conditions. For powder diffraction data, Rietveld analysis programs generally assume counting (Poisson) statistics. According to Chall [4], the data is Poisson-distributed, since repeated measurement of the same diffraction angle by different pixels yield a Possion distribution. For this distribution, Chall [4] indicates that the probability finding a particular intensity is:

$$p(I) = \frac{\bar{I}^I}{I!} e^{-\bar{I}} \tag{10}$$

where $\bar{I}$ is the mean intensity and the standard deviation is $\sigma = \sqrt{\bar{I}}$. The mean intensity may be calculated as the arithmetic mean:

$$\bar{I}_{arith} = \frac{\sum_{j=1}^{N} I_i}{N} \tag{11}$$

The standard deviation of the mean intensity of the Poisson distribution is:

$$\sigma_I = \Delta\bar{I} = \sqrt{\bar{I}/N}. \tag{12}$$

Vogel [17], whose analysis focuses on texture, suggests that the intensity error should be calculated based on the variations in distribution of a particular population instead of the variation of the mean. The equation for variance is Equation 13:

$$\sigma^2 = \frac{\sum_{i=1}^{N} (I_i - \bar{I})^2}{N} \tag{13}$$

Since there are more pixels in higher two-theta bins the standard deviation in these bins is lower. For intensities higher than around 20 counts, a Gaussian measurement could be used to approximate the data. The probability for finding a particular intensity for a Gaussian distribution is shown in Equation 14.

$$p(I) = \frac{1}{\bar{I}\sqrt{2\pi}} e^{-\frac{(I-\bar{I})^2}{2\bar{I}}} \tag{14}$$

The standard deviation of the Gaussian can be used as a measurement of uncertainty, if a single measurement is performed [4], as it would be for experiments using a one-dimensional detector. For a case with a single measurement, the average intensity is undetermined. In this case, the measured intensity is used to find error. The 68% confidence interval for the Gaussian distribution is one times the standard deviation.

$$x_{\pm} = \mu \pm 1 \times \sigma \tag{15}$$

So, with a confidence of 68% the mean value is within the calculated error of the measurement. In Chall [4], Gaussian and Poisson distributions are compared by fitting them to data. Gaussian error estimation can approximate

data effectively [4], but a random chosen intensity on a particular Debye-Scherrer ring would probably fail to reveal diffraction peaks in data. This is seen in Figure 5.

Chall does find that averaging the bin intensity values can reveal peaks [4]. However, the standard deviation was not found to be an appropriate measure of the error since it discounts the number of pixels used in the measurement. For multiple measurements, the number of pixels must be accounted for. For the goodness of fit, Equation 7, the ideal value is 1.0 to 1.5 [18] The correct weighting scheme for w_i for the residuals, in the R-weighted pattern equation, is $w_i = l/a^2$, where a is the 67% confidence interval.

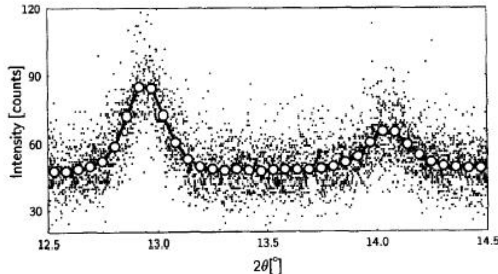

Figure 5: A figure from Chall [4] demonstrating the difference between the average pixel value for a particular bin and the spread of the data. The scatter was calculated to on the order of $I^{1/2}$. Chall uses this figure to demonstrate that the second peak is within the order of magnitude of the error and therefore could not be considered observed for some estimates of the error.

Only with weight,

$$a = \frac{\sigma}{\sqrt{N}} \tag{16}$$

are these values produced.

Chall experiments using different weighting schemes to determine their effect on the goodness of fit. Correct weights, $w_i = N_i^2/\sigma^2 = 1/a^2$, unit weights $w_i=1$, the default weights by most software $w_i=1/I_i$, and the equivalent from Chall [4] calculated from Equation 17:

$$\sigma^2 = \frac{\sum_{i=1}^{N}(\bar{I} - I_i)^2}{(N-1)} \tag{17}$$

as $w_i=1/\sigma_i^2$. This results in Figure 6, which uses the same data from the previous figure.

Wrong weights result in incorrect goodness of fit values. Correct values produce realistic goodness of fit values. For their dataset, the goodness of fit,

based on the calculated standard deviation of a Poisson distributed dataset, indicates the data does not have a Poisson distribution. Chall suggests the distribution may be corrected by removing diamond diffraction peaks, which cause a deviation from unimodal intensity distribution [4]. Chall also mentions that some authors multiply the estimated standard deviation by the goodness of fit. When the goodness of fit is greater than one, as it is for correct weights, this is not problematic and is usually conservative [4]. When overly large errors are found due to incorrect weighting statistics, the underestimated standard deviation reduces the goodness of fit to unrealistic values.

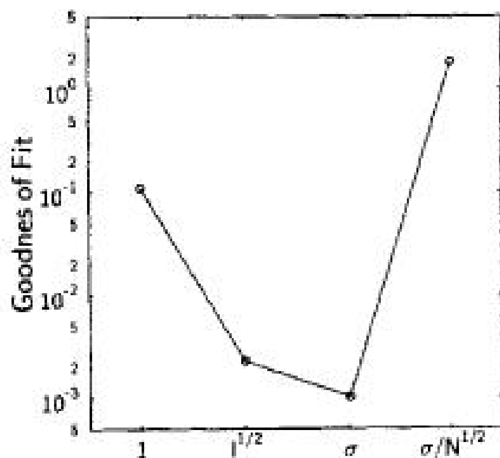

Figure 6: A figure from Chall [4] comparing the effect of different weighting schemes on the goodness of fit.

Some care should be taken in order to determine the effect of the dataset used on the statistics, since this may affect the distribution of the dataset. An ideal dataset is required. Powder diffraction experiments usually do not have the properties to realize an ideal pattern. This pattern would be generated by a large number of uniformly sized and randomly oriented crystals, which produce an ideal normal intensity distribution [11]. Since this is not ordinarily true, normally distributed intensities must be extracted from data. For our purposes, an ideal image for processing is necessary to account for, understand, and eventually correct for systemic biases. Hammersley [8] suggests using data from the National Institute of Standards and Technology (NIST). They produce reference 674a and other reference calibration standards for powder diffraction, detector calibration, and other calibration purposes. Hammersley [8] suggests that very small amounts of powders and very small X-ray beams may produce datasets with different characteristics. Hinrichsen [11] uses a high quality calibration image of LaB_6. This data is

shown in Figure 7 for a 0.02 degree bin. However, datasets with real samples may not be appropriate since an aziumuthal intensity deviation occurs due to sample absorption. Correcting the data for polarization produces a uniform azimuthal intensity distribution.

For high pressure data, the distribution seems to indicate a power law may be appropriate because there are a large number of low intensity pixels and very few high intensity pixels. The high intensities complicate analysis because they have an unpredictable effect on average 2-theta bin values after integration. Even if experimental effects are removed, the data usually does not produce an easily recognizable intensity distribution solely as a result of the samples low intensities. For ideal data, a normal distribution is unconvincing [11]. See Figure 7. The power law Hinrichsen advocates is the Pareto distribution, which originally was utilized to describe wealth in societies.

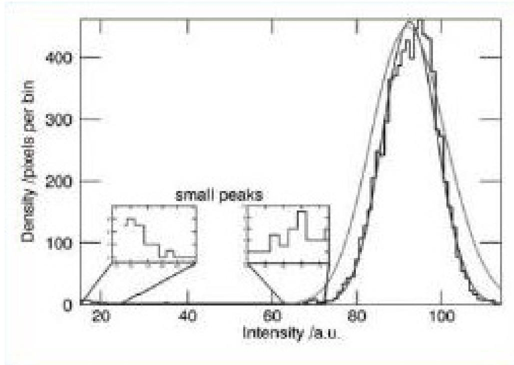

Figure 7: A figure from Hinrichsen [12] comparing a Gaussian distribution to a Poisson distribution for a 0.02 degree bin. Hinrichsen suggests that the normal distribution is a better fit to the data. Unfortunately the distributions are labeled in color and are indiscernible on a black and white text. The finding is in contrast to the selections of previous authors.

$$P_{Pareto}(x) = \begin{cases} 0 & for x < b \\ ab^a/x^{a+1} & for x \geq b \end{cases} \quad (18)$$

The probability for finding a particular value x in a Pareto distribution is shown in Equation 18. It describes well the main peak of the normal distribution as well as the largest and weakest intensities.

In order to correct some of the problems intrinsic to the sample in the datasets, some processing is done in the structural analysis programs. According to David [5], there is no substitution for good normally distributed data. According to Hinrichsen [12], the difference between the integrals of the ideal Normal dataset and the Normal-Pareto distribution can be used

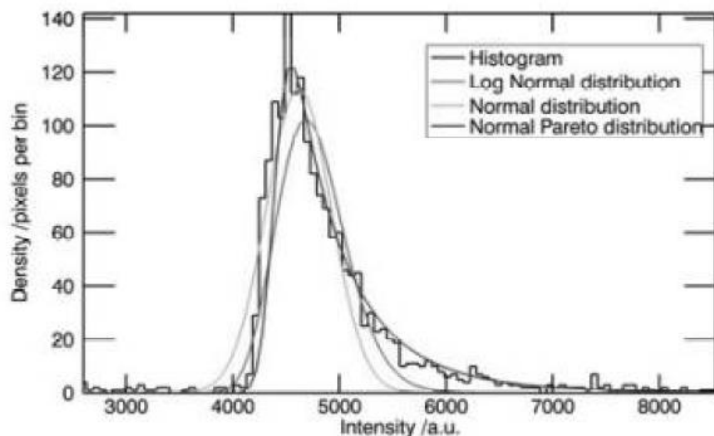

Figure 8: A plot from Hinrichsen [12] of the number of pixels versus an intensity value in which the fit of Normal-Pareto, Log Normal and Normal distributions are compared for a sample of $SnSO_4$ at 16 GPa. The author concludes the normal Pareto has the best fit [?]

to estimate the normal fraction of the distribution. Since the low intensity slopes are equal, the difference can be used to calculate the portion of high intensity data that must be filtered, using the approximation that the high intensity slope of normally distributes data is infinite [11]. This allows the extraction of normally distributed data.

1.4 FIT2D

FIT2D [8] is a commonly used program for data pre-processing of x-ray powder diffraction data. It has several limitations. The program does not have methodology to provide error estimates, which are required by Rietveld refinement programs. In the absence of error estimates, Rietveld programs assume counting statistics are sufficient. FIT2D does contain methodology for manual masking of high intensity spikes due to diffraction from the diamonds. It allows removal of small regions and polygons to remove larger areas. However, the processes is manual, tedious, and somewhat subjective. There is no methodology to remove individual dead pixels. With the goals of producing more accurate uncertainties to better intensity analysis and to automatic mask bad data, another program is necessary.

2 Methods

2.1 Experimental Methodology

2.1.1 Components

High pressure diamond anvil cell experiments simulate the hydrostaticity, temperature uniformity, thermodynamic and chemical conditions from the Earth's interior [3]. They can be used to study structural, chemical, and physical properties of materials and phase transitions at these conditions. For our purposes, the diamond anvil cell, sample, detector, and synchtrotron X-ray beam are the most important elements of the setup. Details of the equipment and setup are useful, since the experiment produces artifacts in the diffraction profile that should be removed. Synchtrotron facilities allow measurements under the high pressure and temperature experimental conditions.

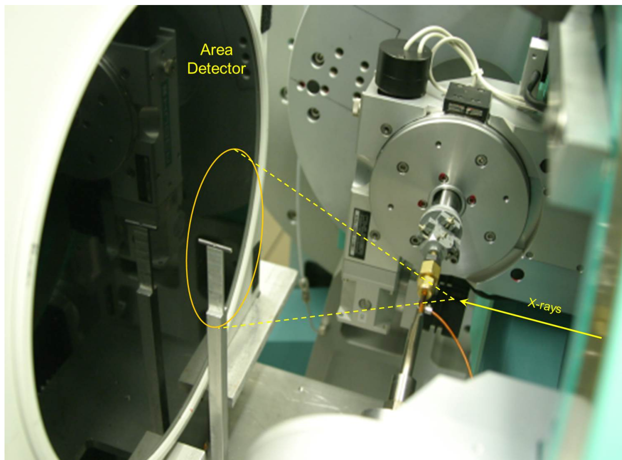

Figure 9: A photograph of the setup with an area detector shown. Indicated in yellow is the path of the x-ray beam and the Debye-Scherrer cone that would be resultant if a sample was in position.

The diamond anvil cell (DAC) consists of two diamond anvils and a gasket. The gasket has a pre-drilled hole in the center, which is used as a pressure chamber. The chamber contains a pressure transmitting medium, which transfers quasihydrostatic high pressures to the sample uniformly. By doing so, the medium eliminates shear pressures on the sample due to the diamonds and creates quasihydrostatic conditions in the gasket. The gasket supports the diamonds and reduces diamond breakage due

to very high shear forces. At very high pressures, gasket thicknesses are limited to less than 50 micrometers. Gaskets can be made of many materials. Steel is conventionally used for room temperature experiments, however it is inadequate for high temperature conditions due to its tendency to creep at those conditions. Rhenium or tungsten can also be used and are advantageous due to their high melting temperatures. The gasket thickness depends on a variety of factors, such as the diamond anvil cell size, pressure medium and desired pressure range. The diamond anvils are used to create the high pressure conditions by amplifying pressure exerted on the base to a much smaller area at the tip of the anvil. Diamonds are used because of their hardness, range of transparency, thermal conductivity and limited thermal expansivity. Their transparency allows a range of wavelengths to be used for investigation. Their hardness increased the pressure extremes attainable [1].

2.1.2 Experimental Details

For synchrotron experiments, there are two techniques; angle-dispersive and energy-dispersive. A continuous energy spectrum is used in energy-dispersive technique, which offers fast data acquisition. Angle-dispersive technique involves a single wavelength. It provides more accurate intensities for structural refinement and has higher spatial resolution than energy-dispersive. We utilize angle-dispersive technique. During the experiment, an x-ray beam from a source is diffracted through a sample contained in a DAC. The diffracted photons, in the shape of Debye-Scherrer cones, are then captured by a flat two-dimensional detector as rings. See Figures 9 and 13. The incident beam is prevented from reaching the detector by a beam stop. The data is recorded as greyscale, 16-bit, *.tiff images. First, a dark image is taken without the sample or beam present in order to record and later remove any electronic noise that may be present. Subsequently, actual measurements are taken with the beam and sample present. Examples of real data are shown in Figures 10 and 11.

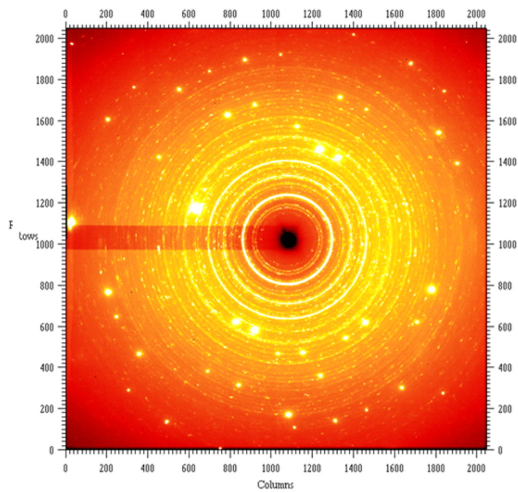

Figure 10: A raw powder diffraction data image collected on an area detector. The image has be scaled, in order to display high intensity peaks and spikes

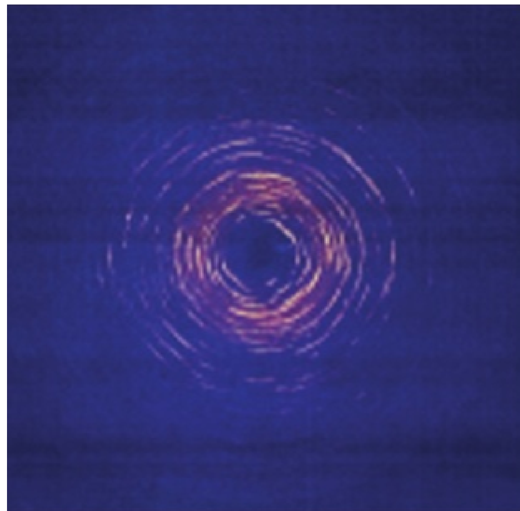

Figure 11: An example of a texturized data image. Texture is one of the problems that may be present in a dataset.

2.1.3 Sources of Error

The distribution of diffraction peaks is altered by artifacts in the data created by the experiment. The sample itself, the diamonds, and the gasket can cause problems. The small sample and beam size yield poor powder statistics. The sample can produce other signal besides data relevant for our purposes. There can be a preferred orientation problem, which can be somewhat mediated by finely grinding the sample to a particle size of less than a few microns. The sample contains crystallites that may not be in ideal diffraction positions. Weak peaks that do not conform to common crystallographic distributions would be the result. The problem can be somewhat alleviated by rotating the sample. However, the rotation angle possible in diamond anvil cells is small. Large crystallites within the sample can create high intensity single crystal spots. The gasket generates secondary excitations and powder lines that are mostly highly textured [16]. A smaller beam size would reduce gasket diffraction, but also reduces the volume diffracting [16]. From the diamonds, single crystal peaks can create problematic multimode distributions [11].

There are also artifacts related to the setup. One extremely noticeable feature is due to the shadow from the beam stop. It creates a band of lower intensities across the diffraction image. The detector can have calibration errors, spatial distortions and a nonlinear response to detected intensity [8]. Some problems with the setup can be corrected through data processing. The experimental geometry must be calibrated for the position of the beam center on the detector and the non-orthogonality of the detector with the sample. There is a polarization effect due to the polarized nature of the synchrotron beam and the effect of the sample. The beam also produces a substantial background due to air scattering, which is highest near the beam center and decreases with distance. The sample also produces Compton scattering, as photons interact with electrons in the sample.

2.2 Data Reduction Software

The purpose of the project is to create software to calculate meaningful error estimates and to provide automatic masking capabilities. In order to accomplish this, specific operations must be performed on the data, first to correct for systemic problems and later for actual data processing purposes. The order of processing is shown in Figure 12.

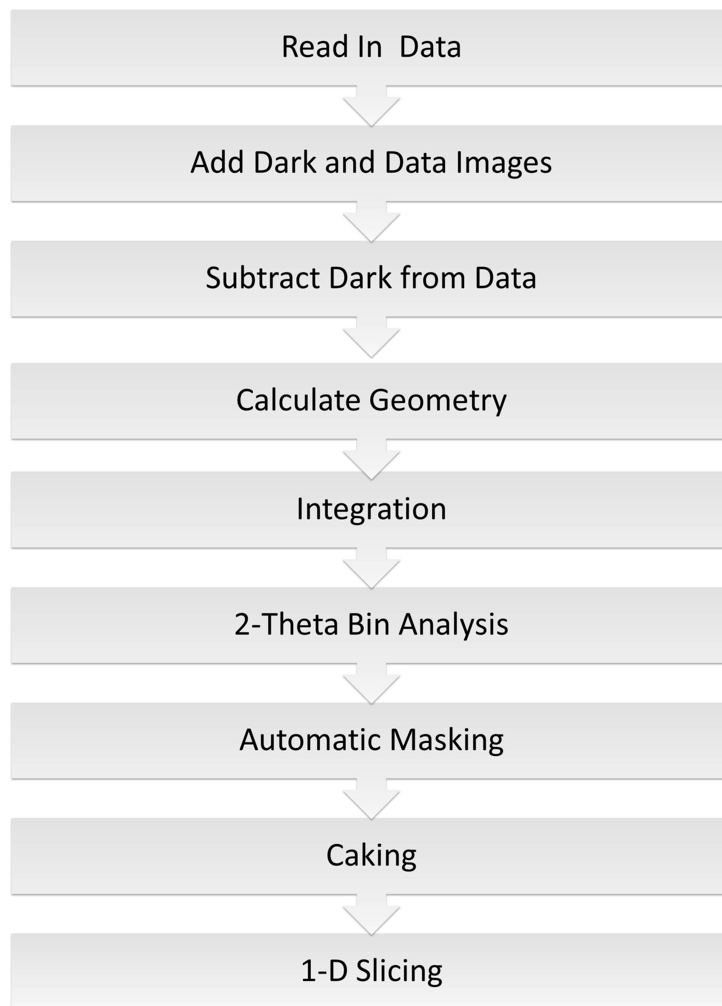

Figure 12: The order of operations involved in image processing is shown in the flow chart. Some procedures may be optional and not included in the current software version.

2.2.1 Read in Data and Background Correction

The first step is to read in the data. Our data consists 16 bit greyscale *.tiff images. Python's Image Library and NumPy are used for this process. The individual pixels of each image are added to a data array containing the intensity values for the whole image. Each pixel intensity is saved at the

same two-dimensional (x, y) location in the array as the pixel in the image. Python array locations begin numbering at zero. In order to combine a set of images, pixel values at the same location in each image are added to the current summation of the pixel counts for that location within in an array. This process is completed for dark images and data images, separately. The dark image array is then subtracted from the data array. There is a loss in resolution due to pixellation, compared to other types of detectors.

2.2.2 Geometry Calculation and Correction

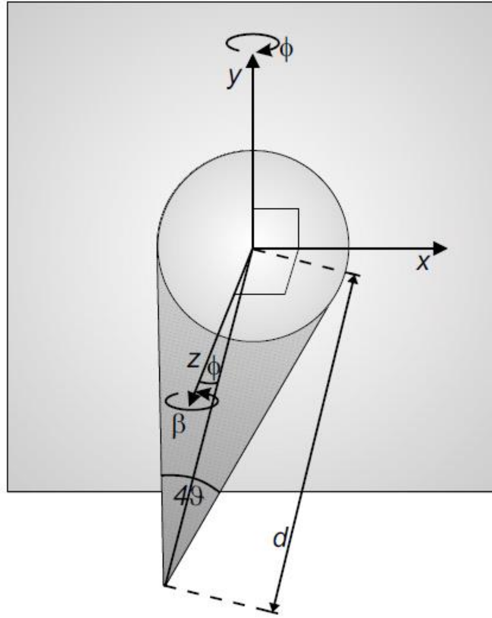

Figure 13: A diagram showing the orientation for the geometry calculation. The variables and angles used in the geometry calculation and correction from Equation 18 are indicated. The figure is from Vogel [16].

Next, the geometry must be calculated and any deviation corrected for using equations from Hammers-ley [8] and Vogel [16]. A map of corrected two-theta values for each pixel location is created and matched to the map of data intensities. This correction consists of a deconvolution involving a series of rotations that is applied to data. A right handed coordinate system is used.

The center point is considered to be the location where the incident beam intercepts the detector. For an ideal case, the Debye-Scherrer cone is perpendicular to the detector, and the intersection of the two is a ring. The Debye-Scherrer cone width is 2θ for a distance (d) from the sample to the detector. Details of the geometry are shown in Figure 13.

$$x_i^2 + y_i^2 = ((d - z_c)tan2\theta)^2 \tag{19}$$

However in powder diffraction experiments, there is generally deviation from the ideal geometry, so a rotation is applied in order to correct for the distortion. If the cone is tilted by an angle ϕ, a rotation of - ϕ is applied to achieve a flat orientation. Including a correction for the rotation, the scattering angle for an individual pixel is calculated as:

$$2\theta = \arctan\left[\frac{((x-x_0)cos\beta + (y-y_0)sin\beta)^2 cos^2\varphi}{(d+((x-x_0)cos\beta)(y-y_0)sin\beta)sin\varphi)^2} + \frac{(-(x-x_0)sin\beta + (y-y_0)cos\beta)^2}{(d+((x-x_0)cos\beta)(y-y_0)sin\beta)sin\varphi)^2}\right]^{1/2} \tag{20}$$

See Hammersley [8] for more details. There is an alternative method to calculate a geometry correction. This method is detailed in Rajiv [15]. Distorted Debye-Scherrer rings are automatically found and when bisected appear as a peak in intensity. Their eccentricity is calculated automatically using a Hough transformation. The geometry correction is based on the determined eccentricity. A quick method to utilize this technique was not found by this project.

2.2.3 Polarization Correction

The next step is the polarization correction. The synchrotron beam is polarized, and the sample can act as a polarizer due to dichroism. The effect on intensity varies based on the difference in polarization angle of the source and sample. The effect on intensities can be corrected based on distance from the beam center. The equation to calculate polarization is Equation 21.

$$P_i = \frac{(1 - Acos^2(2\theta))}{1 + A} \tag{21}$$

where A is $A=(1\text{-}f)/(1+f)$. f is the polarization rate perpendicular to the scattering plane. In this orientation, unpolarized intensities would have a f of zero, and synchrotron plane polarized light would have an f of one. The polarization information is generally provided in the dataset. See Figure 14 for the effect of the polarization correction on individual intensities.

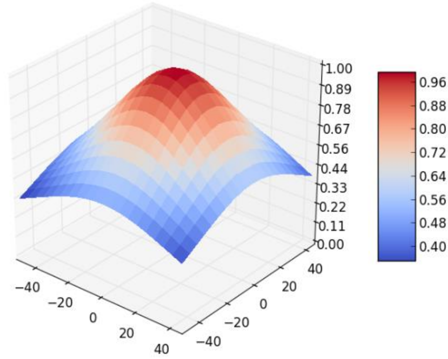

Figure 14: A diagram showing the effect of the polarization correction on the data.

2.2.4 Integration

Next, the data is integrated in order to determine a mean and to allow masking. Integration consists of collecting two theta values between two bounds into a user-specified number of bins. The number of bins is of minor importance, since all data is overbinned [4]. For increasing numbers of bins, the goodness of fit decreases and approaches 1.2; this is considered to be a good value [18]. At binning scales less than the width of an individual reflection, so that the reflections are resolved, the goodness of fit values do not change [4]. Increasing the number of bins is countered by better precision in determining intensity. Below peak resolution, the model becomes inapplicable and goodness of fit increases [4].

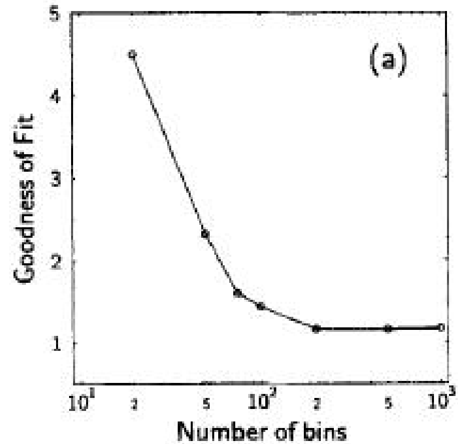

Figure 15: A diagram showing the change in goodness of fit values with increasing bin size. The goodness of fit is the formulation from Equation 7. From Chall [4].

2.2.5 Bin Analysis and Automatic Masking

Several forms of analysis can be performed on individual bins. To produce a diffraction plot, an average intensity must be found for each bin. The method for calculating average is the same for all distributions. Several distributions are used to analyze the data. See Equation 12 and the square root of Equation 17 to calculate the standard deviation of the two distributions. The standard deviation (σ) can be utilized as uncertainty, which is employed to calculate error estimates [4]. See the introduction for a more detailed discussion of intensity distribution. Peaks are measured with increasing accuracy for increasing numbers of contributing pixels. These bins are located at higher two-theta values. The two-theta bin number, average intensity, and standard deviation are saved to a new array. The next step is automatic masking.

Masking consists of removing undesirable populations of pixels. This includes intensities due to diamond peaks, dead pixels with a value of zero and pixels with the maximum intensity due to errors. These pixels produce an abnormal distribution, so the data does not follow any common crystallographic distribution. Removing them can be accomplished several ways. Depending on the assumed distribution, pixels within a user-specified number

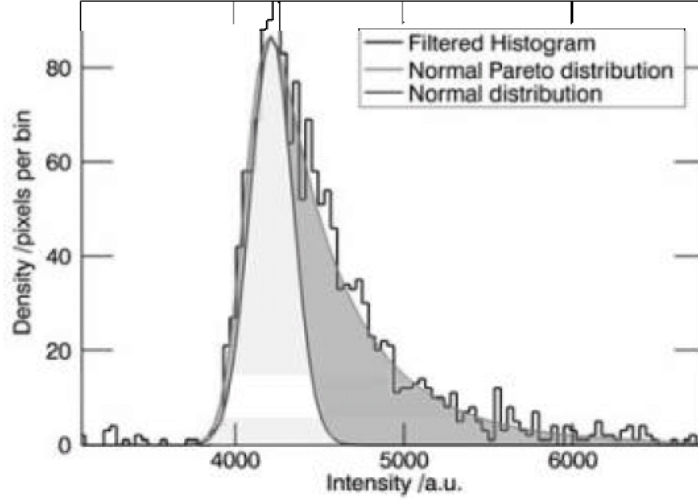

Figure 16: A plot of the fit of the Normal-Pareto distribution compared to the Normal distribution for an idealized data set. The high intensity tail produces the peaks that require filtering. The low intensity tail of both distributions is nearly equivalent. From Hinrichsen [12].

(n) of sigmas around the mean may be kept. This is calculated as Equation 22. The value is added to and subtracted from the mean to produce a cutoff, and pixels with intensities greater and less than the cutoff are eliminated.

$$I_{cutoff} = \bar{I} \pm n \times \sigma \tag{22}$$

For a Gaussian distribution, different numbers of sigma retained correspond to difference confidence intervals. n values of 1 and 1.96 correspond to a 68% and 95% level, respectively, of confidence that the true mean lies within the intervals for an unknown distribution.

There are alternative methods. Hinrichsen suggests using a fractile filter. Fractile filters can be used for any distribution. Pixels with intensity values from zero up to a defined fraction of the data or pixels with the maximum value down to a specified fraction can be eliminated from a single bin. The equation to determine the cutoff is:

$$I_{obs} \geq \frac{I_{max} - I_{min}}{100} \times (100 - Frac_{hi}) \tag{23}$$

$$I_{obs} \leq \frac{I_{max} - I_{min}}{100} \times (Frac_{low}) \quad (24)$$

The method is effective since it removes outliers and retains a normal data distribution [12]. While the filter can be used for other distributions, such as Gaussian, the fraction to be filtered becomes somewhat arbitrary for those functions. The Normal-Pareto function is suggested to fit the observed data best [11]. Ideally, our data should have a normal (Gaussian) distribution. The difference between the Normal-Pareto distribution and the Normal distribution is calculated as:

$$\int P_N(x)dx = \int P_{NP}(x)dx = 1 - Frac_{hi} \quad (25)$$

The difference between the two fractions is the fraction that requires filtering. A comparison of the fit is shown in Figure 16. The upper tail of the data contains high intensity pixels to be removed. This is true for the case that the intensity slope is infinite and the low intensity slopes are equal [12].

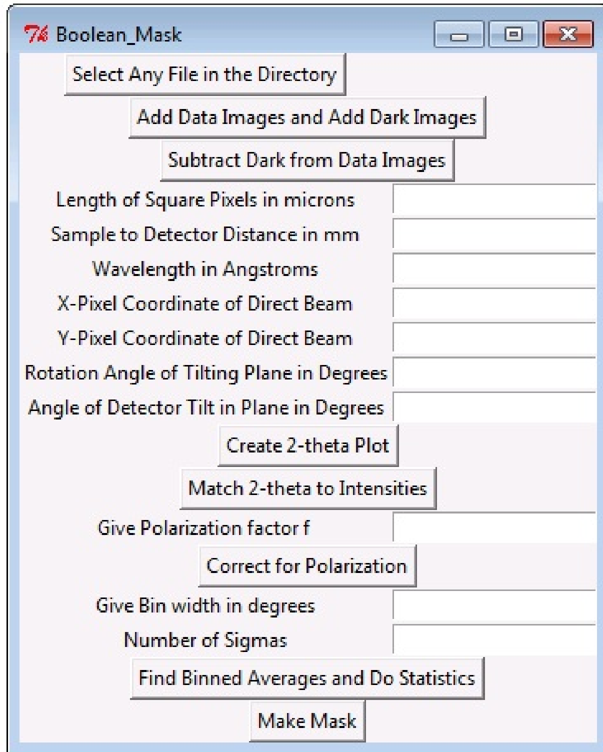

Figure 17: An image of the BeyondFIT2D user interface

2.2.6 Output

The output may be in the form of an image or as a powder diffraction profile, a two-theta versus intensity plot. The data should be suitable as input into any Rietveld refinement program.

3 Results and Discussion

3.1 Data Import

Data is imported as uint-16 greyscale format, which produces intensity values in a range from 0-65535. The pixels of the final image are normalized at some point during the summation process of individual images. This is observable due to a difference between the distribution of the population of normalized summed data and data that has not been normalized. To conform to the same methods as FIT2D, so that comparisons may be made, we also follow this process. In Figures 18, 19, 20 histograms of the images containing data (with the X-ray beam on), background (without the X-ray beam), and the final image are shown. These histograms are for data that has not been normalized.

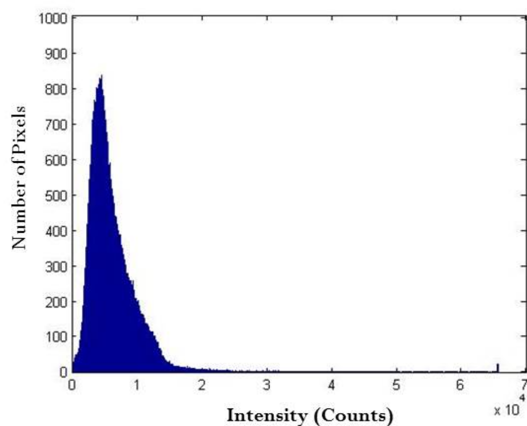

Figure 18: A histogram of the final image, consisting of data with background subtracted.

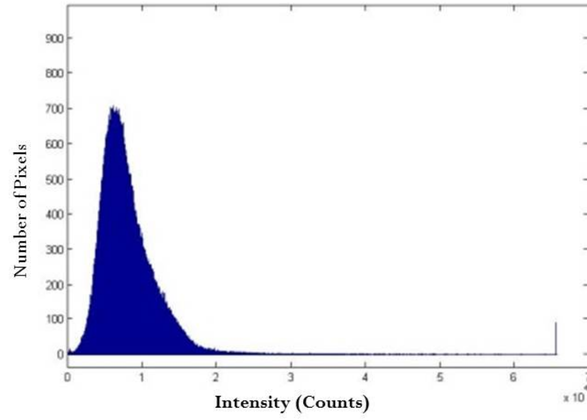

Figure 19: A summation histogram of the final data image, consisting of data with no background.

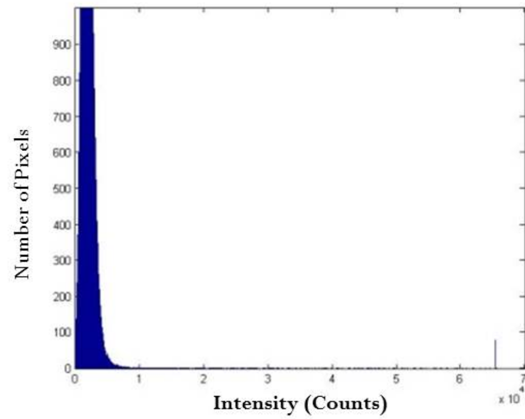

Figure 20: A summation histogram of the final background image, consisting of background images only.

3.2 Data Integration

In order to test the validity of the geometric correction, our integrated pattern should be compared to the pattern produced by FIT2D. FIT2D's previous decades of successful usage make it a reasonable standard. This process proved to be problematic and so that figure is not shown here. Simply calculating the 2θ location of any pixel location appears to be insufficient to reproduce FIT2D data. The calculation of the maximum 2θ value agreed for results from Beyond and from Mablab [14]. The value was within a reasonable amount of values suggested by FIT2D during processing. However, the integrated pattern does not match. While a programming error is possible, the possibility of FIT2D employing an additional correction is implied by other authors [16]. It has been suggested to be a partial transformation based on the eccentricities of individual rings cross-correlated with the calculated geometry [16]. Some variance in the pattern is caused by to differences in the method of calculating bin intervals. Once the data is integrated, the mean was calculated for individual bins. The value of the mean compared to all values for the bin is shown in Figure 21. The spread of the dataset can be calculated as the variance. For all subsequent relevant data, the bin for the third red data point in Figure 21 is used for calculations. It is located at the bin with a midpoint 22.83 degrees. For Figure 21, these three bins were selected because their average was within 0.02 counts of the average for the same bin in FIT2D integrated data.

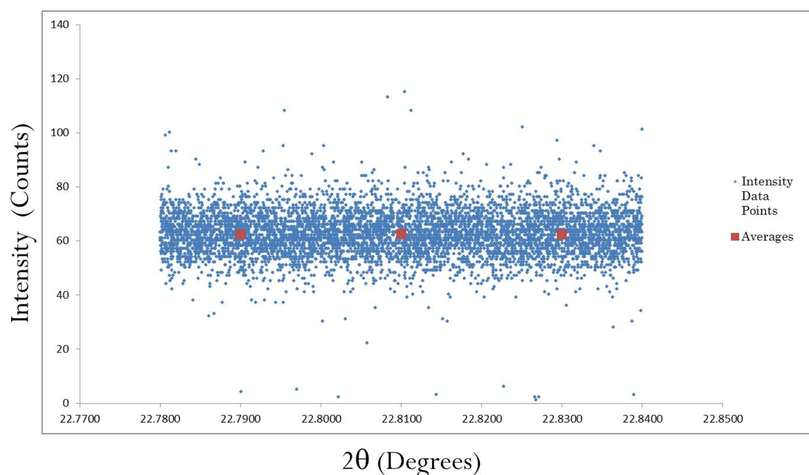

Figure 21: The average values of bins located at 2θ values of 22.79, 22.81, and 22.83. The variance of the dataset compared the final 2θ values is evident.

Rietveld analysis of powder diffraction patterns can be improved by the use of correct estimates of uncertainties in the integrated pattern. Otherwise, unrealistic goodness of fit values are produced. The correct estimate of uncertainty is dependent on the probability distribution used to model the data. In this section, Poisson and Gaussian distributions are fit to the dataset, and chi-squared values for each are determined. This was accomplished by calculating variables for the probability distribution function for each distribution for the 22.83 degree bin. The fitted function was used to calculate expected values for the statistical distribution. The chi-squared value for the expected and actual data were found in order to determine the plausibility of each distribution. A chi-squared value of 3.988 was found for the Gaussian distribution at a significance level of 0.05. Unreasonable Poisson values were found, with a chi square of 36.10. However, this may be due to problems with the Poisson fit. Since the Poisson distribution is discrete, care must be taken in calculating average values for each pixel, in the steps prior to integration.

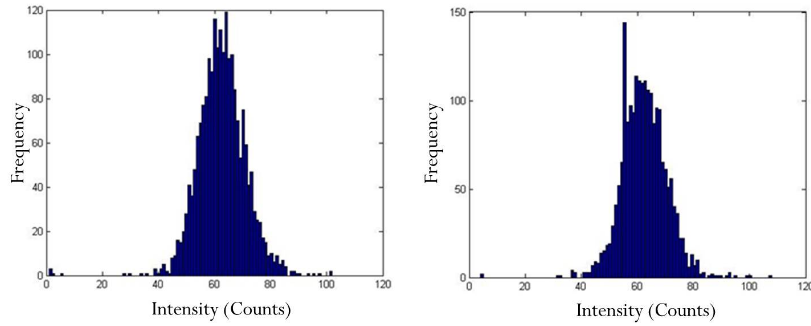

Figure 22: The distribution of the 2θ bins 22.83 (left) and 22.79 (right). The left bin is used for all subsequent data analysis, due to the small difference between its average and the FIT2D integrated bin value.

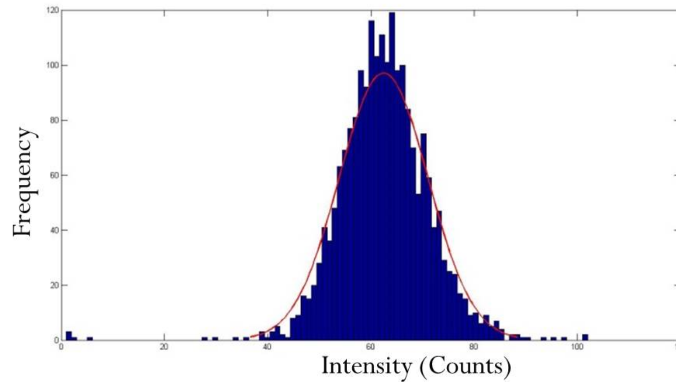

Figure 23: The histogram of the 22.83 bin is fit to the Gaussian distribution. The calculated fit is used to produce an expected pattern (red line). The chi-squared goodness of fit was found to be 3.988.

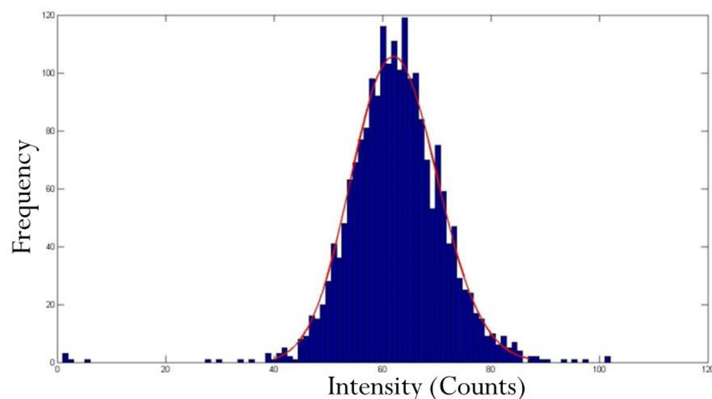

Figure 24: The histogram of the 22.83 bin is fit to the Poisson distribution. The calculated fit is used to produce an expected pattern (red line). The chi-squared goodness of fit was found to be 36.10.

These observations and the high overall averages suggest the Gaussian model could be used.

3.3 Automatic Masking

After an average intensity has been determined, automatic masking is executed. Two options have been introduced. Masking can be accomplished by designing a fractile filter to calculate the fraction of the total data requiring masking from the upper and lower ends of the distribution. The alternative is to retain only pixels with intensities within a certain level of uncertainty (standard deviations) around the mean. In Figure 25, the effect of integrating data and calculating averages while retaining only intensity values that are within 150 standard deviations around the mean for a Poisson distribution is compared to retaining only data that is within a 68% confidence level or one standard deviation for the Gaussian that the mean lies within that region. An integrated dataset without masking is also presented for reference. Removing a number of high intensity pixels results in a lower measurement for the average value of a bin. Outliers from diamond diffraction or single crystal spots from the sample would be likely culprits for high intensity pixels. For this dataset, the elimination may be too severe and remove pixels that are

important for analysis, due to the low signal to noise ratio of the data. The retained interval should be selected with care. Due to the Poisson distribution retaining more peaks than the Gaussian, Poisson masking was used in Figure 26.

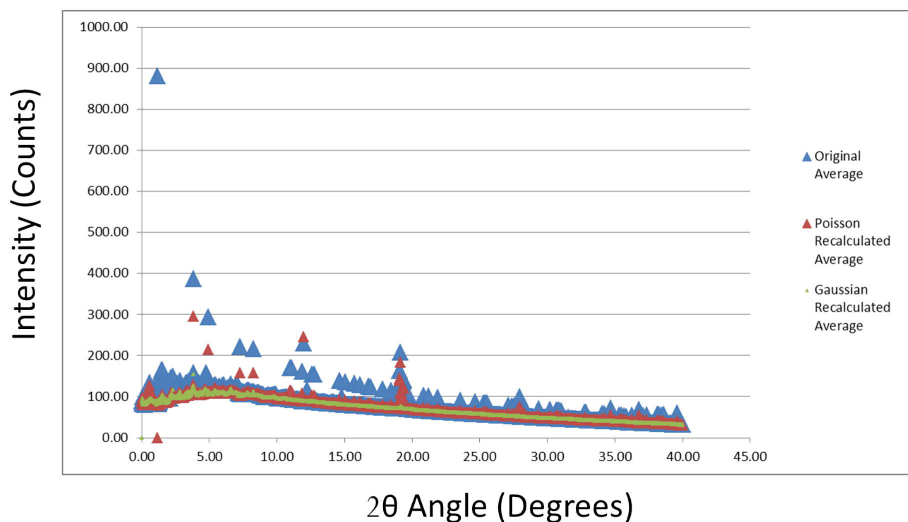

Figure 25: A plot of the integrated pattern for values with a 150 standard deviation Poisson mask, a 1 standard deviation Gaussian mask, and the unmasked pattern. The blue triangles are the original unmasked pattern. The red triangles the recalculated average after the Poisson mask has been applied. The green triangles are the recalculated averages after the Gaussian mask have been applied. The mask applied appears to possibly be too extensive. A large portion of peaks that may be real data appear to be masked. This may be remedied through use of higher sigma values.

Each type of mask removes different amounts of pixels from each bin. The total number of pixels eliminated by the differences in calculation of the masks varies. For a Poisson distribution, the mask was calculated for one, two, and four sigmas. The images of the masks are shown in Figure 26. Retained pixels are a single white value, while masked pixels are black. The sigma value varies for each bin. When utilizing a Poisson distribution, the standard deviation remains small. The quantity of data this eliminates should be noted.

Fraction of Pixels Eliminated Using Varying Sigma Values				
	10	**50**	**100**	**1000**
Poisson Distribution	0.8504	0.6148	0.3836	0.0006
	1	**1.64**	**1.96**	
Gaussian Distribution	0.2092	0.0594	0.0311	

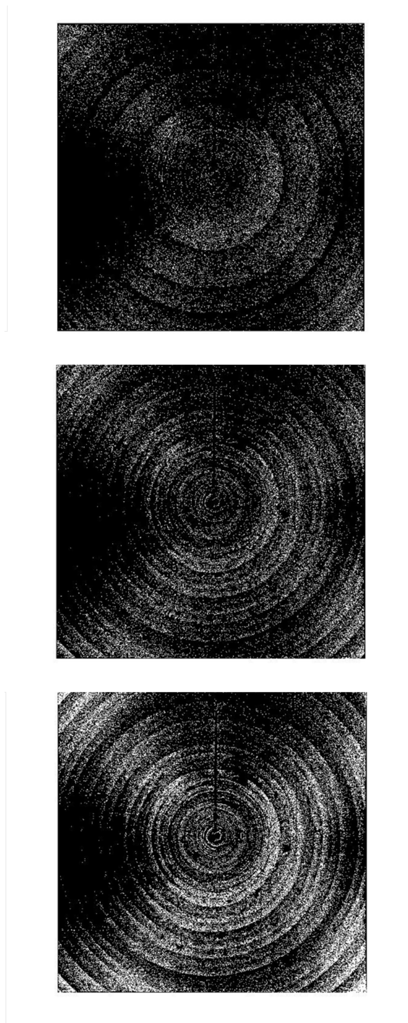

Figure 26: The images consist of the pixels retained after a 1, 2, and 4 standard deviation Poisson mask has been applied. Dark pixels are masked data. White pixels are pixels that contain information. There are no greyscale intermediate values in the image; all pixels are either black or white.

4 Conclusions

The geometry correction creates problematic data. Systematically altering the contents of bins results in distributions that are altered. The calculated chi-square should indicate that at low counting intervals the Poisson distribution is appropriate and the Gaussian is appropriate at higher values [4]. The net effect of the incorrect geometric transformation is to create mostly lower intensities than the expected dataset. This shifts the shape of the distribution and can alter chi-square values and possibly the choice of distribution, the standard deviation, mean, retained pixels in masks, and the effectiveness of filtering mechanisms. The net effect is to generally lower average intensities, and it cannot be minimized at this time. All bins contain some pixels with a different bin assignation due to the geometric correction. Though pixels with lower intensity two-theta values may have less correction, they also contain fewer pixels and therefore are more strongly affected. The converse of this situation is true at high two-theta values. The bin values used for analysis were chosen due to an average intensity that is within 0.02 of the FIT2D data. However, the bin choice may not be ideal since the bin probably does not contain a peak.

Our results suggest the distribution to be Gaussian. According to previous authors, the Poisson distribution is more appropriate for lower mean values and the Gaussian distribution is more appropriate for higher mean values [11]. This distribution produces reasonable goodness of fit values in the experiments by Chall [4]. The usage of automatic masking can eliminate the need for manual masking and produce a standardized methodology. The mask was successful at removing outliers from the pattern. The removed pixels had a significant effect on the mean. Calculating Gaussian uncertainties should provide a better estimate of uncertainty since the uncertainty has a somewhat logical basis. The Gaussian distribution also suggests natural values for filtering pixels based on the variance of the population.

5 Outlook

5.1 Geometry Correction

The geometric correction of FIT2D is currently unknown to this project. Scripts, known as macros, created for FIT2D have been used to extract the geometric correction for this program. Open source software such as PYFAI may also offer an alternative. Post geometry correction, ideally we would examine the effect of the new uncertainty calculation on structural determination goodness of fit values and the effect of masking on the uncertainties.

5.2 Normal-Pareto

The Normal-Pareto distribution suggested by Hinrichsen, may provide a better alternative than the distributions examined in the project. This distribution may work especially well since tails for the distribution do not have to be symmetric. It is appropriate for skewed datasets.

5.3 Fractile Filters

Outliers may also be eliminated through the use of fractile filters, also suggested by Hinrichsen. They suggest an upper and lower fraction of the data to be eliminated. This may be more effective, provided the high fraction of the data does not contain information resultant from the sample. Unlike other methods, values are not retained based on proximity to the mean. The method also suggests that separating the uncertainty measurement procedure and elimination of poor data points may be worthwhile.

References

[1] W.A. Bassett, *Annual Reviews in Earth and Planetary Science*, 7, 357-384 (1979)

[2] J.F. Berar and P. Lelann, *Journal of Applied Crystallography*, 24, 1-5 (1991)

[3] R. Boehler, *Reviews of Geophysics*, 38, 2, 221-245 (2000)

[4] M. Chall, K. Knorr, L. Ehm, W. Depmeier, *High Pressure Research*, 17 315-323 (2000)

[5] W.I. David, *Powder Diffraction: Theory and Practice, Acta Crystallographia A*, 64 (2002)

[6] R.E. Dinnebier, and S.J. Billinge, *Structure determination from powder diffraction data*, (2008)

[7] Y. Fei and Y. Wang, *Reviews in Mineralogy and Geochemistry*, 41, 521-558 (2000)

[8] A.P. Hammersley, et al., *High Pressure Research*, 14, 235 (1996)

[9] N.F.M. Henry, H. Lipson and W.A. Wooster, *The Interpretation of X-ray Diffraction Photographs*, 1951.

[10] B. Hinrichsen, R.E. Dinnebier, P. Rajiv, A. Grzechnik, and M. Jansen, *Journal of Physics: Condensed Matter*, 18, S1021-S1037 (2006)

[11] B. Hinrichsen, R.E. Dinnebier, M. Jansen, *Z. Kristallographie. Suppl.*, 30, 139-146 (2009)

[12] B. Hinrichsen, R.E. Dinnebier, M. Jansen, *Z. Kristallogr. Suppl.*, 30 147-153 (2009)

[13] A. Le Bail, Whole powder pattern decomposition methods and applications: A retrospection, *Powder Diffraction,* 20, 4 (2005)

[14] MATLAB and Statistics Toolbox Release 2012b, The MathWorks Inc., Natick, Massachusetts, United States.

[15] P. Rajiv, B. Hinrichsen, R. Dinnebier, M. Jansen, and M. Joswig, *Powder Diffraction* 22, 1 (2007)

[16] S. Vogel, *High-Pressure and Texture Measurements with an Imaging Plate* , Dissertation, (2002)

[17] S Vogel, L. Ehm, K. Knorr, G. Braun, *Advanced X-ray Analysis*, 45, (2002)

[18] Young, R. A, *The Rietveld Method*, Oxford University Press, Oxford, (1995)

6 BeyondFIT2D Code

```
import Tkinter, tkFileDialog
import os, fnmatch
from tkSimpleDialog import *
from PIL import Image
import numpy
import math
numpy.set_printoptions(threshold='nan')

global directoryname, diffimg0, file
global arrayfinallight, arrayfinaldark
global originalpic, thetamap, arrayfinal
global sortedpic_index,  keepit_pix, keepit_inten

def directoryfinder():
    global diffimg0, originalpic
    diffimg0 = tkFileDialog.askopenfilename()
    originalpic=diffimg0
    global directoryname
    directoryname=os.path.dirname(unicode(diffimg0))

def image2array(im):
    newArr = numpy.array(im.getdata(),numpy.uint16)
    newArr2 = numpy.reshape(newArr,im.size)
    return newArr2

def addimage():  #subcommand
    global directoryname, diffimg0, arrayfinallight, arrayfinaldark
    blank = numpy.zeros(4194304, dtype=numpy.int32)
    blank.shape = (2048, 2048)
    arrayfinallight= numpy.array(blank,numpy.int32)
    arrayfinaldark=numpy.array(blank,numpy.int32)
```

```
    counter=1
    for file in os.listdir(directoryname):
        if fnmatch.fnmatch(file, '*.tif'):
            if counter%2==0:
                im2=Image.open(unicode(os.path.join(directoryname, file)))
                array0 = image2array(im2)
                arrayz=array0.astype(numpy.int32)
                arrayfinallight = arrayz + arrayfinallight
                counter = counter + 1
            else:
                im2=Image.open(unicode(os.path.join(directoryname, file)))
                array0 = image2array(im2)
                arrayz=array0.astype(numpy.int32)
                arrayfinaldark = arrayz + arrayfinaldark
                counter = counter + 1
            print counter
    numpy.savetxt('light.txt', arrayfinallight)
    print 'data done!'
    numpy.savetxt('dark.txt', arrayfinaldark)
    print 'darks done!'

def finalimg():
    global arrayfinallight, keepit_pix, keepit_inten
    global arrayfinaldark, diffimg0, originalpic, arrayfinal
    arrayfinal = arrayfinallight - arrayfinaldark
    numpy.savetxt('final.txt', arrayfinal)
    print 'darks subtracted'
#    im3 = Image.new('L', (2048, 2048))
#    im3.putdata(arrayfinal)
#    im3.save("newimage.tiff")

def twotheta():
    global thetamap
    thetamap = numpy.zeros(4194304, dtype=float)
    thetamap.shape = (2048, 2048)
    S2DD2 = float(Entry.get(S2DD))*math.pow(10, (-3))
    S2DD3 = float(Entry.get(hpixel))*math.pow(10, (-6))
    #S2DDx = S2DD2/S2DD3
```

```
    XPCoorx=float(Entry.get(XPCoor))*S2DD3
    YPCoorx=float(Entry.get(YPCoor))*S2DD3
    phi=float(Entry.get(DetTilt))*math.pi/180
    beta=float(Entry.get(RotAng))*math.pi/180
    for y in range(2048):
        for x in range(2048):
            x1=x*S2DD3
            y1=y*S2DD3
            a=(x1-XPCoorx)*math.cos(beta)
            b =(y1-YPCoorx)*math.sin(beta)
            c= math.pow((a+b), 2)
            d=math.pow(math.cos(phi), 2)
            first_term=c*d
            e=-1*(x1-XPCoorx)*math.sin(beta)
            f=(y1-YPCoorx)*math.cos(beta)
            second_term=math.pow((e+f), 2)
            h=(x1-XPCoorx)*math.cos(beta)
            i=(y1-YPCoorx)*math.sin(beta)
            j=(h+i)*math.sin(phi)+S2DD2
            bottom_term=math.pow(j,2)
            k=(first_term + second_term)/ bottom_term
            l=math.sqrt(k)
            thetamap[x][y] = math.atan(l)
#    numpy.savetxt('map_ctr.txt', thetamap)

def combine():      #kind of redundant
    global thetamap, arrayfinal, sortedtheta
    global sortedinten, sortedpic_index
    sorted_index = thetamap.argsort(axis=None)
    pic_index=numpy.arange(4194304)
    sortedpic_index=pic_index[sorted_index]
    sortedtheta=thetamap.flatten()[sorted_index]
    sortedinten=arrayfinal.flatten()[sorted_index]
    sortedfinal=numpy.dstack((sortedtheta,sortedinten))
    #numpy.savetxt("file_array.txt", sortedfinal)
    #print sortedfinal

def depolarizer():
```

```
    global sortedtheta, sortedinten
    correction_array= numpy.zeros(4194304, dtype=float)
    factor1=float(Entry.get(factor))
    A=(1-factor1)/(1+factor1)
    counter=0
    while counter < 4194304:
        correction_array[counter]=((1+A*math.pow
        ((math.cos(sortedtheta[counter])),2))/(1+A))
        global sortedinten
        sortedinten[counter]=sortedinten[counter]/correction_array[counter]
        counter= counter+1

def eliminate(bin_intensity_array, sigma, mapped_pic_index1):
    global sigmanum, keepit_pix, keepit_inten
    midpoint1=numpy.median(bin_intensity_array)
    lower_bound= float(midpoint1 - 0.5*sigmanum*sigma)
    upper_bound= float(midpoint1 + 0.5*sigmanum*sigma)
    locale=numpy.where((bin_intensity_array>=lower_bound)
     & ( bin_intensity_array<=upper_bound))
    keepit_pix.extend(mapped_pic_index1[locale])
    keepit_inten.extend(bin_intensity_array[locale])
#    print bin_intensity_array[locale]

def freq_occurance(k, sigma1, midpoint1, mapped_pic_index1):
    frequency=[]
    intensities=k
    last=int(numpy.size(k)-1) # upper bound
    sorted_bin=sorted(k)
    n=sorted_bin[0]# lower bound
    o=sorted_bin[last] # upper bound
    #print 'upper and lower intensity bounds:', n, o
    bin_intensity_array = []
    total = int(round(o-n))
    counter2=1 # TO COUNT THRU THE WHOLE INTERVAL;
    while counter2 < total:
        intensity_value = float ((round(n)-2)+counter2)
        lower_bound= float(intensity_value - .5)
        upper_bound= float(intensity_value + .5)
```

```
        locale=numpy.where(( intensities>=lower_bound)
        & ( intensities<=upper_bound))
        mini_bin= intensities[locale]
        bin_size = numpy.size(locale)
        frequency.append(bin_size)
        bin_intensity_array.append(intensity_value)
        counter2 = counter2 + 1
    freq_vs_inten= numpy.dstack((bin_intensity_array, frequency))
    frequency_statistics=eliminate(k, sigma1, mapped_pic_index1)
    #print 'frequency vs intensity:', freq_vs_inten

def bin_averaging():
    global sortedtheta, keepit_pix, keepit_inten, sortedinten
    global lower_bound, sortedpic_index
    global upper_bound, sigmanum, trash, keepit
    keepit_pix=[]
    keepit_inten=[]
    trash=[]
    sigmanum1 = (Entry.get(sigmanum))
    sigmanum=float(sigmanum1)
    delta = float(Entry.get(delta_input))
    delta2theta= float(delta)
    halfdelta=float(delta2theta/2)
    counter=1
    counter_total= int(numpy.amax(sortedtheta)/delta)+1
    locale=[]
    bin_avg=[]
    arrayindex=[]
    while counter < counter_total:
        print counter
        midpoint = float (delta2theta*counter)
        lower_bound= float(midpoint - halfdelta)
        upper_bound= float(midpoint + halfdelta)
        locale=numpy.where((sortedtheta>=lower_bound)
         & (sortedtheta<=upper_bound))
        mapped_pic_index=sortedpic_index[locale]
        mapped_bin=sortedinten[locale]
        mapped_theta=sortedtheta[locale]
```

```
        bin_sum=float(numpy.sum(mapped_bin))
        bin_size = float (numpy.size(locale))
        if bin_size ==0:
            counter = counter+1
            bin_average= float(bin_sum/1)
            sigma=math.sqrt(bin_average)
            #print 'empty bin', midpoint
        else:
            bin_average= float(bin_sum/bin_size)
            sigma=math.sqrt(bin_average/bin_size)
            freq_occurance(mapped_bin, sigma, midpoint, mapped_pic_index)
            arrayindex.append(midpoint)
            bin_avg.append(bin_average)
            counter = counter + 1

def maskmaker(): # zero is black white is one
    global keepit_pix, keepit_inten
    mask=numpy.zeros((4194304), dtype=int)
    mask[keepit_pix]= 1 #good intensity location
    #print 'mask', mask
    mask_data=numpy.resize(mask,(2048,2048))#double check indexing
    numpy.savetxt('mask.txt', mask_data)



root = Tkinter.Tk()
root.wm_title("Boolean_Mask")

Button(root, text='Select Any File in the Directory',
 command=directoryfinder).grid(row=2, column=0, columnspan=1)

Button(root, text='Add Data Images and Add Dark Images',
 command=addimage).grid(row=6, column=0, columnspan=2)

Button(root, text='Subtract Dark from Data Images',
 command=finalimg).grid(row=10, column=0, columnspan=2)

l1 = Tkinter.Label(root,
```

```
 text="Length of Square Pixels in microns")
hpixel = Tkinter.Entry(root)
l1.grid(row=20, column=0)
hpixel.grid(row=20, column=1)

l3 = Tkinter.Label(root, text="Sample to Detector Distance in mm")
S2DD = Tkinter.Entry(root)
l3.grid(row=24, column=0)
S2DD.grid(row=24, column=1)

l4 = Tkinter.Label(root, text="Wavelength in Angstroms")
wavelength = Tkinter.Entry(root)
l4.grid(row=26, column=0)
wavelength.grid(row=26, column=1)

l6 = Tkinter.Label(root, text="X-Pixel Coordinate of Direct Beam")
XPCoor = Tkinter.Entry(root)
l6.grid(row=30, column=0)
XPCoor.grid(row=30, column=1)

l7 = Tkinter.Label(root, text="Y-Pixel Coordinate of Direct Beam")
YPCoor = Tkinter.Entry(root)
l7.grid(row=32, column=0)
YPCoor.grid(row=32, column=1)

l8 = Tkinter.Label(root,
 text="Rotation Angle of Tilting Plane in Degrees")
RotAng = Tkinter.Entry(root)
l8.grid(row=34, column=0)
RotAng.grid(row=34, column=1)

l9 = Tkinter.Label(root,
 text="Angle of Detector Tilt in Plane in Degrees")
DetTilt = Tkinter.Entry(root)
l9.grid(row=36, column=0)
DetTilt.grid(row=36, column=1)

Button(root, text='Create 2-theta Plot',
```

```
 command=twotheta).grid(row=40, column=0, columnspan=2)

Button(root, text='Match 2-theta to Intensities',
 command=combine).grid(row=44, column=0, columnspan=2)

l10 = Tkinter.Label(root, text="Give Polarization factor f ")
factor = Tkinter.Entry(root)
l10.grid(row=46, column=0)
factor.grid(row=46, column=1)

Button(root, text='Correct for Polarization',
 command=depolarizer).grid(row=48, column=0, columnspan=2)

l11 = Tkinter.Label(root, text="Give Bin width in degrees")
delta_input = Tkinter.Entry(root)
l11.grid(row=52, column=0)
delta_input.grid(row=52, column=1)


l12 = Tkinter.Label(root, text="Number of Sigmas")
sigmanum = Tkinter.Entry(root)
l12.grid(row=54, column=0)
sigmanum.grid(row=54, column=1)

Button(root, text='Find Binned Averages and Do Statistics',
 command=bin_averaging).grid(row=56, column=0, columnspan=2)

Button(root, text='Make Mask', command=maskmaker).grid(
row=58, column=0, columnspan=2)


root.mainloop()
```