2015

Image detectors for online mineral liberation analysis

Gregory John Harvey
greg.j.harvey@gmail.com

UNIVERSITY OF WOLLONGONG

AGREEMENT FOR DEPOSIT OF HDR THESIS IN DIGITAL REPOSITORY
KEY DETAILS

1.	Student Name	Gregory Harvey
2.	Student No	2789036
3.	Email address	greg.j.harvey@gmail.com
4.	School	Physics
5.	Faculty	Engineering and Information Sciences
6.	Supervisor	Marco Petasecca
7.	Title of thesis	Image Detectors for Online Mineral Liberation Analysis
8.	Keywords	List 4 or more keywords XRF, charge sharing, Medipix2, X-ray Imaging
9.	*Field of Research (FoR) codes (optional) See explanation below	List up to 4 FoR codes (4 or 6 digit codes only)
10.	Restricted Access The University is committed to making HDR theses publicly available. If a restriction is required, please consult your supervisor prior to completing this section	Is a restriction required for this thesis, or part thereof? ☐ Yes ☒ No If Yes: a) Nominate restriction period: _____ months b) Provide details of the reason for the restriction: ☐ Publishing/patent ☐ Third party copyright ☐ Commercial /in confidence ☐ Other (please specify) _____
11.	Release of thesis automatically at the end of the expiry period (if applicable)	☒ Yes ☐ No (To ensure your thesis is deposited and accessible via the UOW digital repository at the expiry of the restriction period, tick Yes) If No, provide the reason why the thesis should not be released at the expiry of the restriction period. (Note: If you tick No, the Library will contact your supervisor at the end of the expiry period to request release of the thesis)
12.	Third party copyright material that has been used without consent (if applicable)	If the thesis contains material whose copyright belongs to a third party and has been used without consent: a) Would gaining consent to include the third party copyright material in the electronic copy of the thesis be onerous or expensive? ☐ Yes ☐ No b) Would removal of the copyright material compromise the thesis? ☐ Yes ☐ No

***Field of Research (FoR)** code classifications are used in the measurement and analysis of research and experimental development (R&D) undertaken in Australia,according to the field of research. In this respect, it is the methodology used in the R&D that is being considered. UOW uses FoR codes to classify research outputs, and has been used to report research data to the Commonwealth Government.

The FoR has three hierarchical levels, namely Divisions (at the broadest subject or discipline level), Groups and Fields (at finer levels). Students submitting their HDR thesis have the option to classify their research using the Group level (4 digit code) or Field level (6 digit code). All Divisions, Groups and Fields are assigned unique codes.

TERMS & CONDITIONS

1. Definitions

1.1. In this Agreement:

"**Digital Repository**" means Research Online or any other open access electronic repository operated by UOW or its agents;

"**UOW**" means the University of Wollongong;

"**Work**" means the thesis you are depositing including any abstract, object, text, image and related data;

"**You**" means the Candidate identified in item 1 of the Key Details;

"**Third party copyright**" means any material where the copyright is owned by another person

2. Your warranties

2.1. You warrant that:

(a) all or part of the Work is your original work;

(b) the Work does not violate or infringe any copyright, trademark, patent other rights of any person;

(c) if you have used another person's copyright work or intellectual property rights:

(i) you have obtained permission from that person for the use of this material as contemplated by this agreement; or

(ii) you have confirmed in item 10 of the Key Details reasons you have not obtained permission to use or to grant to UOW the licence contemplated by this agreement;

(d) if the Work has been commissioned, sponsored or supported in any way, you have fulfilled all obligations required by such contract or agreement;

(e) the Work may be deposited in the Digital Repository without breaching the intellectual property rights of any third party provided UOW complies with any restrictions identified in items 8 and 9 of the Key Details.

3. UOW rights

3.1. You grant UOW a perpetual, non-exclusive, worldwide, royalty-free, sub-licensable licence to:

(a) deposit the Work in the Digital Repository;

(b) distribute copies of the Work (including the abstract) in electronic format;

(c) electronically store and copy the Work; and

(d) convert the Work to any format necessary for the preservation of the Work or the deposit on the Digital Repository.

3.2. UOW:

(a) may remove the Work from the Digital Repository for professional or administrative reasons at its sole discretion;

(b) is not under any obligation to take legal action on your behalf in the event breaches of your intellectual property rights stem from the deposit of the work in the Digital Repository;

(c) is not under any obligation to reproduce, transmit, broadcast or display the work in any particular format; and

(d) is not liable for any loss or damage to the Work whilst stored in the Digital Repository.

3.3. If you have used another person's copyright work without consent as identified under item 10 of the Key Details UOW may:

(a) remove this material from the Work and deposit in the Digital Repository;

(b) deposit only the metadata of the Work in the Digital Repository; or

(c) elect not to deposit the Work in the Digital Repository.

I declare my digital thesis is the final version approved by the University for the award of the degree.

Student (candidate) signature	Supervisor signature
25/11/15	25/11/15
Date of student (candidate) signature	Date of supervisor signature

Image Detectors for Online Mineral Liberation Analysis

Gregory John Harvey

Submitted in partial fulfilment of the requirements
of the Masters of Science

November 24, 2015

School of Physics
University of Wollongong

Certification

I, Gregory J. Harvey, declare that this thesis, submitted in partial fulfillment of the requirements for the award of Masters of Science, in the School of Engineering Physics, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Gregory J. Harvey

Contents

List of Figures

Abstract

Mineral liberation is one of the key parameters driving the effectiveness of valuable material extraction. On-line analysis of the liberation is a challenging task requiring the characterisation of mineral particles at the micrometer level. A reliable on-line measurement of mineral liberation is yet to be successfully resolved. A device capable of measuring mineral liberation on-line would be a world's first.

Recently, semiconductor pixel detectors, originally designed for applications in high energy physics, have become available for other scientific, medical and industrial applications. These detectors offer exceptionally good spatial resolution, single particle counting, unlimited dynamic range and other advantages. Their properties are opening up the possibility to apply them to on-line mineral liberation measurement.

This work used the Medipix2 detector to measure the X-ray fluorescent energy of samples to determine elements present and by analysing the spectra in each pixel, spatial information of the elements in the sample are also obtained.

One of the major drawbacks using a pixel detector in this manner is the issue of charge sharing and a unique method to compensate for this problem is also presented.

The results of this work are extremely promising with both simple and complicated samples returning identification of minerals down to the 110x110 μm size.

Acknowledgements

Although at times working on this thesis felt lonely and isolated, what follows would not be possible without the help of many of those around me.

Firstly, I must thank my project supervisor Dr Josef Uher. Without his support and guidance I would not have known where to even begin in this work and also, to all the researchers, staff and my fellow student Rhys Preston working at the CSIRO labs in ANSTO, thank you for all the help, support and friendship you provided me and in particular thank you for the many morning teas spent sharing cake.

At uni, my supervisors Dr Marco Petasecca and Dr Michael Lerch. Thank you for all the help and assistance you provided me in writing this thesis. You put the pressure on me when I needed it but also knew when I needed to be left to my own devices. Your corrections and suggestions were appreciated and the help you gave when dealing with the paperwork minefield was invaluable.

I am in debt to Melinda Tippett and the other staff from the University of Wollongong Southern Sydney Campus, without the late night access you provided me to your campus I would have had nowhere to write this thesis.

Thanks must be given also to Eric Brace for providing an ear to listen as I tried to wrap my head around explaining my research. By explaining it aloud I often found myself with the very words I was struggling to come up with sitting in front of a screen on my own. Also your continued push to review the flow of the narrative helped turn what was a series of disjointed ramblings into the cohesive structure it is now.

Thanks to everyone I worked with and for at NumberWorks'n'Words. To my manager Tim for always being flexible around the time I needed to spend on my thesis and to all my students over the years who have been such a joy to teach. To be able to spend some time each day working on something other than my thesis was important, that I found working with my students both fun and rewarding was an incredible blessing.

To all my friends, thank you for accepting that much of my time had to be given over to this thesis and thank you that in spite of me continually not making it to things you never stopped inviting me. To those few who would study with me late into the night, thank you for your company and for the constant proverbial whacks on the back of my head to stop me procrastinating and get back to work. To my friends who I game with, thank you for providing me countless hours of fun and for putting up with my vents of frustration, I will not thank you for introducing me to LoL which quickly became a constant source of temptation[1]. To my friends from church, thank you for never giving up on me and for never letting me give up, for at least feigning interest in the details of my thesis and most importantly thank you for all the prayer you provided me with over the years.

To my family, thank you for all your support and understanding. For putting

[1] I should thank the uni's IT department for designing the wireless network such that I could not play online games, otherwise said temptation would probably have won out most nights.

up with me living the most unusual hours and always being available for whatever support I needed. Thank you that you provided support despite being constantly kept in the dark regarding the progress of my thesis. I am sorry for the stress it caused you in fearing that this work would never be finished, but thank you for never pushing the issue.

Finally, and above all others, I give thanks and praise to God my Father and Lord Jesus Christ. Thank you for the wisdom, patience and perseverance to see this project through.

1 Introduction

1.1 The Problem

Mineral liberation is one of the key parameters driving the effectiveness of valuable material extraction. On-line analysis[1] of the liberation is a challenging task requiring the characterisation of mineral particles at the micrometer level. A reliable on-line measurement of mineral liberation is yet to be successfully resolved. A device capable of measuring mineral liberation on-line would be a world's first.

Recently, semiconductor pixel detectors, originally designed for applications in high energy physics, have become available for other scientific, medical and industrial applications. These detectors offer exceptionally good spatial resolution, single particle counting, unlimited dynamic range and other advantages. Their properties are opening up the possibility to apply them to on-line mineral liberation measurement.

1.2 The Goal

Within the framework of creating an on-line detector for mineral liberation analysis, certain requirements need to be achieved:

- Resolution: The conceptual position of this device would be at the end of the refinement process, analysing the tailings (waste). It is a safe assumption that processes already in place will take care of the large pieces of valuable minerals. This detector must be prepared to identify valuable particles at the micrometer size range.
- Realistic: If the only possible way of achieving reliable results at the required resolution is to take several hours for each run, then this device

[1] On-line analysis is when the analysis is done alongside or as part of the refinement process. It is meant to be done fast enough that the results can then influence the refinement process, for example maybe by altering the temperature or pressure to improve yields or in the case of this work, it would send the current batch of waste material back for further refinement as it contains sufficient quantities of desired material

would fail at being "on-line". It is unlikely to produce a real-time detector but if positive results can be achieved in a reasonable run time, then the tailings can be easily grouped together and analysed in small batches and depending on the results the batches can be sent for further refinement.

- Reliability: To be of use in a practical solution, this project needs to produce accurate results reliably. Without accurate results, success in the other two requirements would be meaningless.

1.3 Proposed Solution

The proposed solution, already published in the articles ([1], [2]), is to irradiate the sample and observe the resulting X-ray fluorescence (XRF) spectra using a hybrid pixel detector. When sufficient energy is applied to the sample, a number of different physical processes occur which can be used to identify the materials in the sample.

By placing a detector in the beam line but behind the sample, the transmission spectra can be analysed. Depending on what energies were absorbed and by how much, the material composition can be determined. By using a pixel detector there is also the benefit of creating a high spatial resolution image of the sample.

X-Ray Transmission (XRT) analysis is where a sample is illuminated by an X-ray source of a known spectrum. A detector is used to measure the spectrum after it has travelled through the sample where, depending on the minerals and thicknesses inside the sample, certain energies are absorbed more than others. By comparing the final spectrum with the known/open spectrum, it is possible to identify the minerals inside the sample.

An alternate method is to measure the fluorescent spectrum created by the irradiated sample. The energies of the peaks in the spectrum can be used to identify what materials are present in the sample. By using some form of x-ray optics and a pixel detector, spatial information can also be measured.

XRT analysis provides higher spatial resolution but XRF analysis has higher energy resolution. The proposed detector device will employ multiple detectors to combine both analysis methods.

1.3.1 Evolution of the Project

The work of this thesis took place over the period from the start of June, 2009 till the end of October, 2010 at the CSIRO facility located at ANSTO. The work was done within the CSIRO Process Science and Engineering division. This group's core focus at this site is on improving efficiency of processes like mineral refinement and the work of this thesis was a natural extension of their work using Medipix2 as a detector in X-ray Transmission (XRT) analysis for mineral identification in a waste slurry.

The main problem in XRT analysis is that of beam hardening [3]. This is caused by the fact that lower energies are going to be absorbed more noticeably than higher energies. Without accounting for this problem, any sample will appear to have higher amounts of low energy absorbing materials than it should.

Methods of calibrating for beam hardening are possible [4], but is was determined very early that it would be ideal to at least know what minerals are present in the sample and where they are roughly located to improve the beam hardening calibration analysis.

Since exposure time is a realistic constraint on the final product, an ideal solution would be one where the same exposure can be used for a second complimentary analysis. This is how the idea of combining X-ray Fluorescence (XRF) analysis with the XRT results came about.

XRF analysis is naturally linked with XRT analysis, the fact that each mineral absorbs a unique fingerprint of energy lines leads to the fact that each mineral also will emit (fluoresce) a unique spectrum of energy lines.

The advantage of XRF analysis is that it has excellent energy resolution compared to XRT, but it does not provide the high spatial resolution[2]that XRT analysis can give. This means that although XRF gives excellent information regarding what minerals are present in the sample, it can't give as accurate (or sharp) an image of the location of the minerals found in the sample.

Combining the two results was a logical conclusion as the physics behind both techniques will occur regardless of which method used. Neither method will interfere with the other and by first identifying what minerals are present in the sample using the XRF results, it makes it much simpler to create a sharp image of where the different elements are in the sample using the XRT results.

Although any energy detector could be used for the XRF analysis, using a second pixel detector like Medipix2 means that the spatial information gained can further improve the XRT analysis by allowing this analysis to be divided up into different sections depending on what minerals are present in each part of the sample.

1.4 Summary of Research Report

This present study focuses predominantly on the XRF analysis and our method of overcoming charge sharing to produce accurate information regarding mineral location in test samples.

The structure of this thesis is straight forward:

- In Part I, the theory behind X-ray fluorescence and charge sharing is explained as well as a review of the literature dealing with three main methods already presented that deal with compensating for charge sharing using Medipix2.

[2]It doesn't provide any spatial information without also using some sort of optics technique (in this case a simple pinhole setup was used).

- In Part II, the current proposed method of charge sharing correction is presented and the experimental setup described. A per-pixel method of calibrating the energy scale of Medipix2 is presented and this part finishes by explaining the key details of the analysis process and how it takes the raw data and converts it to an image of different minerals present in a sample.
- In Part III, the results are presented and conclusions are given.

Part I

REVIEW

2

The Challenge of Reliable, Realistic Results

2.1 X-ray Fluorescence

X-ray Fluorescence (XRF) is a phenomenon were a sample emits photons of specific energies when it is sufficiently excited by an external energy source [5](such as gamma rays and electron beams). In the current case, the samples have been irradiated by electromagnetic radiation within the X-Ray spectrum. When the sample is excited by this irradiation, certain energy values are absorbed to push electrons up to higher energy levels. The energy values absorbed depend on the elements found in the sample, with no two elements having the exact same set of energies. The excited electrons will drop back down to lower energy levels and as they do this, they emit a photon with an energy value depending on the levels at which it started and finished. As each element has a unique signature of energies it will absorb, they also produce a unique set of energy values when the electrons drop back down [6]. It is this property of the process which enables the identification of the elements in the sample by radiation detecting technologies, such as the Medipix2. Whilst the physical phenomenon is understood by the field, the challenges lie in developing technologies to accurately measure the emitted energy spectrum.

2.2 Medipix2

The Medipix2 detector is a silicon imaging detector with 256×256 pixels, each $55 \times 55\mu$m in size. The total active sensing area is 1.98cm[21].

The Silicon sensor is 300μm thick and is bump-bonded to an ASIC readout chip. Each pixel has its own electronics, this consists of an amplifier, energy discriminators and a counter.

The device works by setting an energy threshold value and any particles detected with energies equal to or less than the threshold will increase the

[1]This represents 87 percent of the entire chip area.

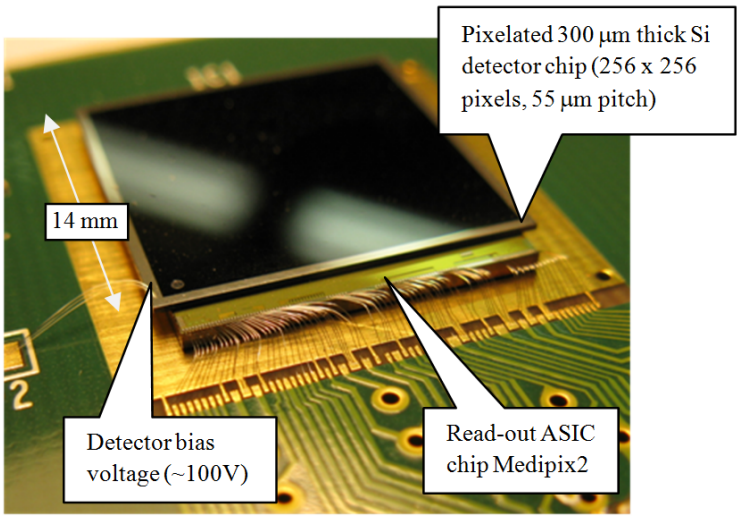


Fig. 2.1: Image of the Medipix2 Detector Chip

counter by one in the pixel it was detected by. By performing a threshold scan, where measurements are conducted using a series of increasing threshold values, an integral spectrum is created.

By using threshold settings, leakage current and noise are separated from the proper signal to prevent false counting [4]. This allows for an unlimited dynamic range and to reach an almost arbitrary SNR through the setting of exposure time. Due to the counting nature of the device, the result is absolutely linear [5].

A few different designs exist for connecting the detector chip to a computer, however the USB2.0 design provides the best capabilities [7]. In figure 2.2 on the following page, a screen shot of the panel used to control the detector is shown. The spectrum in the figure has been numerically converted into a differential spectrum for the display. The peak in the spectrum is created by the K_α energy line of the sample tested and counts due to noise are seen on the right of the spectrum. Medipix2 works in its own arbitrary units of threshold levels (THL) and it is a small peculiarity of Medipix that this means high energies correspond to low THL values and low energies correspond to high THL values, this feature is corrected when converting into standard units of Electron-Volts.

2.3 Charge Sharing

In figure 2.2 on the next page, between the energy peak and the noise level an undesirable low energy tail can be seen. This is an issue that affects all hybrid pixel detectors to some degree [8] and is due to the charge deposited in one pixel being shared between neighbouring pixels. The charge sharing effect must be dealt with for accurate identification of the energy peak as it also results in the

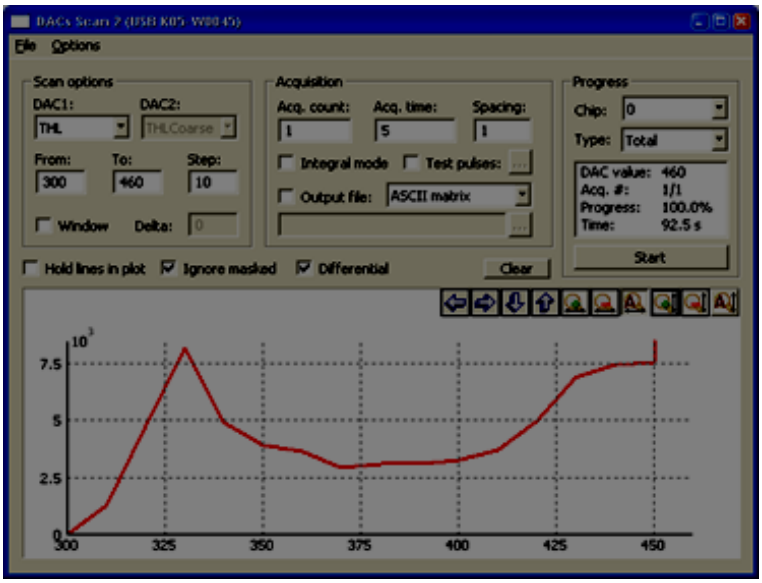


Fig. 2.2: Screenshot of the Medipix control software

observed energy peak shifting to a lower energy value [9].

2.3.1 The Problem

Charge sharing is a problem [10] with pixel detectors where the energy deposited in a specific photon spreads inside the sensor layer and instead of a single count of the intended energy, the detector measures several counts in a cluster of pixels of a reduced energy. This results in a low-energy tail of the measured x-ray spectrum and causes significant problems when trying to identify the different peaks associated with a compound sample of multiple elements.

2.3.2 Why does it occur

Charge sharing is due to the ionizing particle incident on the silicon sensor forming a charge cloud. Electrostatic repulsion and charge diffusion causes this cloud to spread and expand as it travels from the sensor layer to the collection electronics. If this expansion becomes large enough to spread to neighbouring pixels, then the original incident pixel will now detect an energy level lower than what it should and neighbouring pixels will detect energy where no incident photon existed.

A number of factors influence this problem with the physical dimensions of the pixel playing a critical part [11]. A detector which has large and shallow pixels will not be able to spread as much of the energy to neighbouring pixels before it is measured by the electronics and so the charge sharing effect will not be as significant. That is not a solution to the problem though since shallow

pixels will decrease the detection efficiency and large pixels will reduce the resolution of the image, so a physical solution to the problem is not available.

2.3.3 How does it affect the results from Medipix2

Due to the way Medipix2 operates, then charge sharing can have one of three results:

1. The charge sharing spread can cause the incident pixel to register an energy level that is lower than what it actually should be, but this drop of energy is not enough to fall below the threshold limit and the energy that spread to neighbouring pixels is not sufficient enough to be above the threshold limit. Although the charge was affected, only the incident pixel has its counter increased by one and this would have been the same end result if the charge sharing effect did not exist.

2. The incident pixel registers enough energy to be above the threshold level but enough charge has been lost to the neighbouring pixels that they also receive enough energy to count a hit. This causes a cluster of pixels to count a hit from one incident photon and it is this situation that causes most of the difficulties in analysing the spectra.

3. The charge spread is sufficient to prevent a count from being recorded in the incident pixel and is therefore[2]also insufficient to allow any neighbouring pixels to register any counts. This means that an incident photon is lost and not recorded.

Each of those situations mentioned above assumes a single particle incident on the detector; a more likely scenario is when many photons are incident on the detector and the pixels are gaining and losing energy at the same time. This makes the analysis of results difficult as a pixel with no incident photon might receive enough charge from several neighbouring pixels that it has a final energy level greater than the threshold value but each of the pixels that were originally struck might not receive enough final charge to count a hit.

This is dependent on the value of the threshold, the charge sharing effect is more prominent in the lower energy region where less energy is needed for a pixel to register a count. The effect this has on the overall spectra is a noticeable low-energy tail caused by higher energy photons being counted as multiple lower energy photons. The energy that ends up in the tail of the spectrum comes from the main XRF peak, which in turn is less intense and shifted slightly towards a lower energy.

It is this shifting of the energy peak that is the primary obstacle to material identification. All methods of using XRF for material analysis require a comparison of the energy spikes from the measured spectrum with the known energy

[2]Regardless of how much the charge spreads out, the original pixel will always get the highest amount of incident energy, so since the recorded energy in the incident pixel is not enough to exceed the threshold limit, then it is safe to say neighbouring pixels could not have received enough charge to put any of them above the threshold limit.

values created for each element in the sample or at least if one is searching only for certain elements, then the known energy values for each of them. For example, if one is investigating a sample for Copper, then some[3]of the energy spikes you would be looking for are at 8.027, 8.047, 8.905 and 8.977 keV. If the resulting energy spikes of the measured spectrum have been shifted (and worse not all to the same degree of shift), then without compensating for this shift, it is difficult to determine that this measured spectrum was created by a sample of Copper, at best, one could determine that it was caused by an element in the region of Copper. This becomes even more difficult to assess when dealing with a sample that could contain a number of different elements scattered within the sample (as would be the case in a real life situation involving the tailings of a mine).

A method of compensating for the charge sharing effect is therefore required for accurate identification of elements in the sample.

2.4 Overview of Current Approaches

A number of distinct approaches to minimise the charge sharing effect and recreate the original spectrum have been presented by different groups working with Medipix2. Each method have their benefits and weaknesses and a summary of three different approaches follows. These approaches range from the straight forward but time consuming method of manually measuring a complete library of responses to the more complex methods of simulating the response either mathematically or using a Monte-Carlo process.

2.4.1 Czech Technical University

This method [6] centres on using measured responses of the pixel detector to a range of single element samples. Using this created library of responses, it is theoretically possible to decompose any complex multi-element sample and determine the different elements that are present in the investigated sample. By decomposing the spectra of the investigated sample in each pixel, it is possible to form a spatial image of where the different elements are found in the sample.

They investigated the method using both Medipix2 and Timepix. Timepix is an evolution of the Medipix2 design [12], it has the same size and number of pixels as the Medipix2 chip but it allows the charge collected in each pixel to be measured[4].

This team was able to demonstrate that it is possible to identify elements with an atomic difference of 1 and they state that this method is applicable for identifying elements with atomic numbers greater[5]than Z=26 (Iron). By using

[3]There are actually are large number of possible transitions, resulting in different energies created, but certain transitions are more likely than others and so produce larger and more obvious energy spikes in the measured spectrum.

[4]Rather than simply keep a count of when a charge collected in each pixel exceeds a threshold value

pixel groups of 4×4 it was possible to create an image of where the different elements are located in the sample.

The theory behind this method is that in the ideal detector which is free of noise, energy smearing, charge sharing defects etc, the measured spectrum would consist of spikes at specific energy values depending on the characteristic transitions of the elements in the sample. Since the detector is not ideal, each of these energy spikes are smeared and create a charge sharing tail but the process is independent for each of the energy spikes. The final observed spectrum is then a sum of the individual spectra created by each energy spike.

By measuring the final spectra for each different element that could possibly be found in an unknown compound sample, it is possible to calculate the linear coefficients that each element in the library contributed to the final unknown spectrum. Due to noise it is not possible to determine these coefficients perfectly, so instead a least squares method is used.

The benefit of this method is that it is the most comprehensive. With the exception of noise, every process that alters the final spectrum of the unknown sample is accounted for because it is also altered the spectra of the library and so it should be the most accurate at identifying the elements in the unknown sample. However this benefit also is its main limitation for use in a practical purpose. To account for every process that alters the unknown spectrum, everything must be kept identical to when the library was measured. Measuring the library would have to be done on site and as there is no way of correcting the library for changes in the detector[6] the library will need constant updating.

2.4.2 University of Erlangen

This team, like the Czech group, demonstrate a method of correcting for charge sharing using a library of responses([13], [14], [15], [16], [17]). The difference between this method and the previous, is that here they simulate the library of Medipix responses rather than experimentally measure them.

They set out to use X-ray transmission (XRT) techniques to identify the type of materials and its positions inside a test phantom. XRT is another method for material recognition which works on the principle that each element will absorb different energies. So that if one were to know what the unobstructed spectra of the x-ray tube was and then measured the obstructed spectra, you would notice that certain energies are no longer present or at least much less intense. By comparing these missing energies to the known values of what energies are absorbed by different elements, it is possible to identify what elements are present in the sample that is obstructing the x-ray beam.

As charge sharing will affect the measured spectra, challenges arise when comparing the observed absorption values with known values and also as charge sharing shifts counts towards the low energy region, the absorptions are less prominent.

[5]Or in the case of using Timepix, they state that it is applicable for identifying elements with atomic numbers greater than or equal to 26

[6]For instance due to ageing or radiation damage

To counter this, a method of reconstructing the original x-ray spectrum by accounting for and removing the charge sharing effects is presented. The method involves understanding how Medipix2 will react to different incident energies (they worked with energies from 5 - 150 keV in steps of 0.5 keV). With access to few monoenergetic sources, they instead created ROSI [13] an X-ray simulation tool for virtual imaging setups. This allows them to use Monte-Carlo simulations to create Medipix response over the range of energies.

For their Monte Carlo simulations, they had to accurately reproduce the Medipix2 detector digitally. Charge sharing is caused by a number of different phenomena and it was not sufficient[7]for them to only simulate the sensor layer and the charge spreading that occurs from the top of the sensor layer to the charge collection region. They also included the bump bonds and glue that is used in Medipix2, as these both created back-scattered secondary fluorescent signals and contribute to the overall problem described as charge sharing [14].

Their simulations proved accurate when compared to the measured monoenergetic spectra of the few energies they could create and now that they could simulate a wide range of response spectra for different input energies, the next step is to try and reconstruct the original x-ray tube spectrum from the measured spectrum and the library of monoenergetic responses they simulated.

They propose two different techniques to do this and both methods ultimately produce accurate results. Both methods start by knowing that in general, the deposited (or measured) energy spectrum, $N(E')$, in a detector is mathematically the result of a convolution of the incoming spectrum, $S(E)$, with the detector response function, $R(E', E)$[8].

$$N(E') = \int_0^\infty R(E', E)S(E)dE \tag{2.1}$$

Determining the incoming spectrum, $S(E)$, is the goal but it is what cannot be directly measured. $N(E')$ is the measured spectrum obtained from the derivation of a threshold scan and the response function, $R(E', E)$, is simulated using their Monte Carlo technique over a range of incoming photon energies using a set energy bin width[9], δE.

So, for a known response function, R, and a measured spectrum, N, it is possible to deconvolve the integral equation to obtain the initial spectrum, S. It is at this point that the authors demonstrate two different mathematical methods to determine the initial spectrum, S:

[7]Although they knew that it was important to simulate the bonds and glue, they also simulated the response curves with out these components present and showed how it was inaccurate enough for recreating the response functions for energies greater than about 30 keV (this energy region is below the k-edge of tin or silver and so do not create back-scattered x-rays)

[8]Where E and E' represent the primary photon energy and the deposition energy respectively. The value of $R(E', E)$ is the probability that an impinging photon of energy E causes an energy deposition of E' in one pixel

[9]In this case, they used an energy bin width of 0.5 keV and simulated the response for energies from 5 to 150 keV

- Spectrum Stripping: Converting equation (2.1) on the preceding page into a set of linear equations, an iterative method is used to calculate scaling factors for use with the simulated spectra responses to subtract out the measured counts in each energy bin. Once all bins are reduced to zero the scaling factors form the desired initial spectrum with charge sharing effects removed.

- Matrix Inversion Method: Writing equation (2.1) on the previous page in vector form ($\tilde{N} = R\tilde{S}$), an inverse of R is calculated to solve the equation for $\tilde{S}$. More precisely they specify that since a true inverse is unlikely to exist, they apply a pseudo-inverse to the measured spectrum N which results in a best estimation of the unknown incident spectrum S in a maximum-likelihood-sense. A drawback of this method is that the quality of the result is related to the independence of the response coefficients $R_{ij=const}$. This creates a requirement that the energy bins need to be chosen large enough such that these coefficients differ sufficiently from each other [17].

Of the two methods to calculate the incident spectrum, they state that the spectrum stripping method proved to be more robust for their purposes [17].

Using this method of simulating the response functions, they demonstrate that it is able to reliably determine incident X-ray spectra [16]. As their method can be performed in individual pixels it also allows for material reconstruction imagery [15] and to demonstrate this they use a phantom consisting of a PMMA (acrylic glass) cube partly filled with a 1.2% Iodine solution and also includes an Aluminium sheet to cover a region of the cube.

Using the known components of the phantom as basis materials for the reconstruction images they were able to demonstrate good results with a separate image for each material and in each image on the region of that material producing any noticeable signal, although in the Aluminium image an inverted ghost image of the PMMA is visible, with a possible cause being Compton scattering.

This process has the benefits of being very accurate like the previous method, but as the Medipix responses are simulated, it is not required to remeasure the library if the geometry is changed. It is still a very time consuming task to create the responses using Monte-Carlo simulations and as time passes key simulation variables will need to be changed due to, for instance, the detector ageing otherwise the accuracy of the reconstructions will begin to decrease.

2.4.3 European Synchrotron Radiation Facility

Research presented from the ESRF [18], provided a method to restore incident energy spectra from the pixel detector's raw data. Unlike the previous methods discussed, which relied on creating a library of responses, here they showed an analytical method to suppress the charge sharing tail and significantly enhance the resolution of energy peaks[10].

[10]When compared to the un-altered spectra

They are able to reconstruct incident spectra for a variety of sources[11]. For the most part, their results were obtained using the average counts over grouped pixel areas, they do finish by stating that they were able to reconstruct the spectra of a 15 keV monochromatic beam using data from single pixels.

The development of their iterative deconvolution method begins with understanding what the charge sharing effect would have on a spectrum from a monoenergetic source of energy E_0. A simple energy response model was presented to calculate the measured spectrum density, $n(E, E_0)$:

$$n(E, E_0) = (1 - k) \times n_p(E, E_0) + k \times n_{cs}(E, E_0) \tag{2.2}$$

It consists of two parts, the first being the unsplitted charge component, $n_p(E, E_0)$, which was assumed to take a Gaussian distribution with an integral value of 1 and a width of w to represent the spectral enlargement due to noise and gain variations between pixels[12].

The second component was for the charge sharing tail, $n_{cs}(E, E_0)$, which was assumed to be a first-order approximation that the splitted charge events have an equal probability in the energy range $[0, E]$, so:

$$n_{cs}\,(E, E_0) = \begin{cases} \frac{1}{E_0} & : E < E_0 \\ 0 & : E \geq E_0 \end{cases} \tag{2.3}$$

The parameter k represents the probability of charge sharing and needs to be evaluated. They used Medipix2 to measure the responses of the detector to two different beam energies of 15 and 8.62keV for three different Si sensor thicknesses of 300, 500 and 700μm.

By fitting experimental data they were able to estimate the charge sharing probabilities, k, of 0.67, 0.78 and 0.88 for the three different sensor thicknesses.

With values of k determined, an analytical expression for the measured spectrum, $N(E)$, can be derived from equation (2.2) to cover responses from any input spectrum:

$$N(E) = (1 - k) \int_{-\infty}^{+\infty} n_p(E - E_0) \times \Phi(E_0) dE_0 + k \int_{E}^{+\infty} \frac{\Phi(E_0)}{E_0} dE_0 \tag{2.4}$$

Since equation (2.4) is linear with respect to the incoming spectrum Φ, it can be written using a transfer function, H, as:

$$N = H\Phi$$

Which in the finite space approximation, can be written as:

$$N_i = \sum_{j=0}^{N-1} H_{ij}\Phi_j \tag{2.5}$$

[11] synchrotron beams, X-ray tubes and radioactive sources

[12] Another simplification they made here was that w was assumed constant over the energy range in interest

H_{ij} represents the contribution from incident photons in the j^{th} energy bin to the results in the i^{th} measured energy bin. The coefficients of H_{ij} are calculated from equation (2.4) on the preceding page by approximating the integrals by finite sums.

An iterative inversion algorithm is used to calculate the desired incident spectrum, Φ:

$$\Phi_{i+1} = \Phi_i + \mu(N - H\Phi_i) \tag{2.6}$$

The algorithm is initialised using $\Phi_0 = N$ and for $\mu = 2$, the convergence of the series is obtained usually after $40 - 100$ iterations. For equation (2.6) to properly converge it was found necessary to impose positivity and smoothness to the Φ_i vectors.

This analytical method is both simple and fast at restoring the incident energy spectra from the raw data of a pixel detector, but it is not the most accurate. The author admits that more precise models are available and better results may be obtained if the model is improved, however as a first proof of concept the model used was sufficient.

For imaging applications this method was shown to have some use, it was demonstrated that the incident spectra of a 15 keV monochromatic beam was able to be reconstructed using the data from single pixels of Medipix. However in this example, the pixel signal was 6000 counts at minimum threshold. This count rate is unrealistic for the proposed situation of this thesis, from an XRF source with lead pinhole to achieve such high statistics would require several hours. To provide perspective, figure 5.2 on page 32 shows an example integral spectra of a single pixel, here the number of counts did not reach 250 at any threshold point.

2.4.4 Summary of the Presented Methods

A number of different methods have already been presented to the community so far, each working on the idea to correct the charge sharing effect by understanding how it would affect the original spectrum and then working backwards with the altered spectrum to reproduce the original. It is in the methods used to understand and characterise the charge sharing effect that each method differs.

The straight forward experimental method presented by the group from the Czech Technical University [6], measured the charge sharing effect on a variety of single element samples that were stored in a library of responses. By virtue of it physically measuring the response, it will provide the most comprehensive reconstructions as every process that affects the measured spectra is present in the library of responses. It however is also the most time consuming method to measure all the responses for each possible element and if the setup geometry is altered, then the library would require remeasuring or the accuracy would decrease.

Rather than measuring a library of responses to each possible element, the group from the University of Erlangen [15] demonstrated that is is possible to

use Monte-Carlo simulations to create the responses of Medipix2 to individual energies. Although not as accurate, as they can not simulate every possible process that alters the spectra, they do demonstrate that it is possible to simulate the key process that cause charge sharing and are able to recreate the incident spectra from each pixel. Although accurate, the method is also the most complicated and will still need to re-simulate the response library as time passes due to changes to the detector (e.g. from ageing).

As an alternative to both groups method of creating a library of responses for fitting the measured spectra, the team from the ESRF [18] presented an analytical method of suppressing the charge sharing. This method only requires a few initial parameters to be experimentally determined before it can begin acquiring data for material identification and so is also easier and quicker to adjust the fitting equations as the detector ages. The equations are not the most comprehensive and so the results will not be as accurate as the other methods, but these are only first concept equations which employ are number of simplifications and with refinement the accuracy would improve. This method also was shown to be of use only with high statistics (over 6000 counts in the minimum threshold scan alone) and makes this impracticable for online use with an XRF source.

Although each method produce accurate results to varying levels, none seemed to be carried out with the restriction of time and only the direct experimental method was performed using XRF results[13]. In the next part, a method is proposed attempting to combine the advantages of the high accuracy from the Monte-Carlo method with the simplicity and speed of the analytical model of charge sharing.

[13]Which produce considerably less counts than from XRT results

Part II

METHODOLOGY

3

The Detector Response Matrix

As has already been discussed in the previous section, the problem of charge sharing and its treatment has been approached by different groups with different methods [6], [18], [17]. The Monte-Carlo simulation, see subsection 2.4.2 on page 11, provides better agreement in the charge-sharing region compared to the analytical method, see subsection 2.4.3 on page 13, but on the other hand, the analytical method is much faster and simpler than the full simulation.

The method proposed in this thesis, and also in these articles [1], [2], attempts to combine together the advantages of both approaches; the high accuracy from the Monte-Carlo simulation [17] with the simplicity and speed of the analytical method [18]. The proposed method closely mirrors the method used by the Czech group, see subsection 2.4.1 on page 10, in that a library of responses are compared to the measured spectra in each pixel (or small pixel group) and the linear combination coefficients are calculated to indicate the original incident energies emitted by the sample. Where the two methods diverge is in how the library of responses are generated. They experimentally measured the entire library, whereas the presented method here sets out to analytically create the library using some experimental results to correctly calculate the various parameters.

Extensive study on the charge collection in pixelated Silicon sensors has been presented already (e.g. [11], [12]). This work was done using Timepix, a descendant of the Medipix2 detector. Timepix has the same size and number of pixels as Medipix2, but it has the ability to measure the charge collected in each individual pixel[1] and therefore can be used as a tool for the charge collection characterisation. The main design factors[2] of the device; pixel pitch (100μm), applied bias (typically 100V) and sensor thickness (300 μm) are the same, so the results from Timepix are still valid when using the Medipix2 device.

The two important conclusions from the presented work with Timepix were that the charge spread can be described using a 2D Gaussian distribution [5] and that the width of this Gaussian[3] was changing linearly with the depth of

[1] This allows one to observe the cluster of pixels that register charge due to a single incident photon.

[2] Relating to what affects the charge spread.

[3] And so also the cluster size.

photon interaction [11].

It should be noted that although the ability of Timepix to measure clusters of charge is very useful, it is not ideal for online spectral measurement due to its readout speed. To analyse the detected clusters correctly, cluster overlaps must be avoided. This requires very short exposure times and results in a significant dead time [11] as the whole detector matrix must be read out for each short frame.

3.1 Creating the Detector Response Matrix

3.1.1 The Theory

The effect of the charge sharing on the x-ray spectrum measured with the Medipix2 can be calculated as follows:

Divide the Silicon sensor into L layers and assume[4] that the spread of the charge is limited to an area of 3×3 pixels [11].

One can calculate the i^{th} layer convolution as:

$$S_i = G_i * P \tag{3.1}$$

Where G_i is the 2D Gaussian describing the spread of charge for photon interaction in the i^{th} layer of the sensor and is determined by the equation:

$$G_i\left(x,y\right) = \frac{e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma_i^2}}}{\iint\limits_A e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma_i^2}}\,\mathrm{d}x\,\mathrm{d}y} \tag{3.2}$$

The parameters x_0 and y_0 are the center of the 3×3 pixel array. σ_i is calculated as the linear interpolation between values in the top ($i = 1$) and bottom ($i = L$) layers of the detector:

$$\sigma_i = \left(\sigma_{i=L} - \sigma_{i=1}\right)\frac{i-1}{L-1} + \sigma_{i=1} \tag{3.3}$$

A restriction to the values of $\sigma_{i=1}$ and $\sigma_{i=L}$ is that since the charge spreads more if the photon is absorbed in the top most layer[5], then $\sigma_{i=1} > \sigma_{i=L}$.

The function P, in equation (3.1), represents the central pixel of interest that the charge sharing is being calculated for and can take on the following

[4]This assumption is possible because the observed cluster size for the anticipated energy range (7 to 30 keV) was less than 3 pixels.

[5]This is a common sense assumption as the charge has to travel the whole length of the sensor layer before being collected into a pixel electrode if it is absorbed at the top of the sensor layer. This means that this charge has more time and room to spread out to neighbouring pixels than a charge created at the bottom of the sensor layer.

two values depending on[6] x and y:

$$P(x,y) = \begin{cases} 1; x \in \left\langle x_0 - \frac{p}{2}; x_0 + \frac{p}{2} \right\rangle \bigcap y \in \left\langle y_0 - \frac{p}{2}; y_0 + \frac{p}{2} \right\rangle \\ 0; x \notin \left\langle x_0 - \frac{p}{2}; x_0 + \frac{p}{2} \right\rangle \bigcup y \notin \left\langle y_0 - \frac{p}{2}; y_0 + \frac{p}{2} \right\rangle \end{cases} \tag{3.4}$$

The function S_i in equation (3.1) on the previous page, is therefore a function of (x, y) and represents the proportion of deposited energy in the central pixel when a photon strikes the position (x, y) within the 3×3 matrix of neighbouring pixels.

If the middle of the centre pixel is hit, the full energy is deposited in that pixel[7].

If the photon strikes the edge of the pixel, then a cluster of 2 or 3 pixels are created and the central pixel would only collect part of the deposited energy it was meant to collect.

If a photon absorption event does occur in the i^{th} layer at position (x, y), then the energy detected in the central pixel is:

$$E_{ij}^{D}(x,y) = E_j S_i(x,y) \tag{3.5}$$

E_j is the energy value of the input spectrum in the j^{th} input energy bin. To generate the final detected spectrum, the energies E_{ij}^{D} are calculated for all layers (D) and positions (x, y) across the 3×3 array of pixels. These numbers are binned into the calculated spectrum $\Xi(E)$ with weights:

$$w_{ij}(x,y) = \Gamma_j e^{-\mu_j \rho \frac{j-1}{L-1} d} \tag{3.6}$$

- d = the thickness of the Silicon sensor $(300\mu m)$
- Γ_j = the X-ray intensity in the j^{th} bin of the input spectrum
- μ_j = the X-ray mass attenuation coefficient of Silicon at the energy E_j.[8]
- ρ = the density of Silicon

If $E_{ij}^{D}(x, y)$ is lower than a set threshold of E_{TH}, the weight $w_{ij}(x, y)$ is set to zero. The resulting spectrum is finally convolved with electronic noise, so that the final measured energy spectrum is given by:

$$\Xi_{FIN}(E) = \int_0^E \Xi(E - \varepsilon) \frac{1}{\sqrt{2\pi}\sigma_e} e^{-\frac{\varepsilon^2}{2\sigma_e^2}} d\varepsilon \tag{3.7}$$

Where σ_e is the detector resolution given by the readout electronics and fluctuations in the charge carrier generation [14].

[6] The restrictions on P essentially ensures that the calculations apply for only values of x and y that lie within the current pixel, the parameter p simply represents the pixel size and in the case of Medipix2 $p = 55\mu m$.

[7] clusters of only one pixel are possible and have been detected in Timepix.

[8] This excludes Rayleigh scattering that does not generate electron and thus does not lead to detection of the photon.

These equations take into account that the photons with lower energies are more likely absorbed in the upper layers of the detector. These equations do not yet take into account the effects of photon scattering and fluorescent photons created by the bump-bonds and glue behind the sensor layer, which as shown by the Erlangen group in subsection 2.4.2 on page 11, is an important factor to the final measured energy spectrum. They did, however mention that for energies below about 30 keV the simulation of the sensor layer alone was sufficient as no backscattering occurs in this energy range, [14]. So with the samples to be tested it is safe to not include the contributions from the bump bonds.

The advantage of this analytical approach over a Monte-Carlo simulation is in the possibility to use the described functions for fitting measured spectra. The parameters such as $\sigma_{i=1}$ and $\sigma_{i=L}$ from equation (3.3) on page 19 and σ_e from equation (3.7) on the preceding page can be determined from the fit and can be adjusted experimentally for cases were the sensor properties have changed[9] without the need to wait for lengthy simulations to be recreated.

3.1.2 Implementation

The described charge sharing calculations were coded into DataMaster, an in-house data acquisition and analysis software package being developed by Josef Uher at the CSIRO [19]. DataMaster is a Java coded package that consists of a number features useful for spectrum fitting using Medipix2 and can be used as either a standalone program or have its functions called externally by other programs.

For this work, only the creation of a response matrix was used from DataMaster and it was called from within Matlab, the code for which can be found in appendix C.1 on page 80. The time taken to create the response matrix depends largely on the number of bins used[10], it however does not interfere with the Realistic (time) requirement as the response matrix can be created prior to its use and loaded when needed.

For simplicity, the energy bins of the output spectra are set to be the same as the energy bins of the measured spectra and since these energy bins can differ between different samples no single response matrix can cover every sample. To avoid creating a new response matrix for each sample, a careful naming convention is used to include the minimum, maximum and step size of the energy bins when saving the response matrix. A check is then made of the created matrices to see if any cover the same energy bin range and step size, otherwise a new matrix is created and saved.

A detector response matrix was generated using the DataMaster software and parameters of $\sigma_{i=1} = 11.8\mu m$, $\sigma_{i=L} = 1.4\mu m$ and $\sigma_e = 0.591keV$[11]. An image of the matrix can be seen in figure 3.1 on the following page, the horizontal

[9] For example due to radiation damage of the sensor and readout or even by just applying a different bias to the device.

[10] and obviously the specs of the computer used.

[11] These parameters were calculated as average numbers of the values determined by fits of individual spectra.

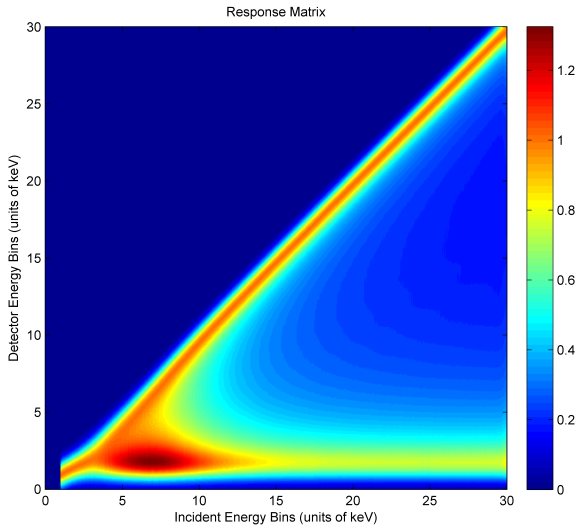


Fig. 3.1: The Response matrix: A matrix of generated spectra to simulate the detector response for a single incident energy value. The ideal detector response would only consist of the prominent diagonal line, however due to charge sharing some of the response is spread out into a tail below the diagonal line. The prominent horizontal line at the bottom of the image is to simulate noise.

scale consists of the incident energies (low to high energies from left to right) and the vertical scale is for the Medipix energy bins (low to high energy bins from bottom to top). Each vertical slice of the image represents the theoretical spectra of Medpix2. The diagonal stripe of response represents the XRF peak in the spectrum, the responses below the stripe represent the charge sharing tail which then begins to peak again due to detector noise at the bottom of the image. An example slice is shown in figure 3.2 on the next page, this was a vertical slice taken from the response matrix at 15 keV on the horizontal axis.

3.1.3 The Final Product

The XRF imaging analysis used in this thesis was qualitative rather than quantitative and therefore Medipix's detection efficiency was not precisely measured and incorporated into the response matrix. It was realised in the course of this work however, that to accurately identify the materials in a sample, a scaling correction was needed. This took the form of:

$$C(E) = \frac{1 - e^{-\mu(E)\rho d}}{1 - e^{-\mu(E_{min})\rho d}} \tag{3.8}$$

This scales the spectrum to account for the fact that low energies get absorbed more than high energies. E_{min} is the lowest energy of the response

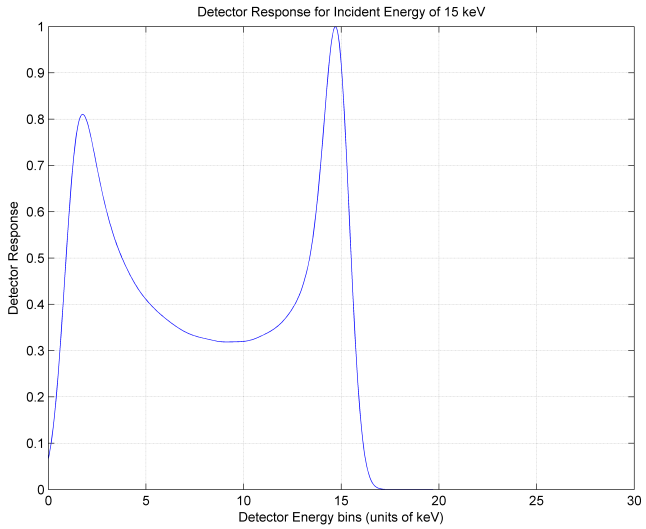


Fig. 3.2: Example Spectra from the Response Matrix: A vertical slice is shown from the response matrix taken at an incident energy of 15 keV. The left of the spectra corresponds to the bottom of the response matrix.

matrix, μ is the mass attenuation coefficient of Silicon[12], ρ is the density of Silicon and d is the thickness of the Medipix2 sensor ($300\mu m$)

Future versions of Datamaster's response matrix calculations will include this step automatically, but for now it is done post creation in Matlab and afterwards the response matrix is ready for use in analysis.

[12]This parameter also excludes Rayleigh scattering

4

Experimental Setup and Process

4.1 The Setup

The experimental setup is shown in figure 4.1 on the following page. It consists of a Hamamatsu X-ray tube, the Medipix2 detector and a lead pinhole. The X-ray fluorescent photons emitted by the sample are projected onto the Medipix2 detector through the pinhole.

The entire detector is enclosed in lead shielding to prevent scattered radiation from reaching the detector and the entire setup is enclosed in a shielded cabinet for obvious safety issues. The detector and sample holder are both mounted on separate adjustable plates which were motorised and controlled by computer to allow adjustments to occur whilst the X-ray tube was active. These adjustable plates allowed for movement in all 3 axes.

The spatial resolution of such a setup is primarily determined by the pinhole diameter used [6]. The smaller the diameter, the sharper the image but conversely, it also then requires a longer scan time to achieve sufficient statistics. A compromise then must be made between spatial resolution and experimental run time.

The approximate distances used were 9cm for the sample-to-pinhole and 15cm for the pinhole-to-detector.

The voltage of the primary X-ray beam was 100kVp and the current was 200 micro-Amps.

An unavoidable drawback of the geometry in figure 4.1 on the next page, is that the detector is aligned perpendicularly to the primary X-ray beam. This means that the sample needed to be rotated 45 degrees and introduced a parallax distortion to the XRF image. This had to be done due to a constraint of the X-ray cabinet available, but will be one of the first changes made to future setups.

The samples were prepared by mounting small pieces of different metals onto photo projector slides which then neatly clipped into place on the sample holder. This provided a convenient way of switching between samples without having to remove any of the pieces, however it meant that it was very difficult to ensure

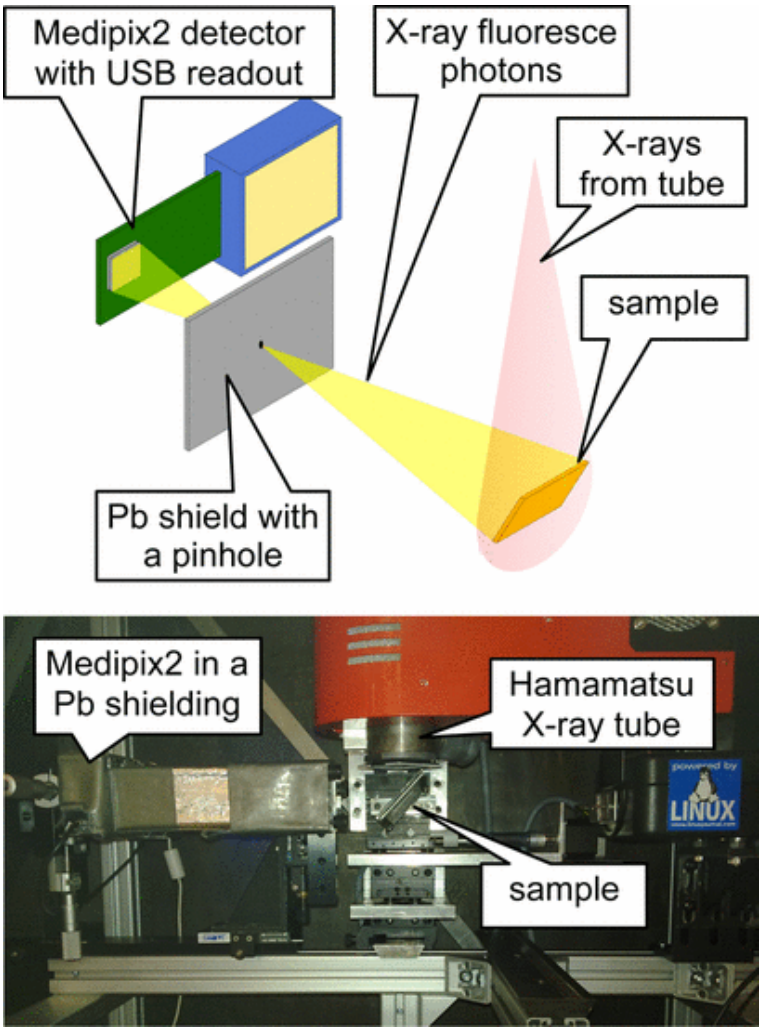


Fig. 4.1: Geometry of the experimental setup: The sample is irradiated with X-rays from the X-ray tube. Fluorescence photons are imaged on the Medipix2 detector through the pinhole. The detector is shielded by a Lead encapsulation to prevent scattered X-rays inside the cabinet from entering the detector.

the sample was mounted in the exact same place each time[1].

[1]So it was impossible to combine data from different experiments of the same sample once it was removed from the mounting.

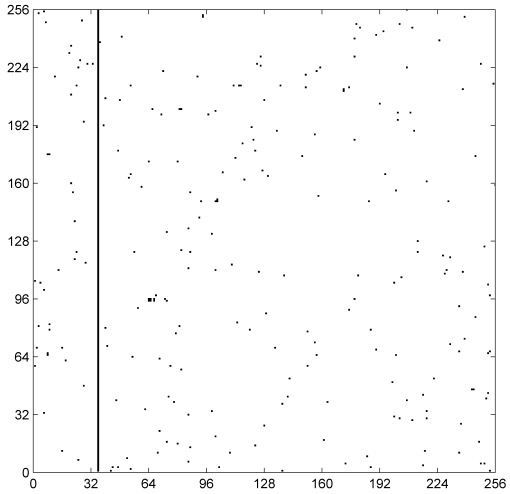


Fig. 4.2: Map of the known dead pixels

4.1.1 Our Medipix2 Detector

It should be noted that the Medipix2 Device used for this work was not brand new and not in the best of conditions. Having been used in many different experiments in its lifetime, this detector has been exposed to a varied number of situations involving x-rays, fast neutrons and ions. As would be expected it has suffered some radiation damage[2] and has lost a number of pixels. In figure 4.2, a map of the dead pixels is shown, this is generated by the Medipix2 Pixelman software but is created by the user manually painting the locations of dead pixels. These dead pixels were accounted for and removed in the analysis stage which is described in subsection 6.2.2 on page 38.

The fact that this detector has suffered some radiation damage and is still able to produce excellent results only highlights the advantages to the proposed method of charge sharing characterisation as the charge sharing parameters used can easily be recalculated periodically to account for the detector lifetime.

4.2 Operation Modes of Medipix2

Medipix2 operates by counting the number of events that occur with energies greater than a set threshold value and can work in a variety of modes.

[2]This radiation damage is not specifically factored in to any of the analysis calculations or the generation of the response matrix

4.2.1 Live/Continuous Data Collection

In this operation mode, Medipix2 will start counting the number of events with energies over the set THL value and will immediately display the results as an image. By setting a scan time, Medipix2 will record and display the counts that occur in that time frame and then reset all the counters to 0 and start again.

This mode is useful for double checking what is in the field of view of the detector and since the sample and detector are mounted onto translation stages that are motorised and controlled through the computer, it is simple to adjust the position of the sample so that it is centered in shot.

This mode is the first step in measuring an energy spectrum of a sample in a laboratory setting as it is necessary to check that the sample is correctly placed.

4.2.2 Single THL value Image

This mode works much like the Live mode and is an adjustment of a few settings. Having made sure that the sample is correctly positioned, by setting the time of the scan longer and clicking the option to save the data, Medipix2 will now create an image of the number of counts at the set THL value and save the data for analysis later if desired. This would be done for a number of purposes:

1. To check that a sufficient number of counts are recorded in all pixels. This would normally be done before taking a full THL scan and so the scan time is set to the same as that to be used in the full THL scan and the THL value would be set to a value below where the noise becomes noticeable.

2. By determining what THL value corresponds with the jump in number of events (caused by the peak K_{α_1} line in the sample's energy spectrum), an image taken at a THL value slightly lower than this peak and slightly above this peak can be used to show the spatial position of the elements in the sample (as it will become much more intense in the second image). For samples with multiple elements present, repeating the process for each element in the sample allows a crude identification of where each element is located in the sample.

4.2.3 THL Scan

In this mode the starting and finishing THL values are set and a step size is chosen. The detector will then take an image at each specified THL value and save the data in separate files. Loading the count values saved in each file into a program like Matlab (or similar) will create the measured integral spectrum of the sample. Whilst the integral spectra is to be used in the later analysis stages[3], the results can be better interpreted from the differential spectra as the reader can easily identify the peaks and tails. The differential spectrum can

[3]As this spectra is less affected by the low count rates in each energy bin, hence its spectra is smoother and thus easier to fit

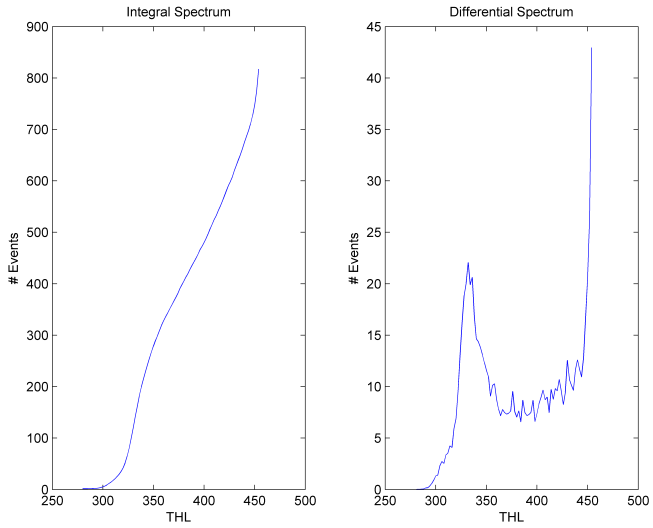


Fig. 4.3: The measured mean integral spectrum of Sn and its corresponding calculated differential spectrum. The differential spectrum is generated from the differences in counts between subsequent energy bins of the measured integral spectrum. Note that the Medipix2 THL scan reverses the energy axis such that high THL values correspond to low energies, which is why the noise limit is on the right.

be calculated from the integral spectrum by calculating the difference between counts in each bin. An example spectrum of Tin is shown in figure 4.3.

It is this operational mode of Medipix2 that will be most used and all results presented are created from a THL scan.

5

Per-Pixel Energy Calibration

Analysis of any x-ray energy spectrum measured by Medipix2, like that shown in figure 4.3 on the previous page, requires that the results be first calibrated from the arbitrary THL units of Medipix into the standard energy units of electron volts. This is necessary as identifying elements in a sample requires comparing observed energy spikes with accepted energy values found in a database like that from NIST.

An additional benefit of converting into units of keV's will make interpreting the spectra more logical since it will reverse the curve around and the observer does not have to keep reminding themselves that a high THL value equals a low energy threshold and therefore most photon interactions will trigger a count in the pixel and that electronic noise becomes a problem.

Calibration of Medipix2 results have already been demonstrated a number of times for other purposes(e.g. [20]), here they used two of the emission lines from a ^{241}Am source. They began by determining threshold values that both of the calibration energy lines occurred at and then calculated the fit parameters needed to describe a calibration curve through these two points. They confirmed their results using test points from a third emission line from the ^{245}Am source and it was demonstrated that the energy calibration was a linear relationship.

In this calibration method, they[1] created their calibration curve using results from all the pixels at once and so created a single energy calibration curve that applied to all the pixels. This is a significant assumption, since Medipix2 is essentially a collection of 256×256 separate detectors and although they should be all similar to each other, it is unlikely that each pixel will set the same energy threshold in units of keV for a given THL value.

Although more complicated, by creating unique calibration curves for each pixel, the results are improved substantially.

5.1 Process

The process of creating energy calibration curves is completed in three stages.

[1] And all others from what can be found in the literature

The first two stages are done separately for each sample and use the HYPERMET function to model the energy spectrum measured from the THL scan, a description of this function can be found in appendix A on page 71. Of most importance is determining the THL value of the main energy peak.

5.1.1 First Stage

The first stage uses the total energy spectrum[2] and the goal here is to determine accurate parameters of HYPERMENT to fit the total spectrum. This requires some trial and error by guessing at the initial parameters and observing how well they fit the curve. Once a reasonable fit is made, a chi-squared minimisation technique is used to refine the parameters and produce parameters that result in the best fit for the curve.

A chi-squared minimisation is an iterative process that uses the equation:

$$\chi^2 = \sum_i \frac{\left(E_i - F_i\left(k_1, k_2, ..., k_n\right)\right)^2}{\sigma_i^2} \tag{5.1}$$

Where

- $i = i^{th}$ data bin. So the χ^2 value is determined by summing over all the data bins.
- E_i = The experimentally observed value in the i^{th} data bin.
- $F_i\left(k_1, k_2, ..., k_n\right)$ = The predicted value of the i^{th} data bin, determined by the model used to fit the data with the parameters $k_1, k_2, ..., k_n$. In this case, the function is the HYPERMET function.
- σ_i = The error of the experimentally observed value in the i^{th} data bin. This value was simplified by making the approximation that the error of counts equals the square root of counts, so $\sigma_i = \sqrt{E_i}$.

The Matlab function FMINSEARCH was used to minimise the chi-squared value, where Matlab alters the parameters $k_1, k_2, ..., k_n$ slightly and observes if this alteration increases or decreases χ^2. If it decreases then the new adjusted parameters are better than the old parameters and so these values are then adjusted, if it increases then the new parameters are worse and so are rejected and a new adjustment is tried. It continues in this fashion until it can find no new adjustments which decrease χ^2. An example of such a fit can be seen in figure 5.1 on the next page.

This process does require that the initial guess parameters need to be close to the best parameters because it is possible for the minimisation to converge on a local minimum which is well away from the global minimum of χ^2. It is this reason why the first stage of this energy calibration is needed, as it is anticipated that no pixels are significantly different from the average and the results of this minimisation are to be used as the initial guess parameters in the next stage.

[2] i.e. A single spectrum is created using counts from all the pixels at each THL value

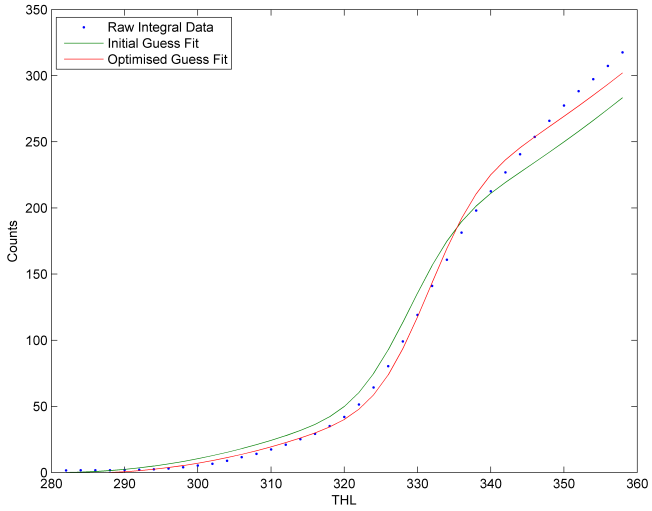


Fig. 5.1: Integral spectrum of Sn showing the initial guess fit and the optimised fit

5.1.2 Second Stage

This second stage follows much of the same process as the first stage, except that it is performed on each individual pixel. Analysing spectra from individual pixels has a significant complication since differentiation of the measured integral spectrum now requires subtraction of large numbers with a small difference, this leads to large statistical errors and a very jagged and difficult spectrum to try and fit as can be seen in figure 5.2 on the following page. For this reason, the integral spectra was fitted rather than the differential spectra[3]. The HYPERMET function was adjusted slightly so that it now fits the integral spectra but no other changes were made.

5.1.3 Third Stage

By the third stage, THL values for the position of the main XRF peak have been calculated in each pixel for each sample. What is now needed is to investigate each pixel and using the accepted energy values (in keV) from NIST, calculate the parameters for the best linear fit of the three points. An example of one of the pixel's calibration curve can be seen in figure 5.3 on the next page.

The mean calibration curve, created using the THL positions from the mean spectra of stage one, is shown in figure 5.4 on page 33. For pixels which either

[3]Although it might seem more logical then to use the integral spectrum in the first stage as well, the differential spectrum was still used because it is much easier for the human eye to spot how well HYPERMET is fitting the spectrum when it has obvious landmarks like the primary energy peak

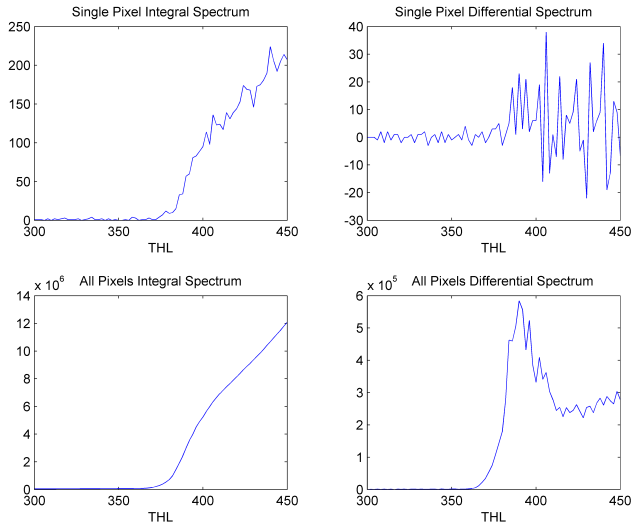


Fig. 5.2: Example of a single pixel integral and differential spectrum (top left and right). For comparison, the integral and differential spectra using all pixels are shown (bottom left and right).

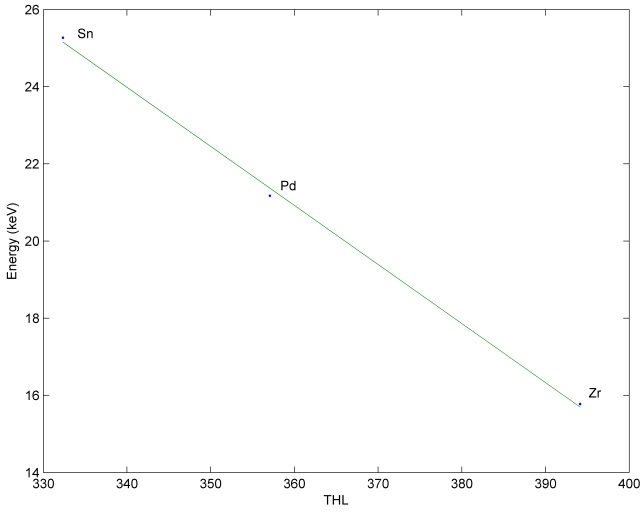


Fig. 5.3: Example Calibration Curve from a single pixel

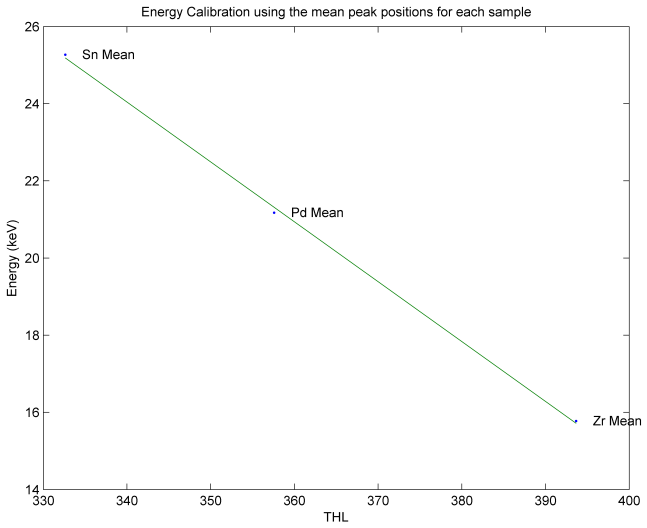


Fig. 5.4: The calibration curve using the peak positions from the mean spectra

were faulty or produced calibration curves so far away from the norm[4], the mean calibration curve was adopted instead.

The mean calibration equation for this Medipix2 detector was determined to be:

$$Energy = -0.16 \times THL + 76.80 \tag{5.2}$$

5.2 Results

Although significantly more complicated than generating a single calibration curve for all of Medipix2, the improvement to the spectra of samples is incredible. Figures 5.5 and 5.6 show the differences between using the mean calibration curve and calibrating each pixel separately. Not only do the primary peaks of each sample become more refined but also secondary peaks, caused by less probable energy transitions, can be seen after using a per-pixel calibration which were not visible in either the THL spectra or the mean calibration spectra.

Figure 5.7 on page 35, shows the range of different energies that each pixel sets when Medipix2 is set at a specific THL value. It is by compensating for this range of different results which helps improve the energy spectra so impressively[5].

[4]This could be for a variety of reasons, most often it was the case that for one of the samples the spectrum in that pixel was very poor but the automated process of fitting the

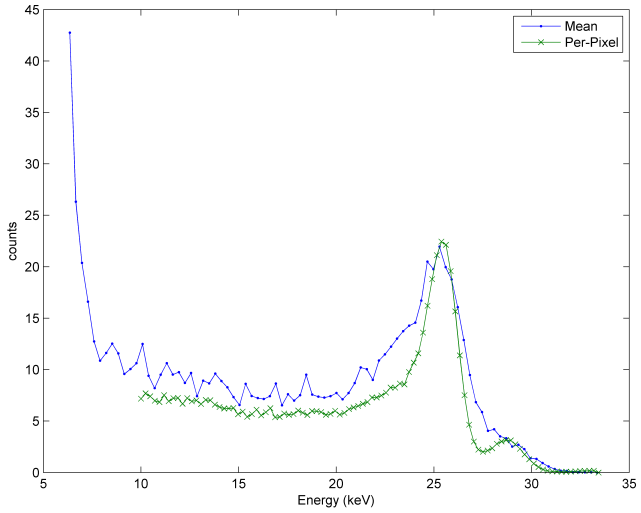


Fig. 5.5: Differential Spectrum of Sn comparing the mean and per-pixel energy calibration

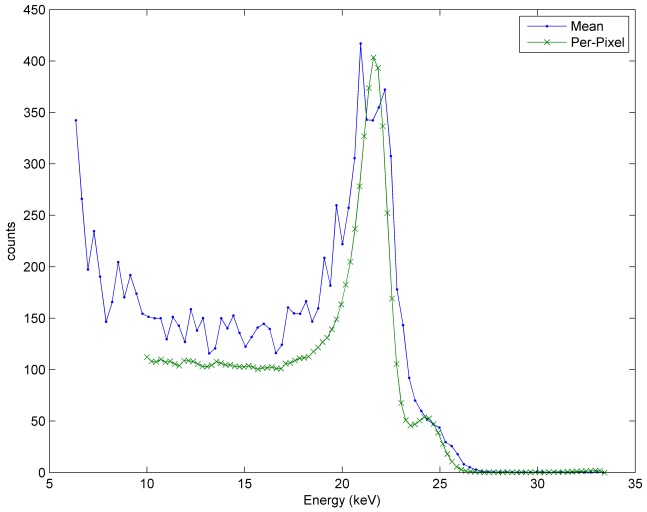


Fig. 5.6: Differential Spectrum of Pd comparing the mean and per-pixel energy calibration

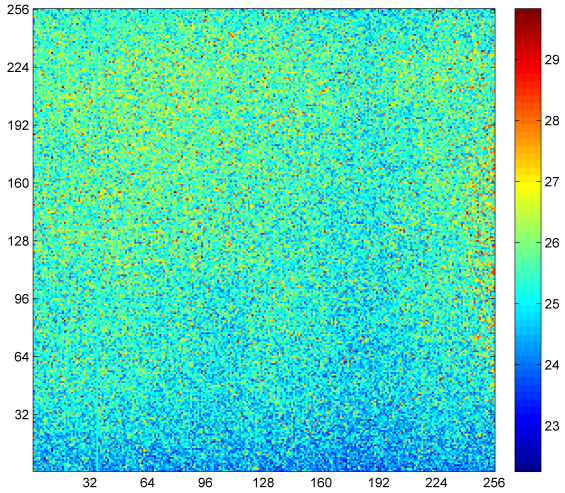


Fig. 5.7: Energy values in each pixel calculated from a test THL of 332: Colour here refers to the calibrated energy in keV. This range of responses in the pixels to the same THL value demonstrates why much of the finer detail is lost in the spectra when using a mean calibration.

The actual process of using the calibration curves is discussed in subsection 6.2.1 on page 37.

spectrum still found something it deemed acceptable.

[5]Or, conversely, it is this range of different energies that explain why the results can be so poor when not considered

6 Analysis Process

This chapter will outline the key stages that take the raw counts from the THL scan through to the final identification of minerals in a sample.

6.1 Overview of the Analysis Process

Before going into the specifics, an overview of the analysis stages shall be presented:

- Pre-analysis: This stage is a once off section which consisted of -
  - Reading into Matlab the raw counts from the Medipix2 saved data file.
  - Converting the data from the arbitrary THL units into units of keV.
  - Grouping the pixels to make a larger pixel detector (poorer resolution but increased number of counts).
  - Correcting for the dead pixels in each grouped set of pixels.
  - Checking for an appropriate Response Matrix and either loading it in or creating one from scratch.
- Element Identification: This stage of the analysis is where the response matrix is used to fit the measured spectra and the resulting coefficient values are used to determine what elements are present in the sample.
- Presenting the Results: Although simple for a computer to comprehend the results when presented in a matrix or list form, it is much clearer for the human operator[1] to view the results visually.

6.2 Pre-analysis

Prior to any analysis, some important steps must be done to prepare the data.

[1] And also you the reader

6.2.1 Converting Energy Scale from THL units to keV

The need and method of converting the energy bins has been discussed already in chapter 5 on page 29, in summary a calibration curve is needed to convert the energy bins used by Medipix2, units of arbitrary Threshold levels (THL), into standard units of electron volts (or specifically units of keV). The calibration curve is linear and for improved results each pixel has its own calibration.

Using the calibration curves is a simple matter of substituting the THL values into each pixel's calibration curve equation, however since Medipix2 sets it so that high THL values equal low energies, some tweaking of the data is also required to swap the order of the energy bins[2] so that it is in a logical order of increasing energy.

An unwelcome difficulty arises from using individual energy calibration curves for each pixel, as now there does not exist a consistent set of energy bins. This makes it very difficult to combine the results from neighbouring pixels to form pixel groups. Using a single calibration curve, each data entry in a pixel was considered from the same energy measurement and so it was a simple matter of summing the counts of neighbouring pixels together to create a grouped pixel count. Using separate calibration curves, it is unlikely that any pixel shares a common energy scale, meaning that the only way to combine the pixels would be to use an energy scale consisting of all the different energy bins arranged in order.

In theory this actually would be a very useful outcome as it would provide a form of sub-energy bin sampling[3], where the final spectrum consists of many more energy bins than were actually measured. In practice, it would require a sufficiently large number of counts in each pixel for any meaningful spectra to be produced and this requires a very long exposure time. Since, for this thesis, exposure time is a constraint and the low number of counts in individual pixels is the very reason for combining pixels together, an alternative solution is required.

Rebinning the Data

So that the counts from each pixel can be combined, they need to be mathematically rebinned into an arbitrary uniform set of energy bins. These bins were usually set from 10 to 30 keV increasing in steps of 0.25 keV. A linear interpolation was used to calculate the counts that would have been detected in the desired bins. The Matlab function `interp1` was used to perform this adjustment and figure 6.1 on the next page shows an example differential spectrum before and after this process.

[2] Remembering also to swap the raw data

[3] Much like the sub-pixel sampling used in calculating the spatial resolution in appendix B on page 73

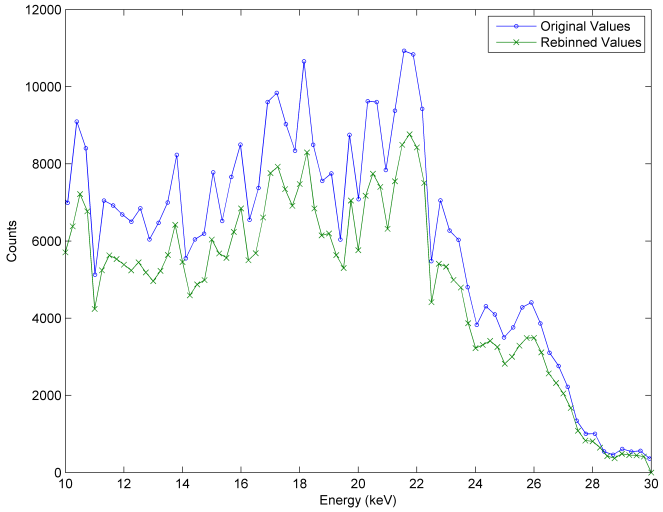


Fig. 6.1: An example differential spectrum before and after rebinning.

6.2.2 Grouping Pixels

It would be ideal to be able to analyse each pixel on its own and so produce a detailed spatial map of the materials present in a sample. However, unless an extremely long exposure time is used for every THL step of the scan, individual pixels do not receive enough counts to make it viable to work with them.

Under the principle that bigger pixels receive more counts, the pixels are sorted into larger groups. The analysis code was all written so that it was flexible enough to change the number of pixels grouped together by changing a single parameter. It was not designed to be so robust that any group size would work, only those sizes that divide neatly into a 256x256 array are allowed[4]

In figure 6.2 on the following page, the compromise between counts per pixel and spatial resolution of the image can be seen. This image is a slice of data[5] from the Pd, Ag, Mo, Sn sample. In each corner was a different element with a small gap between each of the elements.

As will be shown later, accurate energy spectra can be achieved by using groupings of 2×2 pixels and still provided decent spatial information of the sample.

The code used for this section can be found in appendix C.2 on page 82.

[4]From 1 super-pixel 256x256 pixels large, 4 128x128 pixels, 16 64x64 pixels and so on to 16384 2x2 pixels. Working with single pixels is possible, however the method to deal with dead pixels, as discussed next, will prevent analysis at this level.

[5]The slice was taken at 13.5keV

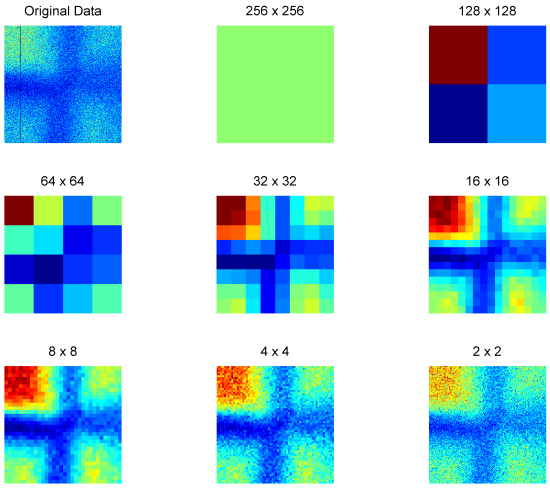


Fig. 6.2: Images of a sample after its pixels are grouped into the different possible sizes

Correcting for Dead Pixels

It is in this stage of the analysis that any dead pixels are corrected. Prior to this stage, any pixels that were identified as 'dead' have been recorded in the Pixel Mask file/matrix and all count values in the data have been replaced with Matlab NaNs[6].

Although a number of ways to address dead pixels were attempted, a simple approach was ultimately chosen since no obvious benefits were observed for the additional complexity and time. This approach consisted of replacing any dead pixel entries with the mean value from all other pixels belonging to that group. Because the dead pixels have entries of NaNs then the Matlab command `nanmean()` was appropriate to be used as this works out the mean value whilst completely ignoring any NaN values.

No consideration in the code was taken for the unlikely case that the entire cluster of pixels being grouped were dead, since this was not the case with the detector being used but this situation will need to be addressed in future work.

Also, due to this approach, the entire analysis process will not function properly if attempted without any grouping of pixels as the dead pixels will not be able to be fitted. However, until improvements are made to the detector count rate[7], this is not an issue since any attempts to fit the data without any grouping would likely result in poor results due to bad statistics.

[6]NaN stands for Not A Number and for this purpose can be thought of as a blank entry of the matrix

[7]One possible improvement mentioned in the conclusion section is to use multiple detectors appropriately placed, this would allow dead pixels from one detector to be replaced with mean

6.2.3 Generating and Adjusting the Response Matrix

The next step in the analysis is to first create the response matrix if it has not already been done[8]. As already mentioned, one of the advantages of the response matrix is that once it has been generated, there is no need to regenerate it again unless parameters need to be changed.

To simplify the analysis and reduce the number of times that values need to be rebinned, the response matrix was created using the same energy bin size and max energy bin as the data. It does not use the exact same energy bins as the data because generating the response matrix requires the energy bins start from 0keV. An image of a response matrix can be seen in figure 3.1 on page 22.

After generating the response matrix, it needs to be adjusted slightly to match the analysis that it will be used with:

- It is here that the response matrix is scaled to include the detection efficiency, as described in subsection 3.1.3 on page 22.
- It is required to convert it to an integral response matrix as the data is in an integral form[9].
- It must be cropped to cover the same range of energies used in the measured data. The vertical energy bins cover a larger range in the response matrix (0:b in steps of c) than the measured data (a:b in steps of c).

6.3 Element Identification

6.3.1 Process Overview

To determine the elements in the sample, the analysis is done in two stages.

In the first stage, all the pixels are grouped together and the analysis is done on the total spectrum. This stage was necessary so to define constraints for the small group analysis. A non-negative least squares method is used to fit the measured spectra with the response matrix. But, when analysing small groups of pixels, the counting statistics are very low (see figure 6.3 on the next page). Therefore, the number of free parameters in the least square fit needs to be restricted.

Using the results from the analysis of the total spectrum, the response matrix is limited to columns appropriate for the K_{α_1} and K_{β_1} energy lines[10] of elements present in the sample. No other constraints are applied.

values from the equivalent pixels of other detectors

[8] A check is performed to see if a response matrix has already been generated using the same energy bins as set in the energy calibration stage, if one exists then it is loaded instead of created

[9] And as mentioned in chapter 5 on page 29, it is better to work with integral spectra when using small groups of pixels as the differential spectra is too erratic and difficult to fit

[10] Plus a small variable window around each line, how the size of said window is set is discussed in subsection 6.3.3 on page 44

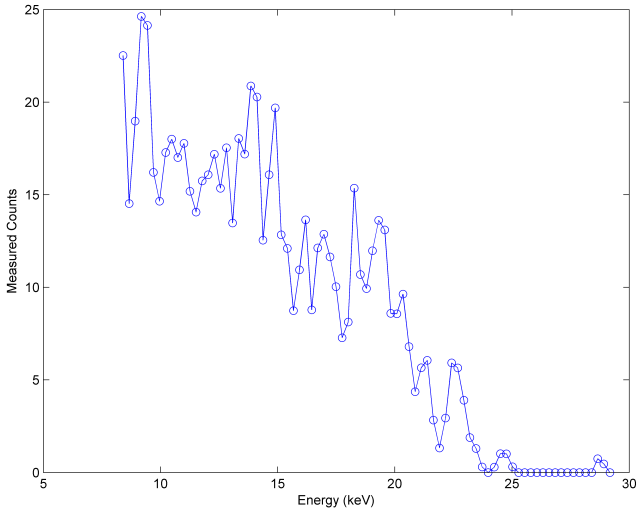


Fig. 6.3: Sample integral spectrum from a 2×2 pixel group.

In the second stage, the same analysis is redone but now using small groups of pixels (usually 2×2 or 4×4) and the constrained response matrix.

6.3.2 Fitting the Grouped Pixels

`lsqnonneg`, a Matlab inbuilt non-negative, linear, least squares fitting function was used to determine the linear coefficients. This function examines each calculated energy spectrum stored in the response matrix, shown in figure 3.1 on page 22, and attempts to determine how much of each energy spectrum in the response matrix went into making the measured spectrum. An example of the results from this least-squares fitting function is shown in figure 6.4 on the following page, this figure of energy spikes represents what the energy spectrum would resemble without charge sharing, noise and the other issues that alter the final measured spectrum.

In figure 6.5 on page 43, for each spike in figure 6.4, the corresponding response matrix spectrum is shown with each spectrum scaled according to the value of the determined linear coefficient. It can be seen in this figure how certain energies are more dominant than others, these theoretically correspond to K_α energy transitions of the different elements found in the sample. The less dominant spectra lines would then correspond to the less probable transitions with the few minuscule spectra lines most likely just anomalies caused by trying to fit noise in the data.

Finally, in figure 6.6 on page 43, all of the individual spectra are summed together to recreate the measured spectrum and this shows good agreement between the measured spectrum and theoretical spectrum created from the re-

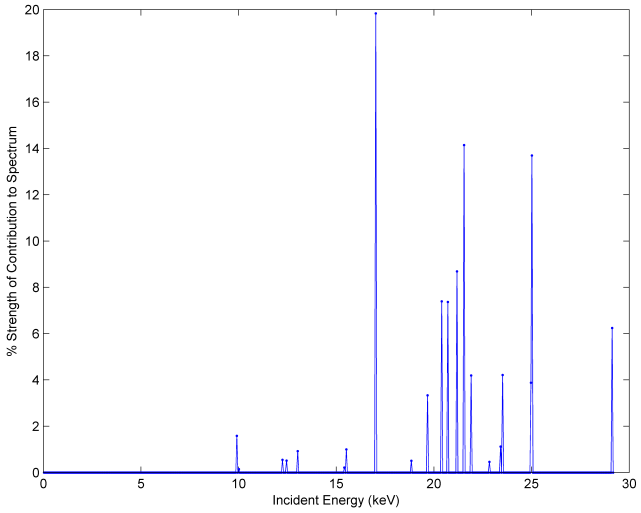


Fig. 6.4: An Example of the determined linear coefficients that when combined will best fit the measured spectrum. Each line indicates how much each simulated spectrum in the response matrix (shown in figure 3.1 on page 22) contributed to the overall measured spectrum. This also therefore represents the ideal energy spectrum of the sample when all negative affects of the detector (energy smearing, noise and charge sharing) are removed. By comparing this set of energy lines with known transition energies of different elements it is possible to come up with a possible list of different elements present in the sample.

sponse matrix.

The Matlab code used for this section is presented in appendix C.3 on page 83.

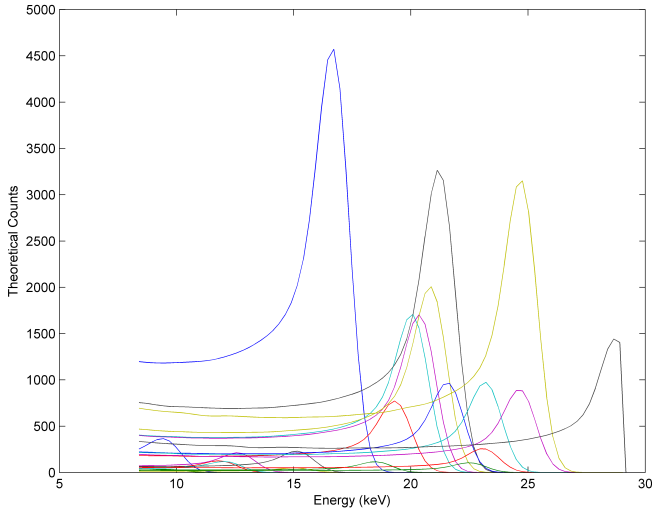


Fig. 6.5: For each energy spike in figure 6.4, the corresponding response matrix spectra are shown. Each spectra is scaled according to the value of the linear coefficient.

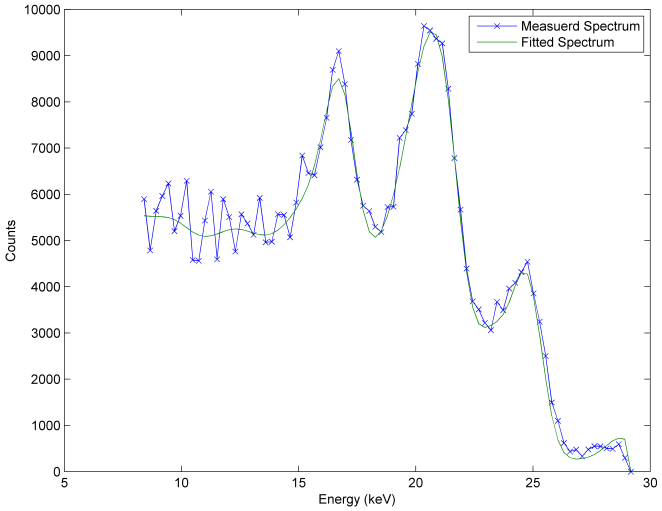


Fig. 6.6: The determined fitted spectrum (calculated by summing each spectra from figure 6.5) is shown alongside the original measured spectrum. This method was able to adequately fit the multiple main and secondary peaks in this spectrum caused by multiple elements being in the sample.

6.3.3 Determining Possible Elements Present

Entering this section of the analysis, Matlab has determined the original XRF energy lines in each pixel group. An example of the current results is shown in figure 6.4 on page 42. The next step of the analysis requires this list of calculated energies to be assigned to the appropriate elements and their transition lines, such that it is possible to visualise what elements are present in each pixel group.

When dealing with very large pixel group sizes, such that there are only a few sets of energy lines to examine, it is possible for somebody to eyeball the results and confirm what is present in the sample. However to get decent spatial resolution, then it is necessary to work with small pixel group sizes which result in a large number of energy line sets to examine. A set of rules were needed to be created to replicate what is simple for somebody to do by just looking at the graphed results.

Although not substantiated by hard scientific reasons, the following steps were created to keep track of how the elements present were manually determined in a sample. These steps fall into four basic categories:

- Initial setup of the determined results to identify the possible elements and transitions that caused each detected line.
- Assigning each detected line an appropriate element and transition from the list of possible transitions.
- Clearing any elements and transitions from the matrix of possible results as they are now not possible due to the assignment of an element and transition to the detected line.
- If it proves impossible to pick between a number of different possible elements and transitions for a particular detected energy line, then certain steps are required to break the impasse.

In more detail, these steps are:

- Before attempting to assign calculated energy lines with appropriate elements and transitions, the analysis runs through every calculated energy line and makes note of every transition for each element that could possibly be responsible for the calculated energy line. A library of transition energies is loaded and each calculated energy line is checked to see if it falls within a set window around each transition energy. This window is set by the user, however a requirement of the analysis is that the most intense energy line in each pixel group be caused by an element's K_{α_1} transition. A check is made that this is satisfied and if failed, the window is increased slightly and the process restarts from the beginning[11].

[11] It restarts from the beginning to ensure that each pixel group is working with the same window size.

- Once every possible cause of each energy line has been determined, all the energy lines are ranked by intensity and everything is ready to begin assigning elements to each energy line.
- Starting with the most intense energy line, the process will continue until all of the rows have been assigned an element and transition:

  1. Taking the current row from the results matrix, the locations of the 1's are identified.
  2. A possible transition matrix is created the size equaling the number of 1's detected. To begin with, every row has at least one possible transition, but as the process progresses, possible transitions are cleared for a variety of reasons that will be explained later. If no possible transitions remain on the current row then it is left unassigned and assumed to be caused by noise and not a real element transition.
  3. Going through each possible 1 on the current row, a number is stored in the possible transition matrix to indicate what type of transition it would be:
     - 1 = this transition is for an element which has already had its more probable transitions picked in earlier rows. The order of transitions[12] are K_{α_1}, K_{α_2}, K_{β_1}, K_{β_2}.
     - 2 = this transition is the K_{α_1} transition for an element and so represents a new element that could be detected.
     - 3 = this transition is for an element which has more probable transitions already picked but is not the next transition in order. For example the transitions K_{α_1} and K_{α_2} have already been picked but the current row's transition would be caused by the K_{β_2} transition which is not next in line.
  4. The priority order that the code will attempt to use is that if a possible transition falls under the type 1 category then it will be assigned that transition. If no type 1's exist, then it will search for any type 2's and finally it will search for a type 3 as a last resort. This priority order is arbitrary but logical. In the event that a row contains multiple possible transitions of the highest priority type present on the row, then a tie situation is required to be resolved (this is explained later in this section).
  5. Once the code has chosen and stored in the results matrix what element and transition is responsible for the current detected energy line, it then clears out any possibilities that can no longer be chosen (this also is to be explained later in this section).

[12] These are obviously not the only transitions possible, and if it is decided that more or less transition types are needed or even the order needs changing then the NIST database needs to only be changed.

6. The process is now done with the current row, so it goes back to the beginning and repeats until every row has been assigned an element and transition.

- Whenever a transition has been assigned for a row it is necessary to clear any remaining possibilities that no longer are valid:

  1. The simplest step in this clearing process is to remove any possibilities for the chosen element and transition that exist for later rows. This is because two detected energy lines can not be caused by the same element and transition. So the remaining rows in the chosen column are now set to 0.
  2. Next, before clearing any other possibilities on the current row, a check is made to see if any of these are K_{α_1} transitions. If any K_{α_1} transitions are present, then after they are cleared it should not be possible to pick any transition for that element until another K_{α_1} transition is assigned for that element. So not only is that K_{α_1} transition cleared but so too are any other transitions for that element until the next K_{α_1} transition is identified.
  3. Once the K_{α_1} checks have finished the current row is cleared of any other possibilities.

- In the event that a row contained multiple possible transitions of the same type (1, 2 or 3) then certain steps are needed to attempt to break the tie:

  1. A temporary version of the results matrix (with a layer for each choice possible) is created. It consists of the choices made so far on earlier rows and the remaining possibilities yet to be decided on the remaining rows.
  2. Using the first choice, the temporary results matrix (on the first layer) is assigned this choice and the remaining choices cleared out (using the same clearing rules already described).
  3. The current layer of the temporary results matrix is then passed back into the transition identification function[13] to identify the remaining rows.
  4. After it is finished assigning all the rows using the first of the multiple choices, the process repeats using the second and so on until all choices have been addressed.
  5. The final number of unique elements detected using each choice is determined.

[13]It was designed in such a way to allow this type of recursive use to cope with multiple tied situations

6. If a single choice returns the fewest unique elements[14], then this will be the final choice. If multiple choices return the same number of unique elements, then a second tie situation has occurred, however no obvious way exists to break the tie so it is arbitrarily assigned the highest element of the choices in the second tie.
7. Since at the end of this tie situation all the results have been assigned, then the RETURN command is used to break out of the function back to where it was called.

An example of a step-by-step demonstration of how these rules are used can be found in appendix D on page 84.

6.3.4 Restricting the Database of future Analysis

The first run through the analysis is always done on the results of the entire Medipix2 detector (i.e. a group size of 256x256). The reason for this is to restrict possible results in further analysis by limiting the possible elements present in each pixel to be only those that have been identified as present in the total image. This is not ideal as it means that if the sample consists of 99% element X and the remaining 1% is element Y, then the total spectrum would consist of mostly the spectrum of element X and the contribution from element Y might go un-noticed. Because element Y went un-noticed in the total spectrum analysis then when analysing the results using a much smaller pixel group size, even if element Y were clustered so that a pixel group consists only of element Y, it would return as a very poorly fitted element X spectrum as the analysis is not allowing for the possibility of element Y. This might seem a major restriction (which it is), but it is necessary, since when working with the much smaller group sizes (e.g. 2x2), the spectrum being analysed is noisy due to the limited counts in each pixel. The analysis can not understand all the noise and instead believe everything to be important and would return an extremely complicated result of a dozen elements not at all present in the sample. If it is possible to increase the counts in each pixel such that the spectra is much smoother, then this restriction will not be needed.

6.4 Displaying the Results

Although it would seem a trivial problem, the best way to present the results proves quite difficult. By this final stage of the analysis, the following has been calculated:

- Using the charge sharing matrix, linear coefficients of best fit have been determined for each grouped set of pixels and represent the incident energies detected. Each pixel group has stored a list of energies and a value of

[14]Which in turn means that each element detected will have the maximum number of transitions assigned per element

strength for how much each energy contributes to the measured spectrum in that pixel group.

- Each energy has also been labelled according to what likely element and transition type caused that energy. So now each pixel group has a list of energies, a value of strength for each energy, a Z value for the element that most likely caused the energy and finally a number to represent which of the many transitions of the determined element was responsible for the detected energy.

A number of different methods have been used to try and best represent as much of this information, but unfortunately, no single method is able to present all of this important information in a single image.

The simplest image that can be created is to go through all the pixel groups and locate the single, most intense transition and store the Z value that caused this transition into a separate matrix. Using the Matlab's built in `imagesc` command, a scaled image of this new matrix is created. The image dimensions are simply calculated by the size of the matrix and the colour of each pixel is set by the Z value.

A slightly more complicated version of this image is shown in figure 6.7 on the following page. This image follows much of the same process as mentioned, except that rather than look for the single largest transition, it looks for the element which contributes the most in total to the pixel group. This requires an extra step of first going through each pixel and collating all the transitions into a single entry for each element in that pixel.

This method has the advantages of being simple to create and simple to read as the `imagesc` command also automatically creates a colour scale legend that directly relates to the Z values. It does not however show any information regarding total intensity in each pixel compared to other pixels. This can be important to consider, for instance the sample used to demonstrate this method consisted of four separate elements (one in each corner) and a small gap of empty space between them. Interpreting figure 6.7 would lead one to assume that each element is connected with very non-uniform edges.

The next method of presenting the results attempts to include not only what main element was detected in each pixel but also to present how the intensity of measured responses compares to all the other pixels. Including this new information takes care of the problem involving empty space as the pixels covering these regions would not receive many counts since there are no elements to produce any. A 3D mesh was created with x and y values being simply the pixel row and column positions. In much the same way as the previous method, a colour map is created to correspond to the maximum intensity element in each pixel.

A brightness map is created by scaling each pixel to the pixel with maximum total intensity, thus the most intense pixel is coloured at full brightness and any pixel that had no counts would be coloured completely black. In figure 6.8 on page 50, the same sample as figure 6.7 is shown using this alternate image

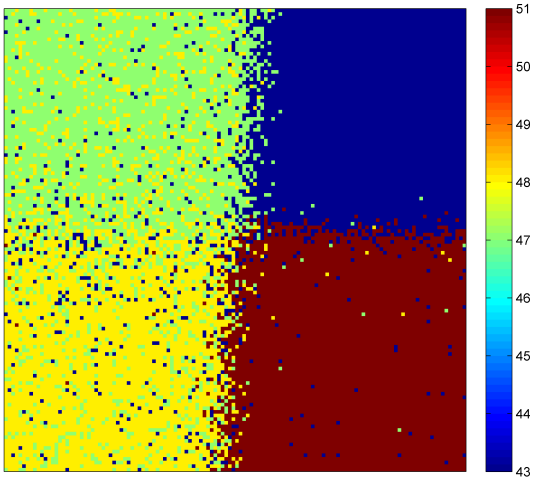


Fig. 6.7: Results of a sample highlighting the determined Z values in each pixel

method. Here the gaps between each corner can be made out and the same information about element position is still visible. Unlike the simpler method, a colour legend corresponding to Z value is not automatically created and is very difficult to manually produce.

Neither method presented can account for the fact that each pixel might actually consist of two or more overlapping elements. The only practical method to show this information was to create a separate image for each detected element. Going through each pixel and using the collated list of elements[15], the intensities of each element are converted to percentage contribution values in each pixel[16].

An image is created for each element detected in the sample, again using the `imagesc` command. Unlike the first method discussed, the colour values in these images consist not of Z values but the calculated percentage contribution values that element made to each pixel. This allows the observer to identify where an element is in a sample and how much it contributes to the pixel, even when the element in question does not make a dominant contribution to that pixel. This presentation method does not, however, allow for direct comparison of specific pixels[17]. Figure 6.9 on page 51 shows an example of this method. In the analysis, only four elements were detected anywhere in the sample. Normally, the code to calculate and generate these images produces a single image for each element

[15] As described earlier in this section

[16] i.e. Each pixel now has a list stating for example: Sn contributed 49%, Mo contributed 36% and Pd contributed 15% to pixel (42,24)

[17] i.e. An observer can not say from the multiple images that a specific pixel contains x% Copper and y% Lead

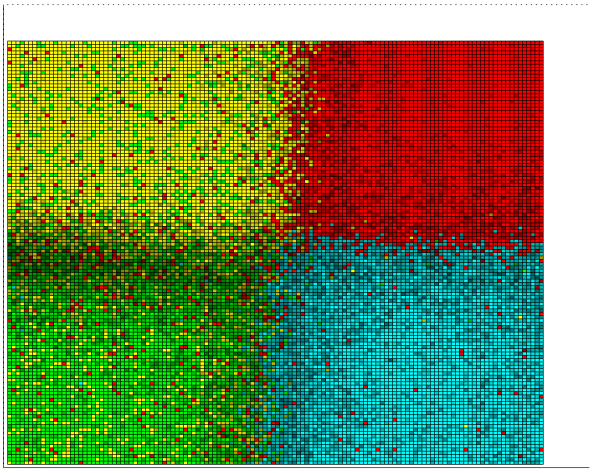


Fig. 6.8: Results of a sample highlighting the determined Z values in each pixel and scaling by the intensity of results in each pixel. Like figure 6.7 on the previous page, colour here represents different elements present.

detected, but for presentation here, the four images were compressed into a 2x2 sub-plot layout.

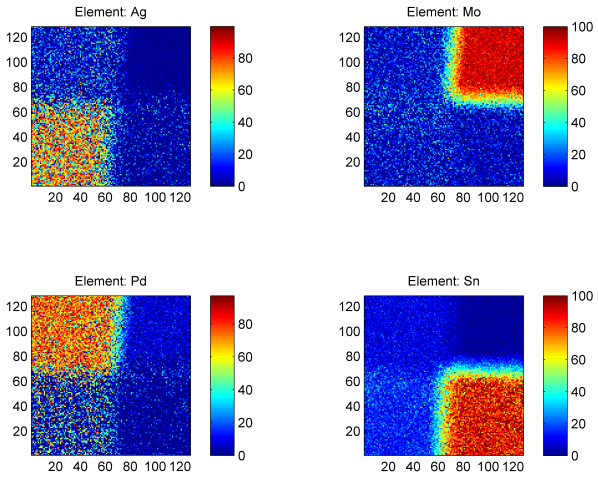


Fig. 6.9: Results of a sample creating a seperate image for each element found in the entire sample. The colour is determined by calculating the percentage contribution that element gave to each pixel.

Part III

RESULTS, DISCUSSION AND CONCLUSIONS

7 Results & Discussions

The following chapter examines the observed capabilities of Medipix2 and the already described analysis method, to distinguish between different elements present in a sample and of more importance the ability to correctly identify the elements present in a sample. For the analysis to remain as objective as possible, all the samples discussed were calculated assuming that the elements and positions of the elements were completely unknown.

It will be shown that Medipix2 and this analysis method can correctly identify the different elements in the sample even when the elements are only one atomic number apart, though it is not as effective at this when dealing with elements close to the detector's noise level[1].

The method is also demonstrated to be effective in situations where the sample has layered elements and elements found only in trace amounts.

7.1 Single Element Samples

To test the analysis method, many of the single element samples measured for other purposes were re-examined. These elements were measured using the large (4.5mm) diameter pinhole with the voltage of the primary beam set at 100kVp and the current at 200μA. Scan times took between 2.5 and 5 hours.

7.1.1 Empirical Energy Shift

One of the first observations from these samples was that none of them were working correctly. They would go through the entire analysis process perfectly and most of the samples would return consistent results identifying that indeed this image was of a uniform sample but of a wrong atomic number. For instance Tin(Z=50) was identified as Antimony(Z=51), Palladium(Z=46) came back as Silver(Z=47), Zirconium(Z=40) showed as Niobium(Z=41), the only sample tested which did not show this pattern was Germanium(Z=32) as it returned a result of Selenium(Z=34).

[1]This was around 7 to 8 keV and elements such as Ni and Cu could be identified but not accurately distinguished between.

The logical first place to investigate for the cause of this problem was in the analysis stage to determine which set of elements could be in the sample to produce the detected set of energy lines (see subsection 6.3.3 on page 44). Manually assessing the determined fit lines results agree with the offset results mentioned above. The problem is not therefore in this section of code, instead the energies of the determined fit lines are offset.

This makes the problem much harder to pinpoint and a number of different areas of the analysis could be the cause or even the problem could be as far back as the Medipix2 detector itself having a small calibration issue.

Although the problem could not be identified and fixed, a simple adjustment to the results has proven effective.

Taking the results of two single element samples, a linear equation was calculated using the determined fit line energies of each K_α line and the known transition energies from the NIST database. The two points used were[2]:

- Germanium - (11.4 keV, 9.855 keV)
- Palladium - (22.15 keV, 21.020 keV)

With these two points, the energy shift equation, in keV, was calculated to be:

$$ShiftedEnergy = 1.0387 \times DeterminedEnergy - 1.9862 \qquad (7.1)$$

This shifting of energy was done between rebinning the Response Matrix and culling parts of the Response Matrix so that it contains only bands of responses around the energy lines of all elements determined to be present in the sample from the 256x256 pixel grouping results.

This solution is not ideal[3]and future work should isolate where the problem is coming from. Although a simple fix, this adjustment works and all of the simple samples tested now return correctly identified elements.

7.1.2 Summary of Simple Sample Results

An example of the results for a sample of Zirconium can be seen in figure 7.1 on the following page, here the colour represents the different Z values found in the sample and the brightness represents the percentage intensity[4] of the K_α line in each pixel.

This graph was typical for all of the single samples tested.

Analysis of a single element sample is far removed from the likely samples to be seen in real life and also greatly simplifies the analysis process so that it

[2]The x-values are the determined energies and the y-values are the values from the NIST database.

[3]Again it should be re-emphasised here that, although it is not ideal to return incorrect element names, of more importance is the fact that this analysis can correctly distinguish between different elements and would always return the same incorrect elements for each of the input samples. Even in compound samples like Pd,Ag,Mo,Sn the Pd would return as Ag just as it did in a pure sample of Pd.

[4]Strength of the energy line in that pixel divided by the pixel with the maximum energy line present.

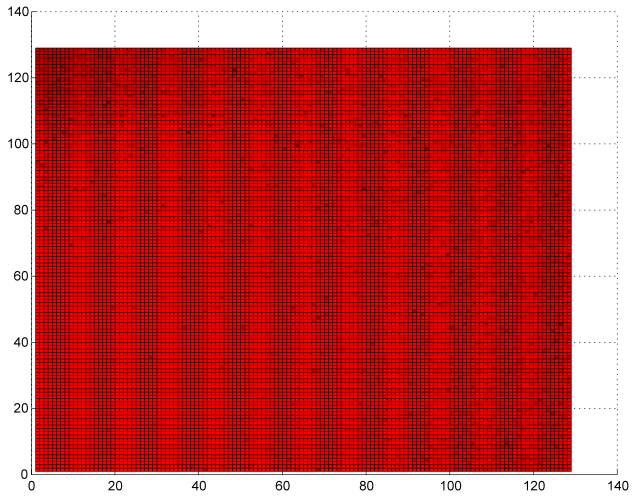


Fig. 7.1: Image of results from a sample of Zirconium. Colour here indicates different elements in the sample where the colour spectrum is evenly divided between all determined elements present in the sample. Red is the first assigned colour, so this image is showing pure Zirconium. Brightness in the sample indicates the intensity of the measured spectra in each pixel, dark pixels in the image indicate less counts detected in that pixel.

favours good results. In more complicated samples, the more elements present means more major energy spikes observed in the total spectra, which means more possible elements to choose from and more possible chances of inaccuracies.

The results from the single samples were helpful but not indicative of the usefulness this device and method would have with more complicated samples.

7.2 Compound Samples

Having determined that the analysis can correctly identify samples of pure elements, the next step was to see if the analysis could handle more complicated spectra created by compound samples.

Of key importance was determining if the sample can distinguish between neighbouring atomic elements, as the K_α lines of both elements would be close and some of the other transition lines will most likely overlap[5].

Another issue to investigate is the ability of the method to deal with identifying elements with K_α values close to the noise level. For this Medipix2 detector

[5]For instance, in order of increasing energies it goes: Ni(K_α), Cu(K_α), Ni(K_β), Zn(K_β), Cu(K_β), Zn(K_β)

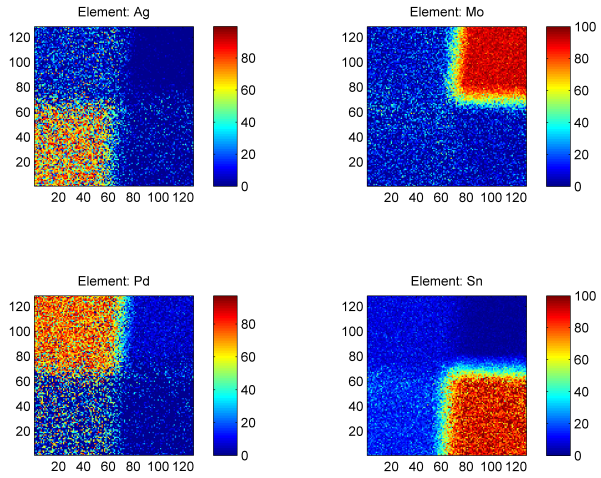


Fig. 7.2: Images of the percentage strengths (As indicated by colour) for each detected element contributes to each 2x2 pixel group for the compound sample consisting of Ag, Mo, Pd and Sn

the noise became significant above THL values of 450[6] and overwhelming above THL values of 460[7].

For the remaining samples, a smaller pinhole size of 1mm was used but the voltage of the X-ray beam and current remained the same (100kVp and 200μA respectively).

7.2.1 Pd, Ag, Mo, Sn

With atomic numbers of 46, 47, 42 and 50, the sample of Palladium, Silver, Molybdenum and Tin was a good first test on how the analysis could cope with multiple elements. As an additional challenge, the Palladium and Silver have energy lines close to each other making it difficult to identify between them. The noise level of this Medipix2 detector was of no concern since the lowest K_α transition occurs at approximately 17 keV.

For this test, Medipix2 was set to perform a threshold scan from THL values of 280 to 454, going up in steps of 2 units[8].

Figure 7.2 shows the results for this test. The analysis stage using all 256 $\times$ 256 pixels, correctly detected and identified the four elements of the sample and figure 7.2 shows the percentage contributions each of these elements made in every 2 $\times$ 2 grouped pixel.

[6] $\approx 7keV$ using a mean energy calibration
[7] $\approx 5keV$ using a mean energy calibration
[8] Or using a mean energy calibration, approximated from 6 to 33keV in steps of 0.3keV

Here the results for Molybdenum and Tin are very strong, with the distinct red corners for each indicating that these pixels could have their spectra described mostly (if not entirely) by the energy lines from the one element. The light blue speckling in the other corners are not caused by the elements being found there, rather they are caused by the least-squares fitting function interpreting bumps in the spectra as being actually important and by including a small amount of the wrong element, the fitted curve in that pixel describes the raw data better. Improving the number of events reaching the detector would result in smoother spectra in all the pixels and these anomalies would not be as evident, but improving the count rate generally results in another aspect of the performance deteriorating (like scan time increasing or spatial resolution decreasing).

The regions of Palladium and Silver are not as definitive as the Molybdenum or Tin regions but they are both distinctive enough to say that the analysis can identify between them, showing that Medipix2 and this analytical method can make it possible to discriminate between elements that are atomic neighbours and have energy lines that are very similar to each other. Again, if the count rate could be improved, the two regions would come to resemble the strong red regions of Molybdenum and Tin.

7.2.2 Sn, Ag, Ni, Cu

This sample of Tin, Silver, Nickel and Copper is more of a challenge than the previous sample. The energy peaks are now spread over a much larger range and also, the Nickel and Copper regions will be difficult because they are one Z value apart and they have K_α peaks very close to the noise level of this Medipix2 detector[9]

Although not intended, this sample had an added difficulty. The proper scan data of this sample was unusable and too late to redo. However, the initial quick scan data still existed and was tested instead. This started at a THL value of 321 and finished at a value of 451 going up in THL steps of 10, or using a mean energy calibration, it went from about 7 to 27 keV in steps of about 1.6 keV. This is not ideal as the large steps of energy will make it very difficult to identify the exact energy positions of the peaks properly.

In spite of this coarse THL step size, reasonable results were obtained as shown in figure 7.3 on the following page. Like the previous sample, the analysis is able to identify the regions of Silver and Tin with the same small fitting mistakes in each other's region.

Interestingly, neither Tin or Silver made a noticeable contribution to the Nickel and Copper regions[10]. This is obviously correct but was not seen in the

[9]Recall, this Medipix2 detector start to become noisy around THL values of 460 and upwards or 6.9759 keV and below. Nickel and Copper have energy peaks at 7.5 and 8.0 keV respectively

[10]A small region of contribution actually can be seen between the Ni and Cu corners, this is a result of attempting to fit essentially noise as this small section is empty space. Comparing the counts in pixels in this region to the average number of counts from any pixels that actually contain an element makes it obvious that this is empty space. It is just difficult to display all

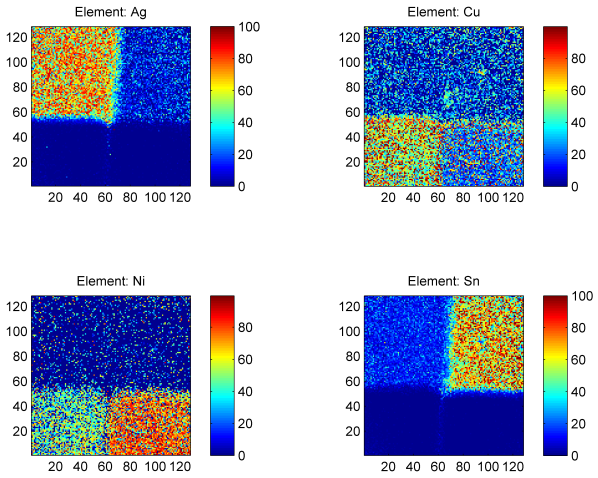


Fig. 7.3: Images of the percentage strengths (As indicated by colour) for each detected element contributes to each 2x2 pixel group for the compound sample consisting of Sn, Ag, Ni and Cu

sample of Pd, Ag, Mo and Sn. Possibly the gap between energy peaks from the Tin and Silver to the peaks of Nickel and Copper was so great that the least-squares fitting function had no reason to include any Tin or Silver to the bottom two corner regions. Why the analysis shows that Nickel and Copper contribute to the Tin and Silver regions when the situation is the same but in reverse is not clear.

The results of Nickel and Copper are not as conclusive. From examining the Palladium and Silver in the previous sample, it is possible to identify between elements that are 1 Z value apart with the two regions returning as distinctly different. The Nickel and Copper in this sample were much more blended together. The Copper region[11], in particular, is shown to be almost a 60/40 blend of Copper and Nickel. The analysis could at least identify the regions as Nickel or Copper. It is important to remember that these results are from a coarse scan and also to be able to capture the K_α lines of Nickel and Copper it is necessary to work close to the noise level of this Medipix detector. The noise level is most likely the cause of the poor distinction between the two elements. If it were a symptom of the coarse scan, then the analysis would have more likely been unable to identify any distinction and would have returned both corners as one element.

this information on the one image.
[10]

[11] bottom left corner

7.3 Printed Circuit Board: A Real World Sample

The final sample to be examined was an unused printed circuit board (PCB). Unlike the previous samples, no prior information (except what can visually be seen) was known, meaning that only common sense could determine the success of the analysis.

This sample is a further step up in complications from those previously tested. The layout of elements in this sample are not neatly separated into the four corners of the image, so the analysis has to deal with different elements touching each other. It also has to handle the possibility of different elements being layered on top of each other.

The sample was examined from THL values of 310 to 458 in steps of 4 units, with an exposure time of 400 seconds per second the scan took about 4.5 hours[12].

Three distinct elements were found (As, Br, Cu) to be the only major contributors to the energy spectrum of all 256x256 pixels. Moving to smaller pixel groups of 2x2 pixels, the percentage contributions each element makes to that pixel group's spectra are shown in figures 7.4-7.6.

Copper in the sample was to be expected, and as shown in figure 7.4 on the next page, it is found strongly in the wire sections and nowhere else. How well the setup and analysis copes with elements which have no gap between them can be seen in the Copper results. Nothing unexpected happens at the edges of the wires and the halo of light blue around the wire edges shows the resolution. Ideally it should be a sharp transition from mostly Copper contributing to no Copper contributing, but a rough inspection shows a one or two pixel transition region[13].

Figure 7.5 on the following page, shows that the non-wire regions are almost exclusively Bromine and the wire regions are about 30%.

FR-4 is a very common laminate (base) material of PCBs, this is a flame resistant glass woven epoxy with Bromine one of the main ingredients.

This allows a degree of perspective when analysing the results shown in figure 7.5 on the next page. As the laminate is the base material of this PCB, without the Copper wiring the results would return nearly 100% Br like any of the other single element samples tested (see figure 7.1 on page 55). Including the Copper regions would mean that the Copper signals received would be strongest as it is found on top and therefore less attenuated. It would not be a pure signal though since the Bromine is still present underneath. This explains why the wire regions return about 30% Bromine as the contribution of Bromine to these pixels is much less than the Copper since its signal is weaker due to needing to travel through the Copper wires and the regions without wires return almost exclusively Bromine. Only the via regions return very little Bromine and mostly Copper because these regions have the Copper going right through the

[12]This would be considered to long for online analysis, but was done so to see if results could be achieved with a real sample. Later work will be done to try and improve the exposure time.

[13]Most likely due to scattering

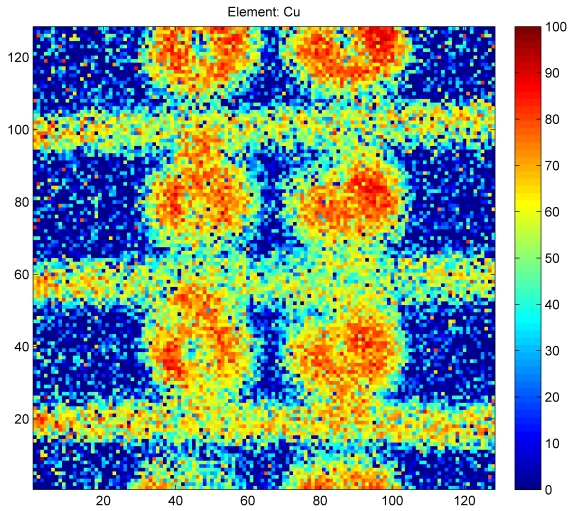


Fig. 7.4: Percentage of Cu calculated to be contributing to each pixels spectrum (Pixel Group Size: 2x2)

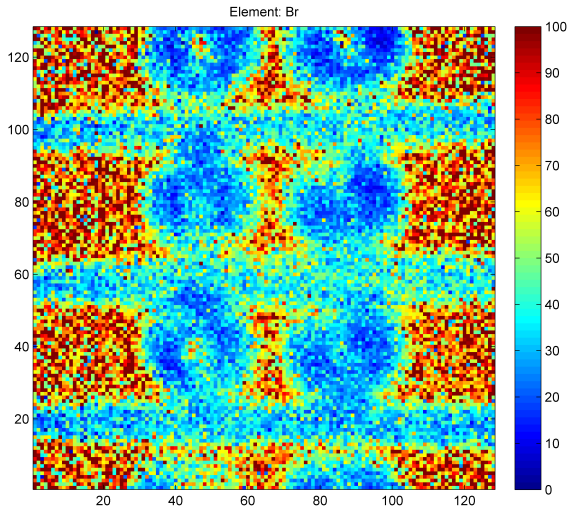


Fig. 7.5: Percentage of Br calculated to be contributing to each pixels spectrum (Pixel Group Size: 2x2)

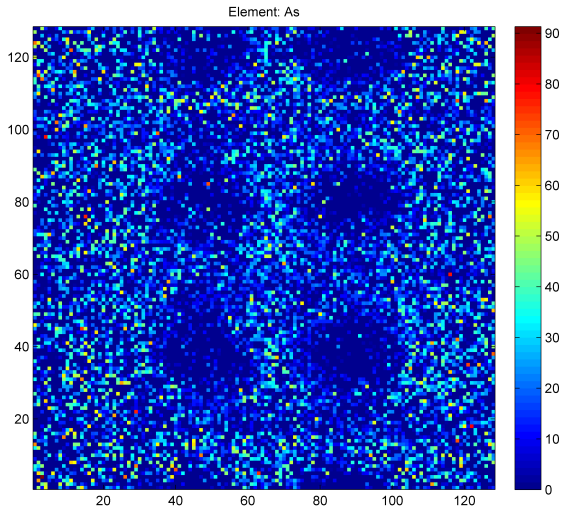


Fig. 7.6: Percentage of As calculated to be contributing to each pixels spectrum (Pixel Group Size: 2x2)

PCB and no Bromine to contaminate the spectra of these pixels. The Bromine found inside the via regions is not an indication that Bromine exists here, rather it is the best effort of the analysis to fit the noise in these pixels (as these regions contain no sample).

This sample then provides an excellent indication of how well this setup and analysis can cope with elements layered on top of each other. It will need further testing using different elements and different thickness to understand its limits, but from what can be seen from this PCB, the results are very promising so far.

The final detected element was Arsenic and the contributions of this element can be seen in figure 7.6. For reference, Arsenic is a commonly used for cleaning purposes after the solder masking process. Overwhelmingly the spectra in each pixel is dominated by either Copper or Bromine, with only a few pixels returning very strong contributions of Arsenic. That the analysis was able to detect the minute traces of Arsenic was an unexpected but extremely positive result.

The results from this PCB show great potential of this detector/analysis setup for mineral analysis. It is able to cope with samples where the arrangement of elements are not neatly defined as was the case with all the samples tested so far and more importantly it could handle where the sample had overlapping elements with both elements being able to be identified. Understandably the lower layer element (Br) did not come through as intense as the top layered element (Cu) and this indicates that there is a limit to the thickness of layers before it becomes impossible to detect any more layers, the fact that in figure 7.4 it was not able to see the alternate Copper wires on the other side

of the PCB is also an indication of this limit. Finally, the detection of Arsenic in the sample showed that it is possible to detect trace elements even when the spectra are dominated by other elements. This is a very positive result in regards to the intended real life applications of this device, since it is hopeful that after the refinement process there would only be trace amounts of desired material remaining in the tailings.

8 Conclusions

8.1 Summary of Results

The analysis was used on a number of different samples with varying degrees of complications in choice of elements and positioning of the elements. These results were obtained using pixel groups of 2×2 pixels which equals pixel sizes of $110 \times 110 \mu m$.

With the exception of requiring an additional shift of energy, all the samples tested returned positive results, with varying degrees of success. The analysis was able to separate between different elements in a sample even when these elements are atomic neighbours (see Pd and Ag in subsection 7.2.1 on page 56). It did, however, have greater difficulty separating atomic neighbours when these elements had their K_α energy lines close to the noise level of the detector (see Ni and Cu in subsection 7.2.2 on page 57).

The final sample tested was a small section from a printed circuit board (section 7.3 on page 59). The results of this analysis showed that it was possible (with no modification of the analysis method) to detect different elements in the sample when they are not neatly arranged and even when these elements are layered on top of each other. The detection of trace amounts of Arsenic in the sample shows that it is possible to detect small amounts of an element even when they do not contribute much to the total sample, this is a very useful result considering the real world applications envisioned for this device.

The results here did also indicate that a maximum sample thickness may exist where it becomes impossible to detect the bottom elements of the layers as the underside Copper was not identified.

8.2 Comments on Future Work

The results obtained thus far, show great promise and certainly warrant taking this project to the next phase of development.

8.2.1 Areas of Further Investigation

Further investigation is required to examine the results demonstrated by the PCB. It is important to see how the method can cope with a real life example, and so the method should be retested[1] using samples of known configuration to gain a better idea of the limits to this method. Of most importance would be testing the limit of thickness that exists, also testing how well the method copes with layered elements that are one Z value apart will be interesting and finally more investigation on how well this method deals with trace amounts is required.

8.2.2 Improvements to the Setup

For the next stage, a new geometry layout should be used. As already discussed briefly in the setup section (section 4.1 on page 24), the detector needs to be moved so that it views the sample as close to straight as possible[2]. This is to reduce any problems of distortion when interpreting the image results.

Finally, of all the samples tested, each one would have improved results if the detected spectra in each pixel was smoother and to do this, the number of counts in each pixel needs to be increased. Normally, this would mean one of two things; increase the pin-hole diameter (but in doing so reduce the resolution), or increase the time of each scan (which is counter-productive to the desired final product). However, an alternative solution could be found by using the very cause of the low flux rate.

Few rays reach the detector because the fluorescent x-rays generated by the sample are ejected in all directions, not just in the direction of the detector. By using multiple detectors placed in the same position except rotated around the beam axis, the results from each detector could be combined together (after compensating for the rotations of the image) creating a much larger count rate in the combined image.

Provided it is possible to compensate for the geometry (and although it could be difficult, it shouldn't be impossible), then this improved setup will improve all three of the R's outlined in the introduction. The results will be more reliable because the increase in count rate will make each spectra cleaner and thus easier to work with. The resolution can be improved if working with grouped pixels is undesired, theoretically by having four detectors, the spectra in each individual pixel should be equivalent to the spectra in each 2x2 pixel group currently used. Finally, it would create a much more realistic on-line detector, as adding more detectors to capture more x-rays would allow the overall scan time to reduce without degrading the final results.

[1] The results of the PCB were positive, however lacking the knowledge of what elements are to be found in that exact sample meant that it is difficult to say with a high degree of confidence how positive the results were. In hindsight it would have proven more effective to test a real world sample, like the PCB, but test a sample with known composition before later moving onto more unknown samples.

[2] It can't be positioned so that it is directly above the sample because it would block the X-ray beam.

8.3 Final Word

For this method to produce a viable detector to be used on-line, certain criteria were mentioned in the introduction:

- Reliable: On the large scale (using all 256x256 pixels combined), this method has shown itself to accurately and consistently return the material composition of all the samples tested. Working with much smaller pixel groupings (2x2 pixels), this method still returned reliable results on all samples tested, however the problems of fitting a noisy spectrum did cause many pixels to return incorrect contributions of elements found elsewhere in the sample. Most of these incorrect contributions were minor, but the problem was much greater when trying to distinguish between elements that have close atomic numbers. Improvements to the overall design mentioned earlier in this chapter, would greatly improve this situation by increasing the effective number of counts in each pixel and thus creating a much smoother spectrum in each pixel.

- Resolution: As mentioned above, this method was able to get accurate results using 2x2 pixel groups or an effective pixel size of $110\mu m$ x $110\mu m$ and the only problem against using individual pixels was to simplify the analysis code regarding how it deals with dead pixels, for certain samples accurate results were most likely achievable using no pixel grouping. Again, the improved design mentioned would make it much easier to conduct the analysis with an effective pixel size of $55\mu m$ x $55\mu m$. More importantly as shown by others already [6], the resolution of this pinhole design is limited by the pinhole diameter, but using an improved design would allow for smaller pinhole sizes achieving the same if not better results to those demonstrated in this work.

- Realistic: The issue of how long each scan needed to be was the most important. Very little doubt was held that the method could not achieve reliable results at a reasonable resolution, but it came down to a question of how long it would take. If it took days to run, then this method would not prove a realistic "on-line" detector. The samples tested, were done using scan times of at most hours. Although not ideal, it is quick enough that a couple of methods are possible to improve this situation; firstly, the improved design mentioned earlier will reduce scan times as more fluorescent particles are being detected by the extra detectors (thus achieving the desired smooth spectra in each pixel quicker). A second method of improving the scan times depends on what the user is desiring to detect. Currently, all the samples were tested under the rough mindset of "I don't know what is in the sample and I want to test for everything", this means that total scan must cover a large range of energies using a small energy step size for the entire scan. But if, for instance, they are only interested in the amount of Gold that is present in the sample, then the user could specify to run a scan that focuses only on specific energy

transitions of Gold by scanning energy windows around those transition line energies, meaning that few energy levels need to be tested and so a quicker total scan time.

Overall, the results show that it is worthwhile taking the project to the next stage and creating a proper system. By being more careful about shielding, positioning and trying to improve the count rate by using a number of multiple detectors simultaneously, it should be possible to create much cleaner spectra. This will improve the spectra in the pixels and allow the analysis to work on more applicable samples using common compositions of materials found in the tailings of mines.

8.4 Acknowledgement

This work was carried out in the framework of the CERN Medipix Collaboration.

References

[1] J. Uher, G. Harvey, and J. Jakubek. X-ray fluorescence imaging with the medipix2 single-photon counting detector. In *2010 IEEE Nuclear Science Symposium Conference Record (NSS/MIC)*, pages 1067–1073, 2010.

[2] J. Uher, G. Harvey, and J. Jakubek. X-ray fluorescence imaging with the medipix2 single-photon counting detector. *IEEE Transactions on Nuclear Science*, 59(1):54–61, 2012.

[3] T. Holy, J. Jakubek, S. Pospisil, J. Uher, D. Vavrik, and Z. Vykydal. Data acquisition and processing software package for medipix2. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 563(1):254–258, July 2006.

[4] Jan Jakubek. Data processing and image reconstruction methods for pixel detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 576(1):223–234, June 2007.

[5] J Jakbek. Semiconductor pixel detectors and their applications in life sciences. *Journal of Instrumentation*, 4(03):P03013–P03013, March 2009.

[6] V. Tichy, T. Holy, J. Jakubek, V. Linhart, S. Pospisil, and Z. Vykydal. X-ray fluorescence imaging with pixel detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 591(1):67–70, June 2008.

[7] Michal Platkevic, Viktor Bocarov, Jan Jakubek, Stanislav Pospisil, Vladimir Tichy, and Zdenek Vykydal. Signal processor controlled USB2.0 interface for medipix2 detector. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 591(1):245–247, June 2008.

[8] M. Chmeissani and B. Mikulec. Performance limits of a single photon counting pixel system. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 460(1):81–90, March 2001.

[9] Brje Norlin, Christer Frjdh, Hans-Erik Nilsson, Heinz Graafsma, Vedran Vonk, and Cyril Ponchut. Characterisation of the charge sharing in pixellated si detectors with single-photon processing readout. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 563(1):133–136, July 2006.

[10] K. Mathieson, M. S. Passmore, P. Seller, M. L. Prydderch, V. O'Shea, R. L. Bates, K. M. Smith, and M. Rahman. Charge sharing in silicon pixel detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 487(1-2):113–122, July 2002.

[11] J. Jakubek, J. Dammer, T. Holy, M. Jakubek, S. Pospisil, V. Tichy, J. Uher, and D. Vavrik. Spectrometric properties of TimePix pixel detector for x-ray color and phase sensitive radiography. *2007 IEEE Nuclear Science Symposium Conference Record*, page 4, November 2007.

[12] X. Llopart, R. Ballabriga, M. Campbell, L. Tlustos, and W. Wong. Timepix, a 65k programmable pixel readout chip for arrival time, energy and/or photon counting measurements. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 581(1-2):485–494, October 2007.

[13] Jrgen Giersch, Andreas Weidemann, and Gisela Anton. ROSIan object-oriented and parallel-computing monte carlo simulation for x-ray imaging. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 509(13):151–156, August 2003.

[14] Alexander Korn, Markus Firsching, Gisela Anton, Martin Hoheisel, and Thilo Michel. Investigation of charge carrier transport and charge sharing in x-ray semiconductor pixel detectors such as medipix2. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 576(1):239–242, June 2007.

[15] Markus Firsching, Patrick Takoukam Talla, Thilo Michel, and Gisela Anton. Material resolving x-ray imaging using spectrum reconstruction with medipix2. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 591(1):19–23, June 2008.

[16] Patrick Takoukam Talla, Thilo Michel, Markus Firsching, Jrgen Durst, and Gisela Anton. Exploiting the MEDIPIX2 detector for the reconstruction of x-ray spectra. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 607(1):103–106, August 2009.

[17] Thilo Michel, Patrick Takoukam Talla, Markus Firsching, Jrgen Durst, Michael Bhnel, and Gisela Anton. Reconstruction of x-ray spectra with the energy sensitive photon counting detector medipix2. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 598(2):510–514, January 2009.

[18] Cyril Ponchut. Correction of the charge sharing in photon-counting pixel detector data. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 591(1):311–313, June 2008.

[19] Josef Uher, Greg Roach, and James Tickner. Peak fitting and identification software library for high resolution gamma-ray spectra. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 619(1-3):457–459, July 2010.

[20] M. Fiederle, D. Greiffenberg, J. Idrraga, J. Jakubek, V. Krl, C. Lebel, C. Leroy, G. Lord, S. Pospsil, V. Sochor, and M. Suk. Energy calibration measurements of MediPix2. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 591(1):75–79, June 2008.

APPENDIX

A
HYPERMET

The HYPERMET function [19] is used to describe the XRF spectrum that is detected by Medipix. It is a combination of different functions which each describe a certain aspect of the physics that goes on behind the scenes to create the final spectrum observed[1].

The HYPERMET function used to calculate the number of detected counts, $y(E)$, in a bin with median energy, E, is given in equation (A.1) and in figure A.1 on the next page the different components that make up HYPERMET are plotted separately.

$$y(E) = \sum_{i=1}^{N} \Gamma_i \left[(1-A)\cdot G(E-P_i) + A\cdot T(E-P_i) + S\cdot B(E-P_i)\right] + a + b\cdot E + c\cdot E^2 \quad \text{(A.1)}$$

Where:

- $G(E) = \exp\left(-\frac{E^2}{\delta^2}\right)$
- $T(E) = \frac{1}{2}\exp\left(\frac{E}{\beta}\right)\cdot \mathrm{erfc}\left(\frac{\mathrm{E}}{\delta} + \frac{\delta}{2\beta}\right)$
- $B(E) = \frac{1}{2}\mathrm{erfc}\left(\frac{E}{\delta}\right)$
- N = the number of peaks in the fitted region of interest (ROI)
- Γ_i = the amplitude of the i^{th} peak in the ROI
- A = the ratio between the number of counts remaining in the Gaussian peak and those suffering from incomplete charge collection in the detector and contributing to the exponential tail
- P_i = the position of the i^{th} peak in the ROI

[1] For example, a Gaussian is used to describe the smearing of the transition line due to electronic noise in the detector

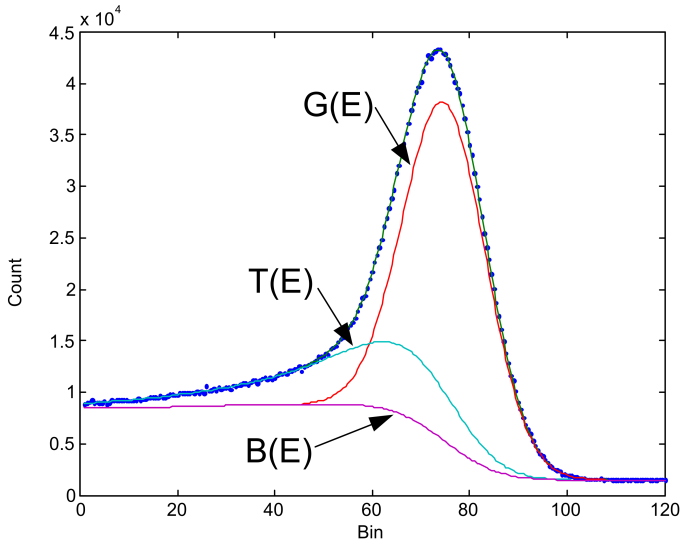


Fig. A.1: The three main components of HYPERMET[2]

- S = the amplitude of the step background under the peak caused by electron-pair production and Compton scattering
- a, b, c = the coefficients of the polynomial background
- $G(E)$ = Gaussian peak with full-width at half-maximum (FWHM) of 1.665δ. Accounts for the transition line smearing.
- $T(E)$ = represents events with incomplete charge collection in the detector volume. It has an exponential slope parameter of β
- $B(E)$ = represents the step background using the complementary error function: $\text{erfc}\,(E) = \frac{2}{\sqrt{\pi}} \int_E^{\infty} \exp\left(-t^2\right) dt$

[2]The quadratic background function is not shown

B

Spatial Resolution of the System

The spatial resolution of the system is a combination of the spatial resolution of the Medipix2 chip and the spatial resolution of the pinhole aperture, however there is no need to be concerned with the contribution of the resolution from the Medipix2 chip since the overall resolution is limited by diameter of the pinhole aperture, [6].

B.1 Determination of the Spatial Resolution

The definition of the spatial resolution used in this analysis is the FWHM of the line spread function (LSF). It is however easier to work with an edge response function (ERF)[1], as experimentally it is easier to measure than the LSF. The ERF is expected to take the shape of a complimentary error function:

$$y = 0.5 \times Amplitude \times erfc\left(\frac{middlePixel - x}{\sigma\sqrt{2}}\right) + Offset \qquad \text{B.1}$$

Where:

- x = pixel position value
- y = pixel signal (i.e. number of counts in this pixel)
- $Amplitude$ = the difference between the high and low response regions
- $Offset$ = the average signal value before the edge
- σ = the sigma of the Gaussian error function
- $middlePixel$ = the pixel that contains the middle of the transition region (i.e. the edge position)

[1] Although the Edge Response Function used in this work is vaguely related to the standard mathematics Error Function, the two are not the same and as a distinction between the overlap of abbreviating both terms, the Edge Response Function will be abbreviated to ERF whereas the Error Function will be abbreviated to erf

The LSF is defined to be the first derivative of the ERF, which means that the LSF will be described by the Gaussian curve:

$$y' = \frac{Amplitude \times \sqrt{2}}{2\sigma\sqrt{\pi}} \times \exp\left(-\frac{(middlePixel - x)^2}{2\sigma^2}\right) \quad \text{B.2}$$

And finally the FWHM of this Gaussian curve is approximately:

$$FWHM \approx 2\sqrt{2\ln(2\sigma)} \quad \text{B.3}$$

So, by measuring the response of Medipix to an edge profile and then fitting the resulting ERF with an error function, then the σ parameter used to fit the ERF can be used to determine the FWHM of the LSF which finally gives a value to describe the spatial resolution of the Pinhole/Medipix2 system.

B.2 Sub-Pixel Sampling

Ideally, the spatial resolution of the system would be high enough that the edge profile observed would be sufficiently sharp enough that there would be only a couple of pixels that fall in the transition region between low and high counts. This causes problems when trying to quantify the spatial resolution, as the region of most importance in determining the fit parameters is the region with fewest data points to test against.

An effective solution to this problem is to take the edge image at an angle, this means that the transition region no longer falls on the same few rows of pixels but now will occur over a much wider range of pixels. Although each individual edge profile still has the problem of few pixels in the transition region, by shifting all of the profiles such that the edge is now straight and then combining all of the profiles together, it results in a single profile with sub-pixel sampling[2]and many more data points in the transition region as seen in figure B.1 on the following page.

B.3 Spatial Resolution using the Large Pinhole

B.3.1 Analysis

A Sn foil sample with angled edge was prepared and a 400 second XRF image was taken with the threshold setting of Medipix set to THL = 410. The resulting image[3]can be seen in figure B.2 on the next page.

In figure B.3 on page 76, an example single column profile is shown. Clearly, and unfortunately, the spatial resolution seems to be low enough that many pixels fall inside the transition region. This removes some of the need to continue

[2]That is, this profile now has data points that fall between the integer pixel values

[3]This is not the pure raw image, this image has had the bad pixels masked out and has also been rotated 180 degrees to simplify the further analysis

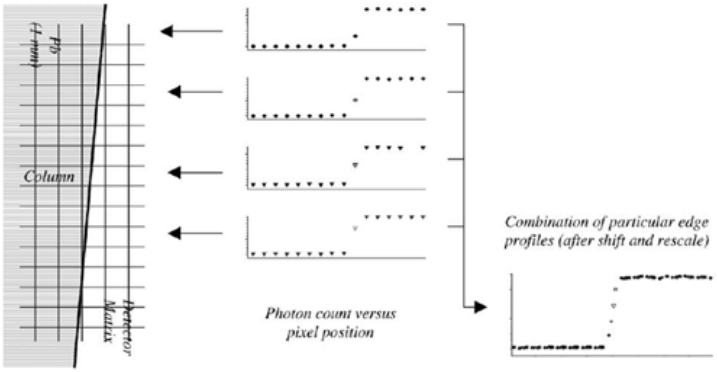


Fig. B.1: Diagram outlining the Sub-Pixel sampling concept

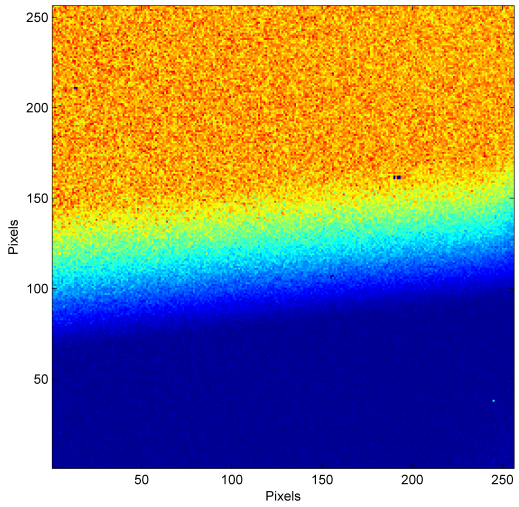


Fig. B.2: Image of Sn Edge

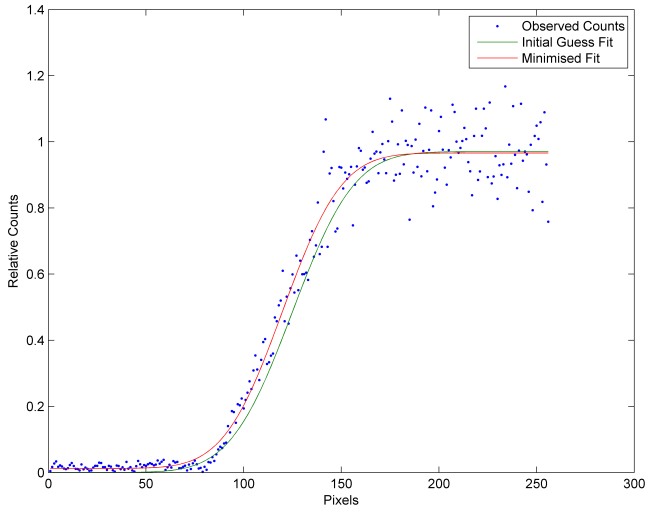


Fig. B.3: The ERF curve of the 128^{th} column shown as an example of the fitting process with both the initial guess and minimised parameter curves plotted

with the sub-pixel sampling process, but since the sample was prepared on the angle and a sub-pixel profile will produce a more accurate result, there is no reason not to continue.

It should also be pointed out that in this analysis, all the count values in each pixel were divided by the average counts value calculated using a section of the image that contains all Sn (i.e. a section from the top of figure B.2 on the previous page). This was done to simplify making the initial guess of the Amplitude parameter used in equation (B.1) on page 73, as now it should always be close to 1 and this should not change depending on the setup settings like exposure time.

Having fitted all 256 columns of figure B.2 on the previous page, the determined middlepixel parameter is compared to the column pixel it came from and a line of best fit is calculated that will be used to determine by how much each column profile should have its pixels shifted by. The results for this sample can be seen in figure B.4 on the following page. What is obvious from this figure is that there is a good linear relationship for most columns of the image except for the last few columns where the edge position seems to shoot up much higher than normal. A closer look at figure B.2 on the previous page shows that the edge does indeed seem to no longer continue in a straight line at the right side of the image, this is most likely simply due to using scissors to cut the Sn foil down the middle to create the edge.

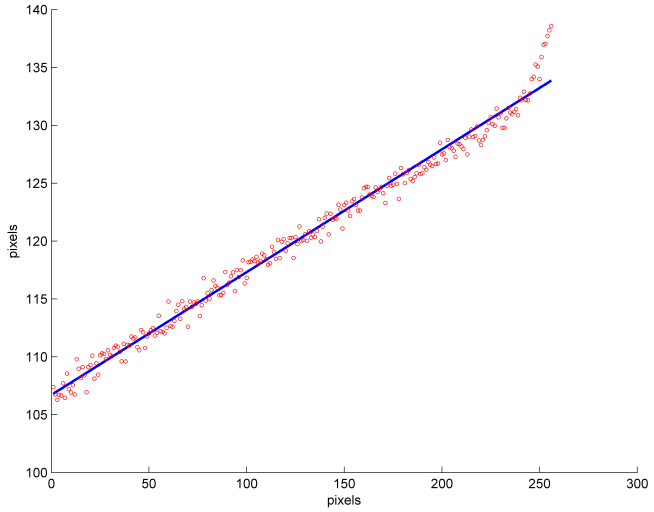


Fig. B.4: The calculated pixel positions of the edge center with the line of best fit that will be used to shift the Edge image

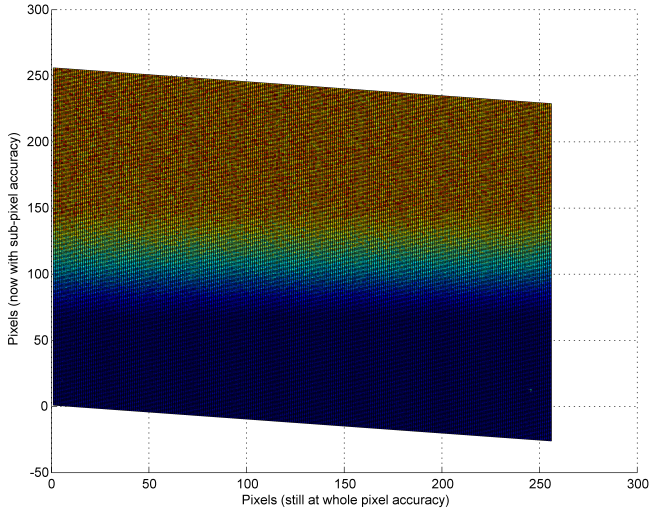


Fig. B.5: The Sn edge image after having each row shifted so that the middle of the edge should now line up horizontally

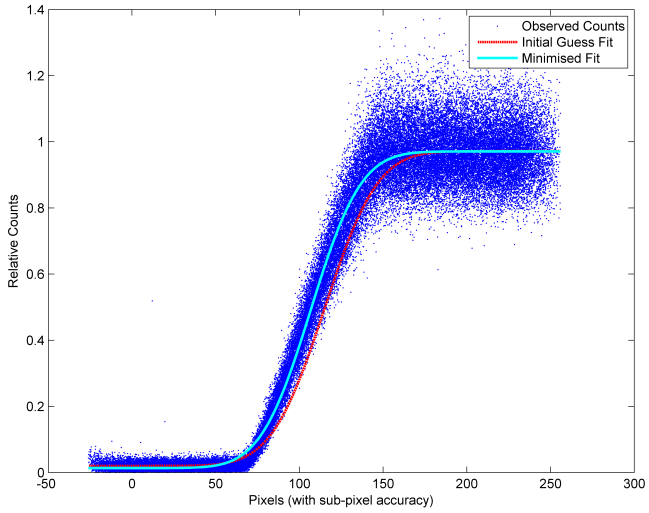


Fig. B.6: The final LSF curve after combining and sorting all 256 shifted LSF curve's together. Again like in figure B.3 on page 76 both the initial guess and minimised paramters LSF curves are also plotted.

B.3.2 Results

Using the fitted line of figure B.4, all of the columns can now be shifted (as shown in figure B.5) and then each edge profile can be combined into one profile (see figure B.6) to create the sub-pixel sampling desired.

After calculating the initial parameters[4], they are passed to `MINUIT` which produces a best fit value of σ=23.6594 pixels or[5] σ=0.7685mm. Using equation (B.3) on page 74, results in the FWHM equalling 1.8543mm.

[4] In the same manner as described earlier for each column profile
[5] As 1 pixel = 52μm

C
Sample Code

In this appendix, much of the Matlab code created is presented. Some created functions that are mentioned elsewhere in this thesis are not included because they are either too simple or conversely, too complicated[1]to be included.

[1]read this as meaning too long or else not elegant enough to include

C.1 Calling DataMaster

This function is to call the external Java program DataMaster for the purpose of creating the response matrix. The mathematics and implementation behind the response matrix can be found in chapter 3 on page 18.

The inputs for this function are:

- bins: This is the energy bins to simulate the responses over. It is a requirement of DataMaster that these bins start at 0.
- energyCutOff: Is the energy value where the response matrix is truncated, columns below this energy will remain as 0. In figure 3.1 on page 22 this is set at 1 and is clearly visible in the figure.
- stepCount: This is a variable informing DataMaster how many layers to divide the sensor layer into.

The parameters used in this function were experimentally determined to best fit sample data and may need to be changed over time due to ageing and changes of the detector.

The function returns the response matrix, this is square matrix equal in size to number of bins used. Each column of the matrix corresponds to the simulated spectra for that column's incident energy.

```
function [resp]=mkMpxResponse(bins,energyCutOff,stepCount)
mpxChr=dmaster.math.MPXChargeSharingFunction;

mpxChr.getParameter('THL').setDoubleValue(energyCutOff);
mpxChr.getParameter('stepCount').setDoubleValue(stepCount);
mpxChr.getParameter('chargeSigma_top').setDoubleValue(11.575);
mpxChr.getParameter('chargeSigma_bot').setDoubleValue(1.425);
mpxChr.getParameter('eSigma').setDoubleValue(0.591);
mpxChr.getParameter('depthSteps').setDoubleValue(30);

resp=zeros(numel(bins),numel(bins));

mpxChr.generateResponseMatrix(bins);

while(mpxChr.isResponseMatrixGenerated && ...
        mpxChr.isResponseMatrixGenerationCanceled)
    pause(0.1);
end

if(mpxChr.isResponseMatrixGenerationCanceled)
    resp=[];
else
    resp=mpxChr.getResponseMatrix';
```

```
    id=find(isnan(resp(:)));
    resp(id)=0;
end
```

C.2 Grouping Pixels

This function, as described in subsection 6.2.2 on page 38, takes as inputs:

- oldData: Is a $256 \times 256 \times n$ matrix, where n represents the number of different energy layers, with each layer containing the count values from all 256×256 pixels.
- mask: Is a 256×256 matrix of all the known pixels that need to be masked as they are either too noisy or dead. A good pixel has a value of 0 and a masked pixel has a value of 1.
- groupSize: Is an integer indicating the number of pixels to be placed into each group. A value of 32 for example would mean that the final product would be a matrix $8 \times 8 \times n$ where each of these pixels would contain 32×32 of the original pixels.

```
function [newData] = groupPixels(oldData,mask,groupSize)

newData = zeros(256/groupSize,256/groupSize,size(oldData,3));

if groupSize == 1
    disp('No need to group Pixels');
    newData = oldData;
else
    r = 1;
    for row = 1:groupSize:256
        c = 1;
        for col = 1:groupSize:256
            box = oldData(row:row+groupSize-1,col:col+groupSize-1,:);
            subMask = mask(row:row+groupSize-1,col:col+groupSize-1,:);
            m = nanmean(nanmean(box));
            id_bad = find(subMask == 1);
            if size(id_bad,1) > 0
                for i = 1:size(box,3)
                    tmbox = box(:,:,i);
                    tmbox(id_bad) = m(i);
                    box(:,:,i) = tmbox;
                end
            end
            newData(r,c,:) = squeeze(sum(sum(box)));
            c = c + 1;
        end
        r = r + 1;
    end
end
```

C.3 Determining the Fit Lines

This function, as described in subsection 6.3.2 on page 41, is how the Response Matrix is used to fit an observed spectrum. Rather than saving the fitted spectrum, the determined coefficients of best fit are stored as these represent the incident energies on the detector.

This function takes as inputs:

- Data: Is a $a \times a \times n$ matrix, where n represents the number of different energy layers and a represents the number of grouped pixels (see appendix C.2 on the previous page).
- ResponseMatrix: Is a square matrix[2] equal in size to the number of different energy layers of the Data matrix. It contains the simulated Medipix2 responses for a range of incident energies.

The function returns fitLines, an $a \times a \times n$ matrix which contains the set of linear coefficients that combine the different columns of the response matrix to best fit the measured spectra in each grouped pixel of the Data matrix. Matrix multiplication of the ResponseMatrix with 1 grouped pixel's coefficients results in the fitted spectra of that group.

```
function [fitLines] = determineFitLines(Data,ResponseMatrix)

fitLines = zeros(size(Data,1),size(Data,2),size(ResponseMatrix.Int,2));

for row = 1:size(Data,1)
    for col = 1:size(Data,2)
        Int_Spectra = squeeze(Data(row,col,:));
        fitLines(row,col,:) = lsqnonneg(ResponseMatrix.Int,Int_Spectra);
    end
end
```

[2]It actually does not need to be square, only that the number of rows equals the number of layers in the Data matrix. In practise it was always kept as a square matrix for simplicity.

D

Determining the Elements in a Sample: An Example

The following is an example of how the chosen 'Rules' for sorting out which element is the cause of each determined incident energy line. The example is based on real results from the compound sample of Pd, Ag, Mo, Sn using a group size of 128x128 it however was altered slightly to demonstrate the procedure more clearly

D.1 Introductory Explanations

	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998973																	1	1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1					1							
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

Fig. D.1: Initial Matrix of Results

Before the step by step example begins it is important to go over what all the rows and columns mean and where each of the values come from.

In figure D.1, a number of different saved matrices are combined. Along the top, in the rows labeled Z, Transition and Energy, are the entries from the NIST database.

The Z row is straight forward and corresponds to the Z value for each of the transitions being tested.

The Transition row contain the different types of transitions being tested, they are only stored as integers from 1 up and each transition type is assigned a number depending on how much it is expected to contribute to the spectrum. At

the moment the order[1] is K_{α_1}, K_{α_2}, K_{β_1}, K_{β_2}. The order is important because it dictates the order of transitions we should detect, i.e. if in the sample some of element A is present, then the contribution from the K_{α_1} transition should be strongest, the K_{α_2} transition should be next strongest and so on. If there is a choice between two or more transitions, then the highest non-assigned transition will be picked[2]. By changing the order the transitions are saved in the database file, the order of expected transitions is changed in the analysis without any need of changing the Matlab code (as far as the Matlab code is concerned nothing has changed).

The final row along the top is the transition energies listed by NIST and it is these values plus a set energy window around the values that pre-determined which of the detected transition lines could belong to each of the accepted transitions. Although these values appear to be rounded to the nearest whole number, it is actually stored to as many decimal places NIST provide. So that the steps of this example will fit on a single page, the columns were reduced in size and only the nearest integer values are shown.

The first two columns are from the calculated energy lines that, when combined, explain the detected spectrum. These calculated lines occur at different energies and with different strengths and to start with, all the calculated lines are ranked in order of intensity. This is what the % Height column lists, each energy line is divided by intensity value of the strongest line. So it can be seen that in the example shown in figure D.1 on the previous page, there are two really strong lines, a couple more reasonably intense and then many small energies compared to the main energy line.

The second column also labelled as Energy, are the calculated energy values that each determined energy line occurs at.

The core of the diagram consists of all the possible transitions that each determined energy line could be. The step before this, had Matlab go through the NIST database and for each transition of each element it tested whether any of the calculated transitions had energy values close to any of the NIST values (it did this using a set energy window). All the cells with a 1 in them mean that the determined transition energy could belong to the above listed element's transition line.

D.2 Example: Step-by-Step

For the first line, since this is the first transition tested, then the rules require that this line comes from an elements main transition (labelled 1 in the database, which corresponds to a K_{α_1} transition). In the rare case that the strongest determined energy line doesn't fit within a window around any elements main transition line, then the energy window is slightly increased and every transition

[1] To change the order or even the number of transitions to be checking for, the library containing these values need only be changed

[2] It hopefully will become clear as the example is explained step by step

is tested again until finally an element's main transition line is close to the most intense determined energy line.

In figure D.1 on page 84, it can be seen that the only transition 1 that is possible belongs to Z = 46 and so this determined energy line is set and is listed as belonging to Z = 46.

Now that the first transition has been assigned[3], what the code does is blanks out any of the the other possible transitions listed (i.e. any other 1's along the 1st row are blanked out, which visually is represented by blacking out the cell but in reality Matlab just turns the other 1's into 0's). Also, now that the 1st transition for Z = 46 has been taken (and it obviously can't be used again), all other calculated energy lines that had a possibility of being the 1st transition of Z = 46 are blanked out (i.e. any other values in the column of what was just assigned are blanked out).

These changes are shown in figure D.2. Since there were no other transition lines that could possibly fit the 1st line, nothing needed to be blanked along that row, but the second row's transition line had a possibility of being the 1st transition of Z = 46 so since this has been taken, it was blanked out.

	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998973																		1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1					1							
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

Fig. D.2: Line 1

Moving onto the next row, now that a transition line has been assigned, it opens up more possible outcomes for subsequent transitions, the next transition line could be caused by the 1st transition of a new element or it could be a lower transition of one already present. The way the rules are set out, preference is giving to assigning lower energy lines to already determined Z values, but only if it is possibly caused by the next ranked transition.

In other words, if it is a choice between the 1st transition of a new element or the 2nd transition[4](if the 1st has already been assigned), then the choice goes to the existing element.

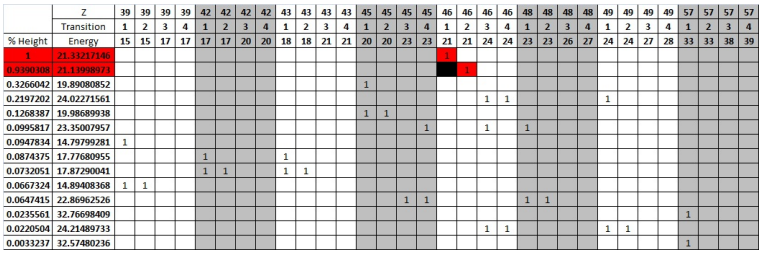
If it is a choice between a 1st transition of a new element or the 3rd transition[5]of an existing element (which hasn't had the 2nd transition assigned), then the choice goes to the new element.

[3]And this happens after every transition has been assigned

[4]Or equally valid, it could be the 3rd transition so long as the 1st and 2nd transitions have been assigned or it could be the 4th transition if the 1st, 2nd and 3rd transitions are already assigned

[5]Or the choice could be a 4th transition without the 3rd transition being assigned and so on

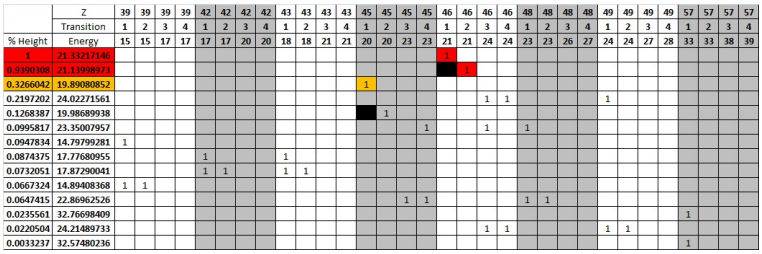
In this example, the only possibility not taken for the 2nd transition line is that it belong to the 2nd transition of Z = 46. So all other values in that row and column are blanked out and the result is shown in figure D.3.



	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998971																		1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1					1							
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

Fig. D.3: Line 2

For the next step, there is only one choice that it could be (1st transition of Z = 45) and in this case it is assigned to Z = 45 and this can be seen in figure D.4. This is not a certainty however, if the only choice for a determined energy line belonged to a 2nd, 3rd or 4th transition for an element with out a 1st transition already assigned, then this transition line is ignored under the assumption that a K_β transition should not be stronger than a K_α transition.



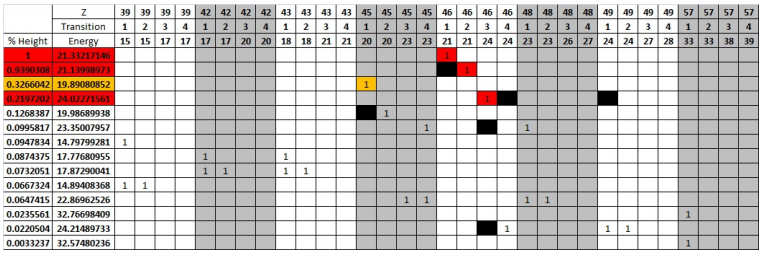
	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998971																		1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1					1							
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

Fig. D.4: Line 3

The next step sees a choice between transitions 3 or 4 of Z = 46 and transition 1 of Z = 49. Because transitions 1 and 2 of Z = 46 have been assigned, then this is an example of the rule already described, so transition 3 of Z = 46 is chosen over transition 1 of Z = 49. These choices are shown in figure D.5 on the following page.

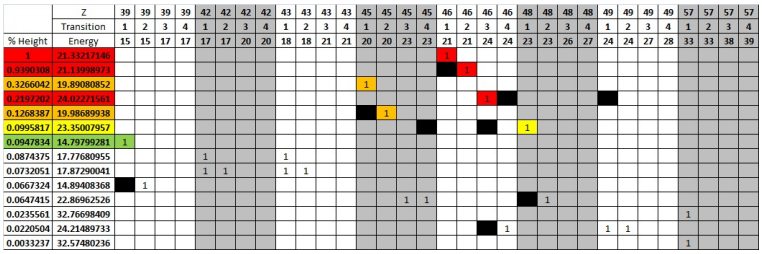
The choice for line 5 introduces nothing new and is assigned transition 2 of Z = 45. Line 6 is given to the 1st transition of Z = 48 over the 4th transition of Z = 45 because the 3rd transition for Z = 45 has not been assigned yet. The choice for line 7 is again straight forward with only one option and so it is assigned as the 1st transition for Z = 39. These three steps are shown in figure D.6 on the next page

The next step on from figure D.6 on the following page is not as simple since it consists of a tie. No already picked element has a transition match for this determined energy but the next option is to pick a 1st transition from a new



	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998973																	1	1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1				1								
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

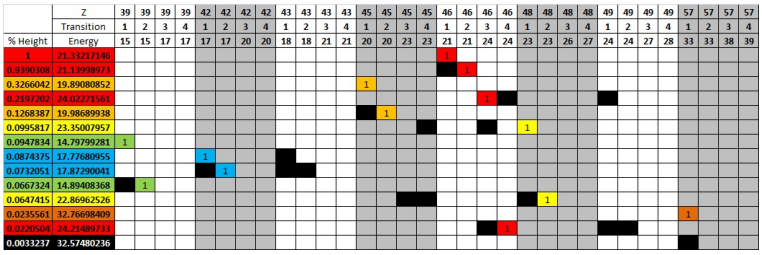
Fig. D.5: Line 4



	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998973																	1	1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1				1								
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

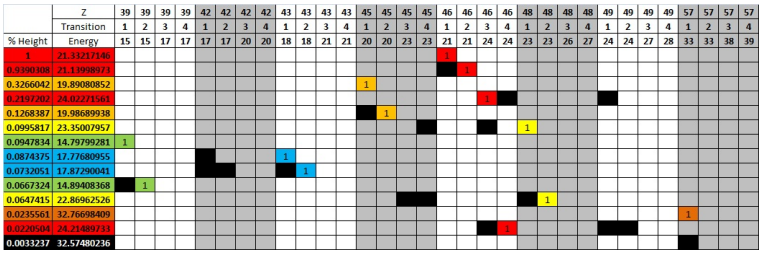
Fig. D.6: Lines 5, 6 and 7

element, both Z = 42 and Z = 43 have a first transition match possible.



	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998973																	1	1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1				1								
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

Fig. D.7: The first possible final result assuming Z = 42 was the choice



	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998973																	1	1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1	1				1								
0.1268387	19.98689938													1	1																		
0.0995817	23.35007957																1			1		1											
0.0947834	14.79799281	1																															
0.0874375	17.77680955					1				1																							
0.0732051	17.87290041					1	1			1	1																						
0.0667324	14.89408368	1	1																														
0.0647415	22.86962526															1	1					1	1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1	1					1	1						
0.0033237	32.57480236																													1			

Fig. D.8: The second possible final result assuming Z = 43 was the choice

In the event of a tie, what the code does is first it creates copies of the results

matrix so far (one copy for each tied element) and then starts with the lowest Z value of the tied elements and blanks out the other options. The code will then go through and finish all the rows under the assumption that the tied option goes to the first Z value's transition. Once that is done, it then moves to the next copy of the results matrix and repeats the process under the assumption that the choice belongs to the second possible Z value's transition. After it has finished for each of the tied Z values, the results of each option are compared and the option which produces the least number of unique Z values[6] is chosen.

In the example being worked on however, the situation is still not as clear. As can be seen in both figures D.7 and D.8, the final results from both choices are the same (with the obvious exception of one have Z = 42 and the other Z = 43). In the case where no benefit can be seen from either choice, at the moment the code simply chooses the higher Z value and this choice can be seen in figure D.9.

	Z	39	39	39	39	42	42	42	42	43	43	43	43	45	45	45	45	46	46	46	46	48	48	48	48	49	49	49	49	57	57	57	57
	Transition	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
% Height	Energy	15	15	17	17	17	17	20	20	18	18	21	21	20	20	23	23	21	21	24	24	23	23	26	27	24	24	27	28	33	33	38	39
1	21.33217146																	1															
0.9390308	21.13998971																		1														
0.3266042	19.89080852													1																			
0.2197202	24.02271561																			1													
0.1268387	19.98689938														1																		
0.0995817	23.35007957																				1												
0.0947834	14.79799281	1																															
0.0874375	17.77680955									1																							
0.0732051	17.87290041					1	1				1																						
0.0667324	14.89408368		1																														
0.0647415	22.86962526															1	1						1										
0.0235561	32.76698409																													1			
0.0220504	24.21489733																			1						1	1						
0.0033237	32.57480236																													1			

Fig. D.9: Final choice of the tie situation

The next five rows are all straight forward and have been demonstrated previously. For the final row, after all the other transitions have been assigned and the available choices reduced, there does not exist an appropriate match for the last determined energy. This could mean that in the least squares fitting process, a bit of noise was misinterpreted and so a false energy line was given. In this case, the line is assumed to be in error and so removed from the results, however it should be noted that except for this slightly manipulated real results example, no determined line has not been able to be fitted.

The final 6 steps are shown in figure D.10 on the following page.

D.3 Interpreting the Results

After all the determined energy lines have been matched to appropriate transition lines, the final results are not concerned with what transitions are detected, only with what Z values are in the part of the sample visible by this pixel. So to simplify the final results, a matrix consisting of the Z values found and the

[6]Which also implies this option has the strongest contributions from each determined Z value

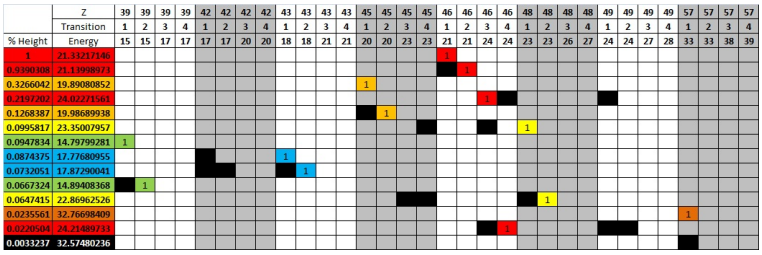


Fig. D.10: The final 6 lines of analysis

total % contributions each Z value made to the pixel's spectrum[7].

Z	% Contribution
39	5.14
43	5.11
45	14.42
46	69.36
48	5.23
57	0.75

Tab. D.1: Final Results from this example showing the detected Z values and how much they contribute to the final spectrum

These numbers are then used to make a visual image of the sample. One possible visualisation is to make a matrix of Z values which are the most intense element in each pixel, as shown in figure D.11 on the next page. A positive of this image is that you can see the image as a whole and get an idea of how the sample is arranged, but this visualisation sort of assumes that each pixel should be seeing only one element and in the case of Pd and Ag where the analysis had difficulty telling the two elements apart it creates the speckling effect seen in the left half.

A second possible visualisation is to make an image for each unique Z value identified and each image is of the % contribution that element made to that pixel as shown in figure D.12 on the following page. This better separates the regions of each element, especially for the elements which are difficult to identify separately, but it does mean that you can't view all the information in a single image.

[7]this was calculated by simply adding the intensities from all the determined transition for each Z value and then converting these sums into appropriate % values

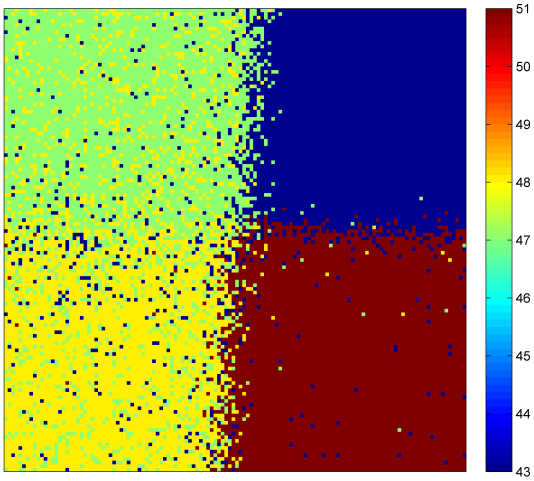


Fig. D.11: A visual image of the most intense element in each pixel. Colour represents Z value

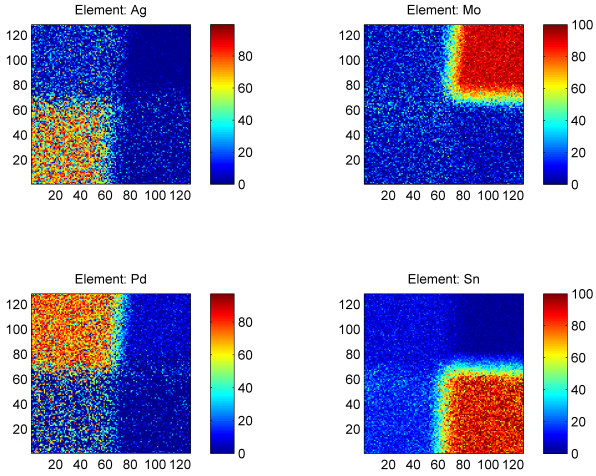


Fig. D.12: Visual images for each detected element and its % contribution to that pixel. Colour represents % contribution

E
NIST Values

Tab. E.1: Standard Transition energies sourced from NIST

Element	**KL2 (keV)**	**KL3 (keV)**	**KM3 (keV)**	**KN3 (keV)**
Ni	7.460	7.478	8.268	
Cu	8.028	8.048	8.907	
Zn	8.616	8.639	9.574	
Ga	9.225	9.252	10.266	
Ge	9.856	9.887	10.984	
As	10.508	10.543	11.727	11.866
Se	11.182	11.223	12.497	12.655
Br	11.878	11.924	13.292	13.471
Kr	12.596	12.648	14.113	14.314
Rb	13.336	13.395	14.962	15.187
Sr	14.098	14.165	15.836	16.087
Y	14.883	14.958	16.738	17.015
Zr	15.691	15.775	17.667	17.971
Nb	16.521	16.616	18.624	18.956
Mo	17.374	17.479	19.607	19.967
Tc	18.251	18.368	20.619	21.006
Ru	19.151	19.280	21.658	22.077
Rh	20.073	20.216	22.724	23.176
Pd	21.021	21.177	23.820	24.302
Ag	21.991	22.163	24.943	25.458
Cd	22.984	23.174	26.095	26.644
In	24.002	24.210	27.276	27.861
Sn	25.044	25.271	28.486	29.109
Sb	26.111	26.359	29.726	30.388
Te	27.202	27.472	30.996	31.699
I	28.318	28.612	32.295	33.042
Xe	29.458	29.778	33.625	34.408