Algorithms to process data from the MARS molecular imaging system

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy by Niels de Ruiter

University of Otago 2015
Abstract

The MARS molecular imaging project aims to research, develop, and commercialize a spectral computed tomography (CT) system. My thesis describes the work I performed to reconstruct volumes from the MARS prototypes. The challenge was to develop the algorithms while maintaining image processing software that met the immediate needs of the MARS team.

Using the Medipix detector, the current prototype is capable of simultaneously scanning up to 8 energy bins. Every additional energy bin improves the potential for material discrimination at the molecular level. Data acquired from the MARS prototypes are a collection of exposures over various geometric transformations of the source, detector, and subject. To process these, I developed two applications, mPPC (MARS Preprocessing Chain) and mART (MARS Algebraic Reconstruction).

The application mPPC prepares data for reconstruction while also improving the image quality. In particular, various issues that result from the Medipix detector are addressed in the preprocessing software.

The application mART reconstructs the preprocessed data into volumes. It adopts a variation of SART to simultaneously reconstruct all the energy bins. The results are a good balance between quality and performance.

To link the components of the data processing chain, all the software adopts the stable and popular DICOM standard. The DICOM standard provides formats to package the data while also providing protocols for both storing (and backup) and transferring the data.

To summarize, the outcomes of my thesis are two applications which, together, perform all the necessary steps to reconstruct high quality volumes from the MARS system. With the addition of the DICOM standard to store and transfer the data, the result is a data processing chain which fulfils the needs of the MARS team.
Table of Contents

List of Figures x
List of Tables xviii
Glossary xix
Academic Contributions xxiii

Chapter 1: Introduction 1
  1.1 Research objectives and outcomes ................................ 1
  1.2 The emergence of spectral CT ....................................... 3
  1.3 MARS molecular imaging prototype ................................... 5
  1.4 Medipix detectors for spectral CT ..................................... 5
  1.5 MARS molecular imaging team ....................................... 6
  1.6 Research Contributions ............................................... 8
  1.7 An outline of my thesis .............................................. 9

Chapter 2: An overview of the MARS system 12
  2.1 MARS camera technology ............................................. 12
  2.2 MARS gantry technology ............................................. 16
  2.3 MARS scanning modes ............................................... 18
  2.4 Summary ............................................................. 20

Chapter 3: MCTBuilder: An initial preprocessing system 21
  3.1 Summary of the condition of the MARS system ..................... 22
      3.1.1 MARS scanner hardware ........................................ 22
      3.1.2 MARS scanning software ....................................... 23
      3.1.3 MARS pre-reconstruction processing software ................. 23
      3.1.4 MARS reconstruction software ................................ 24
      3.1.5 MARS material decomposition software ....................... 24
### 3.1.6 MARS visualization software .......................... 24

### 3.2 Algorithms for the pre-reconstruction processing chain .......... 25
  3.2.1 Darkfield masking ..................................... 25
  3.2.2 Flatfield normalization ................................. 27
  3.2.3 Exposure stitching ...................................... 28
  3.2.4 Dead detector element inpainting ........................ 29
  3.2.5 Sinogram conversion .................................... 30
  3.2.6 Summary of algorithms ................................ 30

### 3.3 MCTBuilder design ........................................ 31
  3.3.1 Navigation of data ...................................... 33
  3.3.2 Live preview of images in MCTBuilder .................... 34
  3.3.3 Import/Export of data .................................. 35
  3.3.4 Graphical user interface ................................. 36

### 3.4 Summary .................................................. 37

---

### Chapter 4: mART: An algebraic reconstruction application 39

#### 4.1 Summary of the condition of the MARS system .............. 40
  4.1.1 MARS scanner hardware ................................ 40
  4.1.2 MARS scanning software ................................ 41
  4.1.3 MARS pre-reconstruction software ......................... 41
  4.1.4 MARS reconstruction software ........................... 42
  4.1.5 MARS material decomposition software .................... 42
  4.1.6 MARS visualization software ............................. 42

#### 4.2 Motivation and goals ..................................... 43

#### 4.3 A review of reconstruction algorithms ...................... 44
  4.3.1 Back-projection ......................................... 45
  4.3.2 Iterative techniques .................................... 46
  4.3.3 Optimization techniques ................................ 47
  4.3.4 Statistical Approaches ................................ 48
  4.3.5 Outcomes of the review ................................ 49

#### 4.4 The basics of algebraic reconstruction ...................... 50
  4.4.1 Kaczmarz solution ...................................... 52
  4.4.2 SART .................................................... 52
  4.4.3 Summary of basics ...................................... 53
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5 Geometric model of the MARS system</td>
<td>54</td>
</tr>
<tr>
<td>4.5.1 Detector element transform</td>
<td>56</td>
</tr>
<tr>
<td>4.5.2 Detector transform</td>
<td>56</td>
</tr>
<tr>
<td>4.5.3 Camera mount transform</td>
<td>57</td>
</tr>
<tr>
<td>4.5.4 X-ray source position</td>
<td>58</td>
</tr>
<tr>
<td>4.5.5 Gantry transform</td>
<td>59</td>
</tr>
<tr>
<td>4.5.6 Subject transform</td>
<td>59</td>
</tr>
<tr>
<td>4.6 Sampling techniques for algebraic reconstruction</td>
<td>60</td>
</tr>
<tr>
<td>4.6.1 Ray driven techniques</td>
<td>60</td>
</tr>
<tr>
<td>4.6.2 Voxel driven techniques</td>
<td>62</td>
</tr>
<tr>
<td>4.6.3 Implementation</td>
<td>63</td>
</tr>
<tr>
<td>4.6.4 Summary of sampling techniques</td>
<td>63</td>
</tr>
<tr>
<td>4.7 Implementing mART</td>
<td>64</td>
</tr>
<tr>
<td>4.7.1 mART version 1</td>
<td>64</td>
</tr>
<tr>
<td>4.7.2 mART version 2</td>
<td>71</td>
</tr>
<tr>
<td>4.8 Summary of mART</td>
<td>75</td>
</tr>
<tr>
<td>Chapter 5: Digital Imaging and Communications in Medicine</td>
<td>77</td>
</tr>
<tr>
<td>5.1 Summary of the condition of the MARS system</td>
<td>78</td>
</tr>
<tr>
<td>5.1.1 MARS scanner hardware</td>
<td>78</td>
</tr>
<tr>
<td>5.1.2 MARS scanner software</td>
<td>78</td>
</tr>
<tr>
<td>5.1.3 MARS reconstruction software</td>
<td>80</td>
</tr>
<tr>
<td>5.1.4 MARS material decomposition software</td>
<td>81</td>
</tr>
<tr>
<td>5.1.5 MARS visualization software</td>
<td>81</td>
</tr>
<tr>
<td>5.2 An overview of DICOM</td>
<td>81</td>
</tr>
<tr>
<td>5.2.1 Data structures in DICOM</td>
<td>82</td>
</tr>
<tr>
<td>5.2.2 Networking in DICOM</td>
<td>85</td>
</tr>
<tr>
<td>5.3 DICOM information object definitions for the MARS system</td>
<td>87</td>
</tr>
<tr>
<td>5.4 DICOM IOD for raw projection data</td>
<td>89</td>
</tr>
<tr>
<td>5.5 DICOM IOD for normalized projection data</td>
<td>92</td>
</tr>
<tr>
<td>5.5.1 Data types, binning, and precision</td>
<td>94</td>
</tr>
<tr>
<td>5.5.2 Dead detector element representation</td>
<td>95</td>
</tr>
<tr>
<td>5.5.3 Normalized projection data summary</td>
<td>95</td>
</tr>
<tr>
<td>5.6 DICOM IOD for reconstructed volume data</td>
<td>95</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>5.6.1</td>
<td>Multi-energy and multi-frame images</td>
</tr>
<tr>
<td>5.6.2</td>
<td>Hounsfield units, linear attenuation, and density</td>
</tr>
<tr>
<td>5.6.3</td>
<td>Final IOD design</td>
</tr>
<tr>
<td>5.7</td>
<td>Summary of DICOM design</td>
</tr>
<tr>
<td>6.1</td>
<td>Summary of the condition of the MARS system</td>
</tr>
<tr>
<td>6.1.1</td>
<td>MARS scanner hardware</td>
</tr>
<tr>
<td>6.1.2</td>
<td>MARS scanning software</td>
</tr>
<tr>
<td>6.1.3</td>
<td>MARS pre-reconstruction processing software</td>
</tr>
<tr>
<td>6.1.4</td>
<td>MARS reconstruction software</td>
</tr>
<tr>
<td>6.2</td>
<td>Transition to DICOM</td>
</tr>
<tr>
<td>6.3</td>
<td>Characterization of Medipix signals</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Ideal detector element</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Dead detector elements</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Erratic detector elements</td>
</tr>
<tr>
<td>6.3.4</td>
<td>Unstable detector elements</td>
</tr>
<tr>
<td>6.3.5</td>
<td>Truncated detector elements</td>
</tr>
<tr>
<td>6.3.6</td>
<td>Drifting detector elements</td>
</tr>
<tr>
<td>6.3.7</td>
<td>Identification and characterization</td>
</tr>
<tr>
<td>6.3.8</td>
<td>Algorithms to remove bad detector elements</td>
</tr>
<tr>
<td>6.4</td>
<td>Filtration of ring artefacts</td>
</tr>
<tr>
<td>6.5</td>
<td>Summary</td>
</tr>
<tr>
<td>7.1</td>
<td>Summary of the condition of the MARS system</td>
</tr>
<tr>
<td>7.1.1</td>
<td>MARS scanning software</td>
</tr>
<tr>
<td>7.1.2</td>
<td>MARS reconstruction software</td>
</tr>
<tr>
<td>7.2</td>
<td>Motivation and methodology</td>
</tr>
<tr>
<td>7.3</td>
<td>Results</td>
</tr>
<tr>
<td>7.4</td>
<td>Summary of truncation study</td>
</tr>
<tr>
<td>8.1</td>
<td>Summary of the condition of the MARS system</td>
</tr>
<tr>
<td>8.1.1</td>
<td>MARS scanner hardware</td>
</tr>
</tbody>
</table>
10.1.2 Creation of mART ......................... 160
10.1.3 Integration of DICOM .................... 161
10.1.4 Evolution of MCTBuilder into mPPC ..... 163
10.1.5 Study of truncation ..................... 164
10.1.6 Design of a MARS processing server ..... 165
10.1.7 Study of random sampling techniques .... 166
10.1.8 Steps towards automation ............... 166

10.2 Summary of the academic contributions ...... 167

10.3 Future endeavours .......................... 168
10.3.1 Short term outlook ....................... 168
10.3.2 Mid term outlook ......................... 170
10.3.3 Long term outlook ....................... 170

10.4 Afterword .................................. 171

References .................................. 172

Appendix A: MARS processing server .......... 186
A.1 Motivation for Image Processing Server ..... 186
A.2 Unified Procedure Step ..................... 187
A.3 Design of a Processing Server .............. 188
A.4 Basic Infrastructure ........................ 189
A.5 MARS Transfer ................................ 190
A.6 Conclusion .................................. 192
List of Figures

1.1 The differences between single energy, dual energy, and spectral CT (using the MARS system). The x-ray source produces a spectrum of x-rays (like white light). In the case of dual CT, two x-ray sources produce two different spectra based on the x-ray tube material and voltage settings. With spectral CT, the detector can filter out specific energy ranges from the photon cloud. The result is colour information which provides more details about the composition of the subject. Image courtesy of the MARS group, originally published in my Masters thesis. .................................................. 4

2.1 An exposure of a CdTe MXR Medipix detector. Issues with impurities and bonding result in a loss of usable pixels. The black holes are dead regions while the wrinkle patterns are a distortion of the signal. ................................................................. 13

2.2 The absorption profiles of Si, GaAs, CdTe, CdZnTe between 20 keV and 120 keV with a thickness of 2 mm. Ideally, the profile should be 100% efficient over the whole range. ................................. 14

2.3 An exposure using the charge summing mode. Notice that many bright spots come in 2×2 sets of detector elements due to the coupling effects of charge summing mode. Note that the large black regions are dead portions of this detector. ................................. 15

2.4 A view of the scanner design showing the subject mount, and the gantry which houses the camera and x-ray source mounts. The x-ray source mount and camera mount a both attached to the primary gantry. The gantry is independent of the subject mount to prevent dynamic influences between the two. ................................. 17
2.5 A comparison between circular scanning (top) and helical scanning (bottom). Note that the two images are two separate scans of a mouse showing two different regions. However, the image resulting from circular scanning clearly has visible vertical lines at regular spaces. These are artefacts resulting from the overlap between the scan sections.

3.1 The flow chart of the original preprocessing chain. The results of pre-computed steps are considered to be constants in the main chain. Therefore, darkfield masking and flatfield normalization are 1 input, 1 output tasks.

3.2 The three planes of projection data including the projection \((x,y)\), sinogram \((x,\theta)\), and another unnamed plane \((y,\theta)\). As projections are acquired at regular intervals around a subject, stacking the projections along the angle \(\theta\) results in an image with a sine wave-like appearance. The black parts of the image are bad pixels.

3.3 The effects of the algorithms in the pre-reconstruction processing chain. These steps include darkfield masking (1), flatfield normalization (2), stitching (3), and inpainting (4). Dead detector elements are marked in red. In this example, only two exposures were stitched, and the darkfield masking and inpainting did not have much work to do.

3.4 A screenshot of MCTBuilder. The top left presents a windows explorer like view for navigation. The tools are presented as a wizard in the bottom left. The preview is shown on the right. Dead detector elements are marked as red.

4.1 An illustration of the Kaczmarz process on the simple case of 2 solution hyperplanes in 2D space. The initial solution at the dot is repeatedly projected onto the hyperplane solutions until the common point is reached.
4.2 An illustration of the first two steps of the Kaczmarz solution in two arbitrary directions. The blue ray is the result of the first step while the red is the result of the second step. The line profile of the red step on the right illustrates how the blue step appears as an impulse which remains in the solution after the red step has completed. 

4.3 The scanner coordinates use the tangential, radial, and axial axis.

4.4 A screenshot of the first versions of mART. The interface shows reconstruction parameters on the left together with a preview during reconstruction on the right.

4.5 The model for calculating the radius. The algorithm uses two right angled triangles to calculate the radius.

4.6 The model for calculating the length. The algorithm calculates three components using the back of the volume. Therefore, the radius must be already known.

4.7 An example of the result of voxels which are too small when treating rays as thin lines. The striped, curved patterns are the result of rays missing some of the small voxels. The large black gaps are part of the same pattern down the axial axis.

4.8 The first test was a Shepp Logan phantom with no noise and minor blurring to make the problem better conditioned. The left is the original and the right is the ART reconstruction.

4.9 The second test was of real data on a phantom scan called FatCaFe. The vials contain solutions of water, lipid, calcium, and iron.

4.10 Reconstructions on a phantom scan called FatCaFe with mART and Octopus CT. Notice the obvious improvements in the image noise levels and how the brightest tube has a well defined edge with mART and some ghosting with Octopus CT.

5.1 A sample exposure from a dual Medipix detector with a GaAs sensor crystal. Notice that once again, the image has significant dead regions as well as distinct wrinkle patterns.
5.2 A comparison of performance when reconstructing with 1, 2, 4, 8, 16, and 32 cores on a single node of a Power 7 based supercomputer. The black line, shows a prediction for the expected performance with 64 cores by fitting a power function to the acquired data. 80

5.3 The data flow between MARS applications. The purple represents the scanner workstation. The blue represents the data storage server. The green represents the image processing workstation. The red represents the visualization workstation. Note that some of the workstations may have the same physical hardware in practice. 88

6.1 The new interface for mPPC. The file list shows all the files in a scan in the order of acquisition. The algorithms are presented in a simple list. Lastly, the preview shows the image as well as the tags. 103

6.2 A normalized exposure with the added thresholding. Dead detector elements are marked in red. Notice that many bright and dark patches still remain in the image. 105

6.3 A plot of the repeated values of a detector element with ideal behaviour. The bottom right shows the histogram of these values. There are some expected fluctuations but the average value is stable across all exposures. 106

6.4 A plot of the repeated values of a detector element with erratic behaviour. The bottom right shows the histogram of these values. Note how it is impossible to determine where the expected value should be as different regions of the plot tend towards different values. The histogram shows no obvious peak with random spikes over a wide range of values. 107

6.5 A plot of the repeated values of a detector element with unstable behaviour. The top right shows the histogram of these values. In this case the instability led to the detector element switching on and off. 108

6.6 A plot of the repeated values of a detector element with a truncated response behaviour. The bottom right shows the histogram of these values. In this case the truncation is moderate, the pixel has tolerable behaviour otherwise. 109
6.7 A plot of the repeated values of a detector element which drifts over time. The top right shows the histogram of these values. In this case the drift is severe along with excessively noisy behaviour. . . . 110

6.8 The results of the Poisson distribution test when applying a threshold of 20%. The left image is the normalized exposure while the right has had the Poisson distribution test applied. Not all bad detector elements were eliminated but the results are better than before. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 113

6.9 The effects of mPPC’s pre-reconstruction processing chain. The steps include darkfield masking (1), flatfield normalization (2), the poisson distribution test (3), dilation (4), user defined masking (5) and the revised inpainting (6). Red areas are detector elements marked as bad. . . . . . . . . . . . . . . . . . . . . . . . . . . 116

6.10 A sinogram section of a ring filtered dataset showing an unstable pixel which couldn’t be filtered. Notice how the line is chequered in an irregular fashion. Problems such as these are difficult to deal with.118

6.11 A region with the corner of an object. Taking the median of the whole 2D region would represent the black structure while a 1D median in either direction would give the grey structure. . . . 119

6.12 The effects of mPPC’s ring filter. The top is the unfiltered image while the bottom is the filtered image. The sharp jumps in the image were scanner faults which led to exposures being dropped. . . . 120

7.1 An illustration showing the typical region of interest which is exposed by 360 degrees of coverage. The truncated region is inside the scanner, the further away from the center of rotation, the less angular coverage there is. Lastly, outside the scanner is no angular coverage at all. . . . . . . . . . . . . . . . . . . . . . . . . . 125
7.2 The results from the truncation test. From top left, the results show the reconstructions with 16, 14, 12, 10, and 8 camera positions. The second row shows the difference between the ground truth (16 camera positions) and the test result, 16-14, 16-12, 16-10, and 16-8. The third row shows the same difference images as the second but with a tight window/level to highlight the differences in the interior region.

7.3 The new GUI for mART which includes the option to explicitly set the volume radius, length, position and the voxel size. A button then uses the old procedure to provide default settings if desired.

8.1 An exposure from a CdZnTe detector with massive distortion. This detector is unusable as all the pins, which should be completely straight, have visible distortion and shear.

8.2 A sinogram section of a set of flatfield exposures. The bright vertical line is a single exposure which flashed. The exposure should be removed before image processing begins.

8.3 A single exposure showing a volume with an irregular grid, a circular x-ray disc, and a square detector. The gantry rotates during the exposure. The shaded section outlines the area where photons may be absorbed during the exposure (ignoring scatter).

8.4 Slice 29 from 1000 steps of a reconstruction with raytraced rays for forward and backward projection. The image on the right is the 1000th update. Notice the subtle striped pattern orthogonal to the direction of the projection in the update image.

8.5 Slice 29 from 1000 steps of a reconstruction with randomly sampled rays for forward and backward projection together with its 1000th update. Notice that no unexpected patterns are visible in the update image on the right.

8.6 Slice 29 from 1000 steps of a reconstruction with sampling from three tables of 65535 random numbers for forward and backward projection together with its 1000th update. Notice the strong curved patterns which have a clear impact on the quality of the reconstruction.
8.7 Slice 29 from 1000 steps of a reconstruction with hybrid sampling together with its 1000th update. The quality is equivalent to the fully raytraced example. .................................................. 140

9.1 Slice 41 after 1 step of reconstructing a phantom. From the left to the right, the first image uses the path length as the weighting factor, the middle image uses the mean, the last image is the difference between the two. The difference image only shows minor differences at edges and the boundaries between rays. .............................................. 149

9.2 Slice 71 of a phantom dataset which has a high number of outliers corrupting the image. The left image has no additional weighting while the right image adopts the Poisson distribution test to weight each detector element. Notice that a good portion of the outliers have been removed with the Poisson distribution test. Lastly, note that the dark streaks are due to dead regions in the detector which will be filled in over time. ...................................................... 151

9.3 Slice 71 of a phantom dataset which has a high number of outliers corrupting the image. The left image only has the Poisson distribution test applied while the right image also adopts the outlier test. Notice that all the outliers have now been removed with the outlier test. The dark streaks are due to dead regions in the detector which will be filled in over time. ...................................................... 153

9.4 A coronal view of a center of a phantom dataset. Notice the harsh lines near the middle. These correspond to a cylindrical artefact which appears as rings in each slice. This correlation between rings is the clue that the cause is the imbalance between exposures. .... 154

9.5 The results of applying a blending function to smooth the transition between exposures. The left image is the original reconstruction without blending while the right image has the blending function applied. The rings are almost completely gone after the blending function is applied. ...................................................... 155
A.1 An overview of the processing server in a larger DICOM system. In this example, the MARS system is one of a few research groups that use a centralized PACS. The red paths show the modality worklist, the blue paths show the data transfer, the green paths show the unified procedure step.

A.2 The components of the processing server. The server is built up of a DICOM server to interact with the PACS, a job manager to schedule jobs based on the system resources available, and the job plugins which perform the tasks.

A.3 A screenshot of MARS Transfer. The top left contains the query terms. The center contains the query results. The right contains the jobs which currently include downloading and uploading tasks. The bottom is the job schedule which contains a single job in this instance.
List of Tables

5.1 An example of the indexing scheme of the nuclear medicine multi-frame module. This example contains two energies and two time steps. .................................................. 90
5.2 The nth detector module which contains the geometric and indexing tags for each detector in the image. Only the relevant tags are shown. The indexing is achieved with the element number. The first two digits (XX) start from 10 and increase per detector module. Note that the group and element numbers are given in hexadecimal form. FD stands for double precision floating point. US stands for unsigned short. ........................................ 90
5.3 The gantry motor module which contains the gantry, camera, subject, and source positions. Only the relevant tags are shown. ........ 91
5.4 The image information module which contains the indexing scheme for the image. Only the relevant tags are shown. ............... 91
5.5 The information object definition for the raw data. The three private modules defined in this chapter are shown in bold at the bottom. Also shown in bold is the multi-frame system from the NM IOD. The other modules are straight from the definition of the CR IOD. Optional private modules are not shown here. M stands for mandatory, C stands for conditional, and U stands for user option. 93
5.6 The information object definition for the reconstructed data. Currently, there are no private modules used. The bold modules are from the NM IOD. The other modules are from the CT IOD. M stands for mandatory, C stands for conditional, and U stands for user option. ........................................ 98
8.1 Average time taken per step for each reconstruction. ..................... 141
Glossary

- **ART (Algebraic Reconstruction Technique)** - This is the general term for algebraic iterative reconstruction algorithms. It is also the common name for the Kaczmarz solution which is the most basic of algebraic solvers for the problem $A\vec{x} = \vec{b}$.

- **ASIC (Application-Specific Integrated Circuit)** - A silicon wafer containing circuits for a specific purpose. In the context of this thesis ASIC is often used to refer to the chip layer of the Medipix detector.

- **ASIC element** - A pixel of the physical hardware in a Medipix detector. These can work together to acquire super-pixels.

- **Biomarker** - This is a term referring to a known substance in a biological subject. An example of this are the proteins which are specific to different types of cancer. When trying to detect a type of cancer, it problem involves searching for the known biomarker.

- **CR (Computed Radiography)** - A computer enhanced radiography imaging modality (commonly just called x-ray images).

- **CT (Computed Tomography)** - A computer aided process to recover an image from a set of projections. It is commonly used to refer to x-ray computed tomography as this is the most common use of tomography.

- **CAT (Computer Aided Tomography)** - An earlier name for CT which is commonly used.

- **Darkfield** - An exposure captured with the x-ray source off. This means that there should be no photons present except for the odd photon from the environment.
• Detector element - A pixel in the acquired image from a Medipix detector. This may be formed by one or more ASIC elements.

• DICOM (Digital Imaging and Communications in Medicine) - This is a standard for communicating, transferring, storing, archiving, and using medical data. It has many sections, each of which tend to different aspects of dealing with medical data.

• DICOM IOD (Information Object Definition) - These are the specifications for specific DICOM data entities. For example, a CT image would be created using the CT IOD. The specification is a built up from a list of DICOM modules, each of which is marked as mandatory, conditional, or optional.

• DICOM module - These are the building block specifications for DICOM data entities. They are made up of a list of tags with a common purpose. For example, the General Image DICOM module contains all tags needed to describe an image including width, height, bits per pixel, etc.

• DICOM tag - These are the basic building blocks for DICOM data entities. They represent a single piece of information. For example, the pixel data tag contains an array of pixels, nothing more, nothing less.

• Dilation - A process which expands the coverage of a mask. Typically, dilation is done on a boolean image to expand white or black pixels to their immediate neighbours.

• Endianess - The order in which a number is stored e.g. twenty four or four and twenty. This has important consequences when communicating data between computers as they could have a different type.

• FDK (Feldkamp-Davis-Kress) - An approximation which allows filtered back-projection methods to operate on cone-beam CT datasets. It is an inexact method which is fast an efficient to calculate.
• Flatfield - An exposure captured without a subject. This means that we can approximate the photon count produced by the x-ray source if we assume the linear attenuation of air to be zero.

• Inpainting - A process to fill in holes in images with a reasonable estimate. Commonly, reasonable means a natural appearing fill. In this thesis, reasonable means an exact recovery of the missing data.

• Medipix - This is the name of a series of photon counting detectors produced by CERN. This forms the core of MARS molecular imaging technology.

• MRI (Magnetic Resonance Imaging) - A medical imaging modality which uses magnets to align protons. This is very effective when imaging water based fluids which has a high count of protons (hydrogen).

• NM (Nuclear Medicine) - A medical imaging modality which uses radioactive substances to target biomarkers and map the objects of interest.

• PACS (Picture Archiving and Communication System) - This is the common name for a DICOM based server. Each PACS implements a list of DICOM services such as the ability to store data in a database and query the contents. No PACS implements the complete DICOM standard as it is too large and is constantly changing. Instead, each PACS reports a conformance statement to specify what services it provides.

• PET (Positron Emission Tomography) - A medical imaging modality which uses a positron creation event to map radioactive sources inside the subject. This is effective when targeting biomarkers in cancer.

• Phantom - A term for a dataset or sample whose purpose is solely to test a feature or perform a specific measurement. A good example is the Shepp Logan phantom, a simulated CT dataset which is a common testing platform for CT reconstruction algorithms.

• SART (Simultaneous Algebraic Reconstruction Technique) - This is an iterative solver for $A\tilde{x} = \tilde{b}$ which uses the weighted average of a subset of the
measured data points to update the solution. It is considered a good balance between the speed of the Kaczmarz solution and the quality of SIRT.

- **Singoram** - An alternative representation of projection data which has the unique appearance of sine waves.

- **SIRT (Simultaneous Iterative Reconstruction Technique)** - This is an iterative solver for $A\vec{x} = \vec{b}$ which uses the weighted average of the complete set of measured data points to update the solution. As each step requires looking at the complete dataset, it produces high quality results. However, it comes at the cost of performance.

- **Spectral CT** - This is a CT based modality which accomplishes two things. Firstly, it acquires multiple CT datasets. Secondly, each CT dataset is acquired over a different portion of the x-ray part of the electromagnetic spectrum. In other words, different x-ray colours. This additional information improves the ability to distinguish and quantify the composition of objects.

- **Stitching** - A process to combine multiple images into a single, unified image.

- **TCP (Transmission Control Protocol)** - A modern protocol for networking that is supported by all operating and computing systems.

- **TLS (Transport Layer Security)** - A modern network encryption and security protocol that may be applied on top of a TCP network connection.

- **VR (Value Representation)** - This designates the type of a DICOM tag. The type may be a string (text), a signed/unsigned integer, a real number, a date or time, or a unique identifier to name the common types.

- **VR (Value Multiplicity)** - This specifies the restrictions on the number of values a DICOM tag must have. A VR of $3 - n$ means that there must be at least three entries but there may be more.
Academic Contributions

During the course of this thesis the aim was to provide an image processing solution for reconstructing spectral CT datasets from the MARS molecular imaging system. This solution would be freely available to the MARS team and assist them in their works in research and development, and biological and pre-clinical studies. As such the following publications and achievements are listed.

Peer-reviewed journal articles


   At the start of my thesis, PCA offered an easy, qualitative form of material decomposition. This paper is the culmination of our investigation in using PCA for material decomposition of MARS data.

   My involvement includes the visualization of the PCA resolved multi-contrast mouse dataset. More importantly, the outcomes of this paper motivated the creation of the mini application mPCA so that PCA could be easily used by all members of the MARS team. mPCA was the first contribution of my thesis.


   This paper was the follow up study of the behaviour of a Medipix 2 detector with a CdTe sensor crystal. As an active observer during this study, the work presented here motivated the action to interpolate between sets of flatfield exposures in MCTBuilder. This was to compensate for the observed temperature dependencies
in the detectors. My direct involvement in the paper was to review the contents in preparation for publication.


*This paper forms part of a clinical study into the classification of human excised atheromatous plaques using spectral molecular imaging. As each application I developed was released to the team, pre-clinical studies such as these formed the first validation of my software.*

*To this end, I involved myself with this study to assist with the necessary image processing and reconstruction using mPPC and mART. For my thesis, I carefully observed the work flow of Raja Panta to validate the user interfaces. More importantly, I made sure to personally review the image quality to ensure it was sufficient for the study to proceed.*


*This paper was a review of the behaviour of Medipix 3.1. In particular, the stability of Medipix 3.1 was characterized as it was the biggest issue for Medipix 3.0.*

*I participated in the study by helping prepare the datasets for analysis ensuring that a fair test was achieved between the various case studies. This study also motivated the eventual removal of the flatfield interpolation in mPPC as Medipix 3.1 and beyond were stable enough that multiple sets of flatfield exposures per scan were no longer necessary.*


This is the first of two publications to present spectral CT data from the MARS molecular imaging system to the wider scientific community. The paper itself presents a study of imaging lamb tissue. Along side the paper, the raw, normalized, reconstructed and material decomposition datasets were uploaded for public access at http://hdl.handle.net/10092/8531.

I oversaw the image processing up to material decomposition to ensure that the data was well presented.


This is the second of two publications to present spectral CT data from the MARS molecular imaging system to the wider scientific community. The paper itself presents a validation of reducing beam hardening effects through narrow energy bins. Along side the paper, the raw, normalized, reconstructed and material decomposition datasets were uploaded for public access at http://hdl.handle.net/10092/8851.

I worked with Kishore Rajendran, reviewing both his work flow and his results using mPPC and mART. This provided a validation of both applications at the later stages of my thesis while ensuring that the study results were an accurate representation of MARS technology.

xxv
Refereed conference proceedings


   This publication published the completion of the first version of mART. It demonstrated that the intended goals for mART had been obtained while outlining future plans for the software. The goals for mART was to produce an in-house reconstruction solution that could reconstruct any MARS dataset with equal or better quality then Octopus CT. The image quality study was led by Nate Tang who presented at the IVCNZ conference 2012.


   This publication illustrates the progress of the material decomposition algorithm developed by Christopher Bateman. All material decomposition results were obtained using the reconstructed data through mPPC and mART.

   In this case, I did not assist with the image processing as the end goal of any software is to eliminate the need for oversight. However, I did review the processed images to ensure that both the workflow and results were as expected. This was a good validation of how both mPPC and mART contribute to the MARS team and the science produced.

To be published in the near future


   This paper describes a technique to denoise energy bins by exploiting their structural similarities. My role in this work was as a consultant. In particular, I
verified that the algorithm preserves edge and energy signal. At a later stage, this algorithm will become part of mPPC or mART.

Invited disseminations

The following are papers and presentations which were presented by Assoc. Prof. Anthony Butler and Prof. Philip Butler on behalf of the MARS team.


**Awards/financial grants based on a substantial assessment**

1. 2011, Recipient of a PhD scholarship.

   *I was awarded a “University of Otago Doctoral Scholarship” which provides funding for up to 36 months of study. From 1 January 2012, the University Council approved an increase to the scholarship’s stipend which made the scholarship equivalent to a “University of Otago Prestigious Scholarship”.*

**Users of my software**

1. All members of the MARS team including the current 16 PhD students from the universities of Canterbury and Otago.

2. Offsite members of the MARS team including Dr Nannette Schleich in the University of Otago (Wellington).

3. External partners including Mayo Clinic, Rochester MN, Virginia Tech, Blacksburg VA, and the Joint Institute of Nuclear Research (JINR), Dubna Russia.

**Supervision and Consultation**

1. My Masters thesis was on the visualization of spectral CT data. While the responsibility of the visualization was passed on to Alexander Chernoglazov I remained active with the visualization team as a consultant. This included consultancy on my previous work, suggestions for future work, and also discussions relating to the import/export of reconstructed data for visualization and data analysis.
Acknowledgments

I would like to acknowledge my family and friends for their support and patience over the time of my thesis.

Also, my thanks go out to the MARS team, the various biological and pre-clinical research teams, and the external partners of the MARS group including the Mayo Clinic, Virigia Tech, and JINR. Their support, testing and feedback helped guide my thesis to completion. It is my sincere hope that the work I achieved has assisted each of them in achieving their various goals in using MARS technology for their research. Particular mentions go to Raja Aamir, Raja Panta, and Kishore Rajendran whose ongoing research served as the first beta testers of all my developed software. Thank you for your patience as the software stabilized into maturity.

Other mentions extend to Dr. Michael Walsh, Dr. Stephen Bell, Christopher Bateman, and Alexander Chernoglazov who helped build the complete MARS software suite from the scanner through to visualization. Your insights, and assistance has been invaluable in refining the design of my software and piecing together all the parts that make up the MARS system.

I would like to thank the University of Otago (Christchurch) and the HIT Lab NZ for hosting me during my thesis and providing the necessary resources. I would especially like to thank the Otago Scholarships committee for awarding me the Otago Doctoral Scholarship. Without it, this thesis would not have been possible.

Lastly, I’d to thank to Prof. Philip Butler, Assoc. Prof. Anthony Butler, Dr. Peter Renaud, and Dr. Nigel Anderson for their supervision and ongoing support. This thesis has been an invigorating journey. I look forward to the future developments in MARS technology and its eventual integration into clinical practice.
Chapter I

Introduction

My thesis describes the development of an image processing chain to reconstruct 3D volumes using data from the MARS molecular imaging system. This system is an undertaking to develop and commercialize spectral computed tomography (CT). Spectral CT has the potential for quantitative material discrimination which will improve the ability to diagnose a wide range of diseases.

This chapter first describes the thesis objectives and outcomes. It then proceeds with a basic overview of the emergence of spectral CT in medical imaging. This is followed by a brief introduction of the MARS molecular imaging system and the Medipix detector which forms the core of MARS technology. Then, a description of the structure of the MARS team is provided with its relevance to my work. Lastly, the research contributions in this thesis are stated followed by an outline of the chapters and their contents.

1.1 Research objectives and outcomes

The objective of this thesis is to develop an image processing chain for data acquired by the MARS system. This chain includes pre-reconstruction processing, reconstruction into 3D volumes, and post-reconstruction processing. Also of importance is the import and export of data from the scanning software which acquires the data, and from the visualization software which analyses the end results.

The aim is to create software which produces high quality images that exploit all of the information contained in the data provided. More importantly, this software should meet the immediate needs of the MARS team so that biological and pre-clinical research can be conducted with the MARS system.

This objective raised many research challenges to overcome. These are generalized through the following goals.
• The MARS prototype has a complicated design which means that many assumptions common to typical reconstruction geometries can no longer be made. The geometry of the MARS system needs to be properly modelled and integrated into the reconstruction solution.

• The Medipix detector is in a state of active development and evolution. The characteristics of the detector element response needs to be identified and appropriately dealt with.

• The volume of the data can easily exceed 20 GB in a single scan. This means that the software must support large volumes of data. Preferably, this should be achieved without relying on custom or expensive computing hardware.

• An end goal of the MARS project is to push spectral CT into clinical practice. Therefore, the software should match modern clinical standards.

• Spectral CT is a new modality. Therefore, algorithms should attempt to exploit features of the novel data.

• The MARS system is in a state of active development and evolution. The software should adjust to meet the ongoing needs of the system.

• The MARS project is made up of groups of development and clinical research teams. The software should identify and meet the needs of these user groups to support their research.

The results from these research challenges contributed to the creation of mPPC (MARS Preprocessing Chain) and mART (MARS Algebraic Reconstruction). These two applications perform the steps required to reconstruct datasets from the MARS system.

The software components are tied together using the DICOM standard. This is a clinical standard for packaging, storing, and transferring data. By adopting DICOM, the MARS system gains a streamlined work flow which is compatible with modern clinical imaging infrastructure.
To support DICOM, information object definitions (IODs) were designed to represent the spectral CT datasets. This was necessary as no standardized spectral CT IODs currently exist.

The developed IODs are based on a fusion of the existing CR (computed radiography), CT (computed tomography) and NM (nuclear medicine) IODs from the DICOM standard. The result is a data format which is transferable and viewable in any DICOM compliant software. In the future, the developed IODs may influence the creation of a fully standardized spectral CT IOD in DICOM.

1.2 The emergence of spectral CT

Medical imaging has become an indispensable tool in clinical practice. Starting from Wilhelm Roentgen’s iconic image of his wife’s hand \[1\], medical imaging has branched out over a large array of technologies, modalities, and possibilities.

A current technological shift is occurring with spectral CT. CT, made popular by Godfrey Hounsfield \[2\], is an x-ray based technology which acquires a set of 2D projection images around a subject. These images are then converted into a 3D volume by a mathematical process called reconstruction. Spectral CT acquires multiple CT datasets; each from a different part of the x-ray spectrum. A common name for these datasets are energy bins.

A good analogy for energy bins are the colour channels (red, green, and blue) in photography. Each of these channels are images acquired over unique portions of the visible light spectrum. In the same way, energy bins are different “colours” from the x-ray part of the electromagnetic spectrum.

The basics of every x-ray technology are the x-ray source, subject, and detector. A source produces a spectrum of x-rays which pass through a subject, some of which are absorbed. The remaining x-ray photons are then measured by a detector. For more details, the books by Bushberg et al. \[3\] and Novelline et al. \[4\] provide detailed descriptions of the technology.

Even back in the late 1970s, discussions existed about how to acquire data with different x-ray spectra in a single scan. For example, in 1976 Alvarez and Macovski introduced theoretical techniques for material decomposition through energy selective CT \[5\]. Later, the spectral CT trend started with dual CT, which acquires two energy bins in a single scan. Today, the material discriminating potential of
dual CT is commonly used in clinical practice over a variety of applications [6–10].

Technologies are emerging which can acquire three or more energy bins in a single scan. This means that even more information is available for discriminating between materials. The advantages include the potential to separate materials that were previously indistinguishable, and the potential for accurate quantitative measurements. Therefore, many believe spectral CT to be an important advancement in medical imaging [11–13].

To summarize, spectral CT is a natural extension of CT. The first instance of this was dual CT which acquires two energy bins. However, this is quickly expanding to three or more energy bins simultaneously. Fig. 1.1 illustrates the
differences between single energy, dual energy, and spectral CT with the MARS molecular imaging system.

1.3 MARS molecular imaging prototype

The MARS molecular imaging project is an undertaking to research, develop, and commercialize a molecular imaging system. The core technology is a spectral CT scanner which adopts the Medipix detector [14–16]. The ability to acquire up to 8 energy bins simultaneously at 55 μm² resolution allows for the possibility of material discrimination at the molecular level.

The term molecular imaging refers to acquiring data on molecular processes in a subject. In clinical use, this often relates to locating biomarkers. Biomarkers are materials (known molecules) which are known to be present in objects of interest e.g. tumors. As spectral CT offers the potential for quantitative material discrimination, it can be used for molecular imaging.

The current MARS prototype adopts a narrow cone beam CT design. A gantry, containing the x-ray source and a small Medipix based camera, rotates around a fixed axis. As the MARS camera is small, it must translate per revolution to cover the span of the subject. Inside the gantry, an independent subject mount can translate to cover the length of the subject. The maximum subject size extends up to 150 mm in diameter and 300 mm in length.

The scanning software provides a few modes which control the sampling procedure and the scan path. The gantry can rotate continuously or start and stop for finer angular precision. Also, the sample bed can continuously translate for helical scanning or be moved between revolutions for circular scanning.

To summarize, the MARS system is currently a prototype adopting a narrow cone beam CT design. For a more detailed description of CT designs, refer to the books by Bushberg et al. [3] and Novelline et al. [4]. Alternatively, refer to section 2.2 for more specifics on the MARS system design.

1.4 Medipix detectors for spectral CT

The Medipix family of detectors are particle counting detectors developed by CERN. The detector is made up of two components: a semiconductor sensor crys-
tal, and an application-specific integrated circuit (ASIC). The sensor crystal absorbs the particles and creates a corresponding electron-hole charge. The ASIC counts any detected charge impulses.

The semiconductor used for the sensor crystal determines what energy range of x-rays will be detected with high efficiency. In the context of spectral CT, we want the sensor crystal to absorb photons from the clinical x-ray range of the electromagnetic spectrum. Currently, silicon (Si), gallium arsenide (GaAs), cadmium telluride (CdTe), and cadmium zinc telluride (CdZnTe) are all appropriate materials which have been used by the MARS team.

The charge impulses are counted according to their position and strength. The ASIC is divided into $256 \times 256$ ASIC elements\(^1\) where each ASIC element has 2 counters. A counter acts if a charge impulse has a greater magnitude than a user-defined threshold. Therefore, each energy bin ranges from the set threshold to the maximum energy emitted by the x-ray source.

The ASIC is $14.08 \times 14.08 \text{ mm}^2$ and the ASIC elements are $55 \mu\text{m}^2$. To improve the energy discrimination, $2 \times 2$ sets of ASIC elements can be combined to produce detector elements with 8 counters yielding 8 energy bins. Therefore, the Medipix detector can switch between the fine pitch mode (detector elements of $55 \mu\text{m}^2$ and 2 energy bins) or colour mode (detector elements of $110 \mu\text{m}^2$ and 8 energy bins). Lastly, a charge summing mode combines data from a $3 \times 3$ array of detector elements (not ASIC elements) to better determine the origin of the photon.

To summarize, the Medipix detector is capable of acquiring up to 8 energy bins simultaneously. Various detector modes switch between the number of energies and the available element size. Advanced modes such as charge summing mode further improve the precision to maximize the quality of the data collected. Refer to section 2.1 for more details on the Medipix detector.

1.5 MARS molecular imaging team

The MARS molecular imaging project is made up of a set of teams over various fields and expertise. These teams can be split into two categories: development

---

\(^1\) It is important to distinguish between what I refer to as a ASIC element (a pixel as a physical hardware unit) and a detector element (a pixel as an element in the acquired image). A detector element may result from one or more ASIC elements depending on the active mode (spectroscopic or fine pitch mode).
teams and clinical teams.

The development teams work on the technology including the hardware of the MARS prototype, the integration of the Medipix technology, and the development of the software. The software teams include the scanning software team, the image processing team (led by me), the material decomposition team, and the visualization team.

Each of the development teams influence each other and meet regularly to update on progress and raise immediate needs that should be addressed. The important issues for my PhD are the import/export of data between the MARS scanning software, the image processing software (mPPC and mART), and the visualization software.

The clinical teams conduct biological and pre-clinical research using MARS technology. Some of the studies conducted include the discrimination of multiple contrast media [17, 18], the discrimination of soft-tissue, lipid, calcium, and other solids in human excised atheromatous plaques [19–22], and various other studies [23–25].

The clinical teams are the first users of the software developed in my thesis. They also play an active role in the evolution of the MARS molecular imaging project as they publish the applications for MARS technology. This is critical in promoting MARS for immediate research and development funding, but also for establishing a solid reputation by the time the MARS system is ready for commercialization. Therefore, an essential part of my thesis was to enable the clinical teams to perform their studies despite the experimental nature of the MARS prototypes.

The consequences of this constraint meant that I was limited in the changes that I could make. Consider an existing image processing chain which all team members are currently using. I then make a change which produces nicer looking images but doubles the processing time. Unless I can demonstrate to the clinical teams that they can now accomplish research that previously was impossible (e.g. detecting a significantly lower concentration of a contrast agent), they will not accept the change. This meant that I am limited in the choice of algorithm I can release to MARS team.

During the course of my PhD, a subset of the clinical teams came together to form a QA (image Quality and Assessment) team. The goal of this team was to
determine the image quality that could be achieved with the MARS system at any point and time. This meant that all changes I release would be further tested by the QA team. This lessens the testing requirements for all the development teams. In this thesis, you may note in chapter 7 that I only tested a single dataset. The QA team is the rationale behind that decision.

The QA process was set up by creating a library of standard phantom samples to scan. This includes various perspex shapes (such as a cylinder), vials of known solutions (to test the sensitivity of the scanner), and the library grows as more tests are needed. Once scanned, these samples would proceed through a set of statistical measurements including MTF (modulation transfer function), PSF (point spread function), signal to noise ratio, and true/false positive/negative once material decomposition routines became available. Therefore, even if my testing procedures are lacking given time constraints, all changes made are thoroughly tested one way or the other.

To summarize, the MARS molecular imaging project is made up of teams which either develop MARS technology or use MARS technology for biological and pre-clinical research.

During my PhD, I led the image processing development team. Close collaboration was needed with the other development teams so that together, a streamlined MARS system could be created.

The clinical teams are the first users of the software and the public face of the MARS molecular imaging project. This means that meeting their needs was of high priority throughout my thesis. Therefore, close collaboration was also maintained with all clinical teams.

1.6 Research Contributions

The contributions of my PhD can be split into three categories. The first is the underlying goal of developing a functional image processing chain for the MARS project. This in turn, allows the MARS project to develop spectral CT technology as a new medical imaging modality.

The second category is the collaborative contributions made with other teams within the MARS group. These contributions are very multi-disciplinary and range from collaborations to test, characterize, and cater to new detectors to collabora-
tions with specific pre-clinical and biological studies. In this thesis, the collaborative contributions are in the form of published works listed in the Academic Contributions section. The relevant published works are also mentioned at the start of the thesis chapters.

Lastly, the final category is the independent contributions conducted as part of developing the image processing software. Some of the more important research questions include the following.

- Chapter 3 - How can we deal with thermal sensitivity in Medipix 3.0?
- Chapter 4 - How should I model the MARS prototype geometry for projection stitching and reconstruction?
- Chapter 5 - How should DICOM data structures be implemented for MARS spectral CT datasets?
- Chapter 6 - What kind of signals do the various Medipix detector produce and how should they be dealt with?
- Chapter 6 - What is an appropriate algorithm for ring filtration?
- Chapter 7 - How can unintentionally truncated scans be salvaged for qualitative analysis?
- Chapter 8 - Are random sampling techniques effective for reconstruction?
- Chapter 9 - How can the quality and performance of basic iterative reconstruction techniques be improved?

There are many more research questions answered throughout my PhD. At the start of each main chapter, these are explicitly mentioned where relevant.

1.7 An outline of my thesis

My thesis is structured chronologically to match the outcomes with the evolution of both the MARS project and the Medipix detector\(^2\).

\(^2\)Note that the time periods of each work based chapter can overlap with each other.
An important component of my thesis was development alongside the MARS teams so that the various groups (development and pre-clinical research) could raise issues and benefit immediately from the implemented solutions. Therefore, it was important to keep apprised of any changes to any facet of the MARS project. This is reflected in a summary on the state of the MARS system at the beginning of every work based chapter. Each chapter then proceeds on to the main content. In this way the evolution of the MARS project is illustrated from my point of view over the time of my thesis.

- Chapter 2 provides more details on MARS technology. In particular, the relevance of MARS components to image quality are discussed.
- Chapter 3 describes the development of the initial pre-reconstruction processing software, MCTBuilder.
- Chapter 4 describes the development of the reconstruction software, mART, including the creation of a geometric model of the MARS system.
- Chapter 5 covers the adoption of the DICOM standard and the creation of DICOM definitions for MARS datasets.
- Chapter 6 illustrates the evolution of MCTBuilder into mPPC with the improvements made.
- Chapter 7 describes a study to salvage truncated datasets which are sometimes mistakenly acquired.
- Chapter 8 describes an investigation into using random sampling techniques for the reconstruction projection algorithm.
- Chapter 9 outlines steps performed towards automating the image processing chain.
- Chapter 10 concludes this thesis summarizing the work completed, my contributions, and a projection of future developments.
Lastly, Appendix A discusses the design of an automatic framework for image processing with the MARS system. This chapter did not have a research component. However, it was an important preparatory step towards a commercial grade image processing pipeline. Therefore, it is left as an appendix for interested readers.
Chapter II

An overview of the MARS system

This chapter outlines the technology of the MARS molecular imaging system. The primary focus will be on how the various components relate to the image processing chain, especially the challenges that the technology presents.

2.1 MARS camera technology

The core of the MARS molecular imaging system is the Medipix detector which provides the MARS camera technology. As introduced in section 1.4, the Medipix detector is made up of an ASIC and a sensor crystal.

The sensor is grown to be a single crystal which is then polished to a fixed thickness. The importance of the quality of the single crystal can not be understated. Any impurities, crystal boundaries or changes in density affect the how the charge impulses travel through the crystal. In the worst case, photon absorption events can be incorrectly registered in neighbouring elements thereby distorting the acquired signal.

The sensor crystal is attached to the ASIC via a process called “flip chip bonding”. The bonding is what defines the detector element as the charge impulse must travel through the bond to connect to the ASIC. Typically, there is only one active bond per detector element, even when the detector element represents a 2×2 set of ASIC elements.

Due to the small size (55 µm²), the bonding process has a high risk of malformations. These can distort incoming charge impulses or may even prevent the charge from reaching the ASIC. In such cases, the resulting detector element is considered to be dead. As an internal layer of the detector, the bonding can not be repaired. At most, the bonding process for the whole detector may be repeated with the hope of better results. Fig. 2.1 shows an example of a sensor crystal
which has both poor bonding in places and distortion in the crystal.

The sensor crystal material is selected such that targeted particles will be absorbed. As mentioned in section 1.4; Si, GaAs, CdTe, and CdZnTe are all appropriate materials for x-ray imaging. Fig. 2.2 illustrates this by showing the absorption profiles of the four materials with a thickness of 2 mm each.

Each of these materials face different challenges. Si, for example, is easy to manufacture and produces high quality sensor crystals. However, the absorption potential is poor above 30 keV (less than 50% efficiency for a 2 mm thick sensor) which makes Si impractical for human spectral CT (higher energies are needed to penetrate humans).

In contrast, GaAs, CdTe, and CdZnTe are each difficult to manufacture and bond. In addition, GaAs has a poor absorption potential above 94 keV. The materials with the best efficiency are CdTe and CdZnTe with high absorption potentials over the desired energy range from 10 keV to 120 keV (120 keV is the limit of the x-ray source in the MARS scanner). Clearly, if manufacturing were not an issue CdTe would be the material of choice.

The thickness of the sensor crystal can improve the detector efficiency. A sensor crystal which is too thin allows more photons to pass through undetected. This, in turn, requires higher exposure settings (exposure time and the tube current) to maintain a good number number of counts in the acquired image.
Figure 2.2: The absorption profiles of Si, GaAs, CdTe, CdZnTe between 20 keV and 120 keV with a thickness of 2 mm. Ideally, the profile should be 100% efficient over the whole range.

On the other hand, the exposure settings also have limitations. If the rate of incoming photons is too high, the ASIC cannot cope and starts to register multiple events as single high energy photons. This phenomenon is called pulse pileup. Spatially, the effects are minor as pulse pileup is a distortion of the energy signal. This can make the problem hard to detect. Refer to the article by Taguchi et al. [26] for more information on pulse pileup.

A sensor crystal can also be too thick. If the high voltage bias across the sensor crystal is insufficient to clear the electron-hole clouds, a build up of charge can occur. Also, a thick sensor crystal is sensitive to the angle of incidence of the incoming photons. The photons may pass through multiple detector elements before being absorbed. This means that there is a correlation between large angles of incidence and image blur.

Photon starvation occurs when there are insufficient photon counts to be statistically significant. The outcomes of photon starvation are bright streaks in the reconstructed image which occlude the volume underneath. The article by Yazdi
and Beaulieu [27] provides a comprehensive description of photon starvation and its effects along with other common CT image artefacts.

Lastly, there is the well documented charge sharing effect. This is when a photon is absorbed close to the boundary between elements. Here the photon may be registered to the correct element, a neighbouring element, or even both the correct and neighbouring elements. The result is a distortion of the energy signal. To counter this phenomenon, the recent Medipix detectors have a functional charge summing mode. Charge summing mode combines data from a 3×3 set of detector elements to improve the ability to determine the charge origin. The article by Doesburg et al. [28] details both charge sharing and the charge summing mode in the Medipix detector.

However, the coupling of sets of detector elements has a side effect where a bad detector element can corrupt its neighbours. Fig. 2.3 demonstrates this effect as
blocks of bad detector elements are more common then single bad detector elements. This complicates calibration processes for the detector. More importantly, this results in distinct error patterns in the acquired exposure images\(^1\). Ring artefacts, in particular, are often derived from multiple elements complicating both detection and elimination.

To summarize, the MARS camera is built using the Medipix detector. The various components including the ASIC, the sensor crystal, and the bonding between the two can cause various errors in the acquired data such as distortion, photon starvation, pulse pileup, charge sharing, and dead detector elements. In addition, some of the detector features such as charge summing mode are double edged swords which can further complicate the error patterns. All of these issues will need to be addressed in the image processing chain.

### 2.2 MARS gantry technology

Other than the MARS camera, the design of the physical scanner has important consequences for image processing. The scanner is designed for small specimens such as mice and uses a narrow cone beam design. The narrow cone beam design is perfect for a prototypical scanner as it saves costs on the construction of a large detector array. However, this means that there are more moving parts, and therefore, more sources of measurement error.

The basics of the scanner are two independent systems. The first system is for the subject. The subject rests flat, horizontally to the ground. The mounting platform can translate so that large subjects can be scanned with a single line of detectors across the subject. The components on the left of Fig. 2.4 are the subject mount system.

The other system is the gantry, whose operation and mount is independent of the subject. This is to minimize negative effects such as vibrations between the two systems. Unfortunately, splitting the two systems does open up the possibility of misalignment between the center of rotation of the gantry and the translation of the subject.

The gantry rotates on a single axis and contains two mounting points as shown

\(^1\) In this thesis, an exposure image is a snapshot from a MARS camera. A projection is the set of exposures that have a common angular position.
Figure 2.4: A view of the scanner design showing the subject mount, and the gantry which houses the camera and x-ray source mounts. The x-ray source mount and camera mount a both attached to the primary gantry. The gantry is independent of the subject mount to prevent dynamic influences between the two.

in Fig. 2.4.

The first mounting point is the x-ray source. The source can translate radially to control the source to object distance. Otherwise, the mount is fixed and can be used as a reference in a geometric model of the scanner. However, the ray from the x-ray source through the center of rotation may not be aligned with the radial translation of the x-ray source.

The second mounting point is for the MARS camera. The MARS camera can be translated both radially and tangentially to control the object to detector distance and to cover the span of the subject with the narrow cone beam. Ideally, the camera would be orthogonal to the x-ray beam that passes through the center of rotation. However, the camera mount and the translation paths all contain tilts and skews of the order of 1-3 pixels (0 to 300 µm).

Inside the MARS camera are the Medipix detectors which are mounted in a grid formation. The grid of detectors is not necessarily regular and may also be affected by tilts and skews.
To summarize, the narrow cone beam design is appropriate for a prototypical system with a small camera. However, the large number of mounting points and moving parts means that there is a high risk of geometric manufacturing errors. However, as long as the conditions are properly modelled and measured, this should not be an issue.

2.3 MARS scanning modes

There exist various modes which may be selected for operating the scanner. Previously, the fine pitch, colour, and charge summing modes have been introduced. In addition to these, are the various paths that the scan may take including start/stop or continuous scanning, and circular or helical scanning.

The start/stop scanning mode acquires data while the gantry is stationary. This means that the position is precisely known and can easily be synchronized between the various camera positions needed to cover the span of the subject. The gantry rotates through its assigned path before translating to the next camera position.

The synchronization relates to the angular positions of each exposure for all the camera positions. This is important because algorithms such as stitching require that all exposures have a common x-ray source position for cone beam CT. Also, all the detector position at each exposure in a projection would typically lie on the same plane.

Start/stop scanning is ideal when exposures need to be stitched together into projection images. The downsides of start/stop motion is that the gantry cannot move efficiently resulting in extended scanning times.

In contrast, continuous motion allows for rapid acquisition. Every exposure is acquired over an arc segment during rotation. This results in a slight motion blur depending on the exposure time and the speed of rotation. However, this motion means that it is more difficult to accurately synchronize the camera positions. This can impact the ability to stitch the exposures together.

Scans may either follow a circular or a helical trajectory. A circular path results in a basic scan geometry which is easy to model. However, the data acquired over the subject will not have uniform coverage due to the heel effect\(^2\) of the x-ray beam.

\(^2\) The x-ray tube emits photons in an irregular fashion. Plotting the photon counts vs angle results in a graph in the shape of a heel.
Also, some parts of the beam will overlap between subject positions. This means that the quality of the reconstructed volume will vary from place to place.

In contrast, a helical path provides uniform coverage of the beam over the subject as each portion of the beam will pass through the full length of the region of interest. Fig. 2.5 shows a comparison between circular and helical scan paths for a saggital view of the edge of a mouse. Notice how the circular scan produces artefacts where the circular sections overlap. In the image, these manifest as lines across the subject.

In addition, it provides benefits to reduce ring artefacts. Ring artefacts occur when static errors in the detector converge in a ring pattern. However, a helical path will not allow convergence as the errors are spread over a spiral. While the errors remain, their impact is lessened.

To summarize, the various scanning can move continuously or start and stop while traversing circular or helical trajectories. Each mode has its advantages, but the best quality typically results from a helical path with start/stop motion. Alternatively, the best speed is achieved with a helical path and continuous rotation.
2.4 Summary

The MARS molecular imaging system is made up of a variety of parts. These parts include the MARS camera which adopts the Medipix detector, the MARS scanner, and the scanning software which provides features using the MARS scanner.

Each part of the MARS molecular imaging system affects the quality of the data which is acquired. The effects range from the spectral measurements themselves, to the geometric characterization, to the scanning coverage and time. The image processing chain will need to be flexible to deal with the various challenges that each part of the MARS molecular imaging system poses.
Chapter III

MCTBuilder: An initial preprocessing system

This work described in this chapter started in August 2011 and it recounts the creation of a pre-reconstruction processing application called MCTBuilder. The goal was to create a single user-friendly application which performed the same tasks as existing pre-reconstruction processing scripts. MCTBuilder would then become available for use by the MARS team.

During this time I was involved with three publications. The first publication was a study of principal components analysis (PCA) as a tool for qualitative material decomposition [18]. While my direct contribution to the paper was the visualization from my Masters thesis, the outcomes of the paper motivated the creation of mPCA during the early stages of my PhD research. This application provided PCA as a tool to the wider MARS team.

The second publication was a presentation given at an annual scientific meeting at the Royal Australia and New Zealand College of Radiologists [29]. At this meeting, Assoc. Prof. Anthony Butler presented results on behalf of the team. This was at an early stage of my PhD while I still learning about the state and work flow of the MARS system. I helped prepare the data for the meeting and provided results from visualization.

A study was conducted by Dr. Paul Ronaldson on characterizing Si Medipix 3.0 detectors [30]. I was not directly involved in this study. However, the results motivated the decision to interpolate between sets of flatfield exposures in MCTBuilder. This was due to an observed temperature dependency in Medipix 3.0.

My third publication during this period was as part of a follow up study on the behaviour of CdTe Medipix detectors [31]. This study was conducted during the creation of mPPC and further motivated the change to interpolate between sets of flatfield exposures.

My role in this paper was to review the contents in preparation for publication.
As an observer throughout the study, it also enabled me to become familiar with the properties of raw MARS data.

Lastly, the outcome of this section of work was the application MCTBuilder. While the majority of this work involved porting python scripts to a unified application, I also had to address the problem of dealing with the thermal sensitivity of Medipix 3.0. This was accomplished through linear interpolation of sets of flatfield exposures.

3.1 Summary of the condition of the MARS system

This section provides an overview of the state of the MARS system at the start of my PhD. In this chapter, the relevant components include the state of the scanner hardware, the state of the scanner software, the state of the pre-reconstruction processing software, the state of the reconstruction software, the state of the material decomposition software, and the state of the visualization software.

3.1.1 MARS scanner hardware

The MARS prototypes were designated v3, and the one in the MARS lab numbered CT4. (CT1 is at the Mayo Clinic, CT2 at Virginia Tech, and CT3 at Charles University, Prague). The design of this series of prototypes is as described in section 2.2. An important feature of v3 was a twisted cable to transmit signals between the gantry and the internal computer. The later v4 design (and retrofitted to CT4) adopted a cable spool to enhance the life of the cables.

This was a simple design which allowed for rapid prototyping, but a consequence of using cables was that motion was limited to a single revolution of the scanner. This was not yet an issue as the scans only operated with start/stop motion. However, this opposed a transition to continuous motion as the lead in/out period (to accelerate the gantry) could not be achieved at high speeds.

With the 2009 prototype and start/stop motion, even small scans (720 exposures over 1 revolution) took hours to complete. Note that modern clinical CT scanners can acquire a full human scan in mere minutes (SOMATOM Force from Siemens advertises acquiring up to 737 mm/s).

The MARS camera had just been upgraded to accommodate the new Medipix 3.0 detector. Medipix version 3.0 boasted better energy resolution, the ability
to acquire 8 energy bins simultaneously, and a charge summing mode to counter charge sharing. See the paper by Walsh et al. for details on the preliminary tests with Medipix 3.0 [32].

The new detectors had Si sensor crystals which were high quality and were well bonded (less than 2% dead detector elements). In the case of Si quad detectors, the detectors were manufactured together in a regular grid with exactly 220 µm spacing.

While the feature list was impressive, Medipix 3.0 did not function reliably. Charge summing mode was not functional and Medipix 3.0 had instabilities due to heat [30,31]. This was especially problematic given the time taken for a single scan.

3.1.2 MARS scanning software

The MARS software was a collection of Python scripts which operated on a lower level C library called libMARS. The images produced by these scripts were numpy arrays.\(^1\) Alternatively, Matlab scripts could interact with Pixelman software [33]. The paper by Zainon et al. on the MARS scanner outlines the setup of the MARS system prior to this PhD research in more detail [14].

Some members of the MARS team had custom versions of the scanning scripts. These versions might name and group files differently or might offer unique scan protocols for their specific needs. The common set of scripts only offered circular scan paths with start/stop motion.

Dr. Michael Walsh took up the responsibility for creating a unified application for the scanning software. Refer to his thesis for more details [34].

3.1.3 MARS pre-reconstruction processing software

The pre-reconstruction software was also a collection of scripts. Some team members adopted Python scripts while others adopted Matlab scripts. Either way, the use of the scripts required basic knowledge of Python or Matlab. The presence of many versions in different languages made it difficult to know which scripts represented the best tools for a given scan.

\(^1\) Note that a numpy array is a Python specific format for storing sets of numbers. It is not an imaging format and can not be viewed in any standard image viewer.
The scripts selected for MCTBuilder were those developed by Dr. Paul Ronaldson. This choice was motivated by the fact that his scripts formed a complete pre-reconstruction processing chain. Also, the scripts’ performance was already validated due to the results Dr. Paul Ronaldson was obtaining towards his thesis [35].

3.1.4 MARS reconstruction software

The reconstruction software was Octopus CT by InCT systems [36, 37]. This is a commercial package for reconstruction which uses filtered back-projection with the Feldkamp-Davis-Kress approximation for cone beam CT [38]. Energy bins were reconstructed as independent CT datasets. This was effective but did require reconstructing $n$ times for $n$ energy bins.

3.1.5 MARS material decomposition software

There was no material decomposition software at this stage. However, previous work with MARS data used PCA for qualitative pseudo-material decomposition. The work by Butler et al. demonstrated the success of PCA in a feasibility study of distinguishing multiple contrast media in mice [18].

To support the use of PCA in the MARS team, I developed a mini application called mPCA. Extending the fast GPU code provided by Andrecut in his white paper [39], mPCA added a comprehensive graphical user interface (GUI). Through mPCA, qualitative attempts at material decomposition became possible at the early stage of my PhD.

In other cases, the members of the MARS team had their own independent material decomposition solutions. Dr. Paul Ronaldson, for example, developed his own algorithm using linear algebra as described in his paper on quantifying soft tissues [19]. However, this was not readily available to the MARS team.

3.1.6 MARS visualization software

The visualization software was called MARSCTExplorer. This was developed as part of my Masters thesis [40]. The software could combine volumetric datasets in real time where the combination was defined by a set of mathematical operators.
This gave MARSCTExplorer a lot of power when appropriate combinations were applied to spectral CT data.

### 3.2 Algorithms for the pre-reconstruction processing chain

The steps in pre-reconstruction processing exist to both correct and convert data into a suitable format for reconstruction. The reconstruction software, Octopus CT, requires the input to be structured as sinograms while representing the intensity ratio. This ratio is \( \frac{I}{I_0} \) from the Beer-Lambert law in Eqn. 3.1 where \( I \) and \( I_0 \) are the photon counts at the detector and the x-ray source respectively, and \( \mu \) and \( d \) are the linear attenuation and distance over the ray respectively. The Beer-Lambert Law is the basic model for x-ray absorption which assumes a monochromatic beam and a homogeneous medium.

\[
I = I_0 e^{-\mu d}
\]  

(3.1)

Octopus CT also provides the option to input raw data \( I \) and \( I_0 \) separately. However, this option was not explored. We wanted the freedom to experiment with algorithms that might better exploit spectral CT data.

Another requirement for Octopus CT is continuity. Each sinogram should contain all the data required from all exposures in the scan. This means that the exposures must be stitched together where appropriate and any dead regions should be filled in with an estimate.

To prepare MARS datasets for reconstruction with Octopus CT, the original scripts provided a chain of algorithms. These include darkfield masking, flatfield normalization, exposure stitching, dead detector element inpainting, and sinogram conversion. The pre-reconstruction processing chain is illustrated in Fig. 3.1.

#### 3.2.1 Darkfield masking

Darkfield masking is a measurement guided technique for identifying dead detector elements. Note that this is different from darkfield correction, which is an offset in a flatfield correction process. The paper by Zhang et al. provides a good description of the normal flatfield correction process [41].

A darkfield is an exposure acquired without an x-ray source. Depending on
Figure 3.1: The flow chart of the original preprocessing chain. The results of pre-computed steps are considered to be constants in the main chain. Therefore, darkfield masking and flatfield normalization are 1 input, 1 output tasks.

In this thesis, a dead detector element is one which does not respond to the presence of photons. This may manifest as a fixed number of counts or random counts. The darkfield should be able to pick up all cases except when the dead detector element is fixed at zero counts. However, those will be picked up in the...
flatfield exposures. See chapter 6 for more details.

Occasionally, a count may be registered due to an ionising particle from the environment, such as a cosmic ray or gamma ray from natural radioactivity. To account for this, a low threshold (such as 5 counts) is used to prevent the loss of valid data.

Also, a set of darkfield exposures is acquired so that the behaviour of each detector element is better quantified. In a pre-computed step, the darkfield exposures are combined by taking the maximum value per detector element. The resulting image can then act as a mask for the scan.

### 3.2.2 Flatfield normalization

Flatfield normalization converts the raw intensity values into the desired intensity ratio. It also balances the variations between detector elements and the variations over the x-ray beam.

A flatfield exposure is acquired while the subject is not present i.e. an air scan. We can represent acquired exposure images using the Beer-Lambert Law as in Eqn. 3.1 while adding a detector response function $D$. This is shown in Eqn. 3.2 for the case of a flatfield exposure where the linear attenuation is purely air $\mu_{\text{air}}$.

$$I_f = D I_0 e^{-\mu_{\text{air}} d}$$

The detector element response $D$ is dependent on the position of the detector element in the x-ray beam, the size and shape of the detector element, and the homogeneity of the sensor crystal. This is a complex function which would be difficult to characterize accurately.

If we assume that $\mu_{\text{air}}$ is zero valued, then $I_f \approx D I_0$. This is a useful result because it means that we can replace $D I_0$ in the scan with our approximation as shown in Eqn. 3.3.

---

2 As the size and shape of the effective detector element volume can be different due to sensor crystal inhomogeneity, the ideal exposures are noisy. This is because larger detector elements count more photons at the same energy.

3 This is very problematic as the response of inhomogeneous sensor crystals will be dependent on the angle of incidence of each photon.
\[
\frac{I}{DI_o} \approx \frac{I}{I_f}
\]  

(3.3)

As long as each flatfield measurement is taken at the same positions and similar conditions as the scan, it means that we do not need to characterize \( D \). Both the correction of \( D \) and the conversion to \( \frac{I}{I_0} \) is performed in a single step.

If the exposure settings are different between the scan and the flatfield exposures, scaling the average counts in \( I_f \) approximates the result. However, this is not ideal as the detector element response function \( D \) is also likely to be different. Procz et al. demonstrate in their paper how the conditions of the scan and flatfield exposures affect image quality [42].

To improve the approximation for \( DI_0 \) repeated flatfield exposures are acquired. These exposures can then be averaged in a pre-computed step to eliminate random noise from the signal.

A problem arose due to the instability of Medipix 3.0 [30, 31]. A temperature dependence resulted in an approximately linear drift over time. This means that \( D \) and \( I_0 \) no longer match between the scan and the flatfield exposures.

Luckily, for scans of moderate length, the behaviour of the drift was roughly linear. This meant that a simple solution was available.

Firstly, the number of flatfield exposures was kept at a minimum. This ensured that the state of the detector was constant while acquiring each set of flatfield exposures.

Secondly, multiple sets of flatfield exposures were acquired at different points during the scan (commonly just the start and end). The average flatfield image from each set could then be linearly interpolated to estimate the true state of the Medipix detector at the time of each scan exposure. For scans of moderate length (less than 3 hours), linear interpolation produced good results with minimal effort.

### 3.2.3 Exposure stitching

Stitching is a process which fuses multiple exposures into a single projection image. This process requires an exact geometric model to correctly position the exposures (and distort if need be). The stitching algorithm provided by the scripts emphasized simplicity and performance by making a few assumptions.
Firstly, it was assumed that each exposure shared the same x-ray source position. This matched the requirements of Octopus CT for cone beam geometry. As the early scans used start/stop motion, this assumption was guaranteed. The position of the camera and the gantry were precisely known allowing perfect synchronization.

The second assumption is that the detectors were axis aligned. In fact, positioning in the original scripts was done at the pixel level. The first set of Medipix 3.0 detectors were Si quad cameras where each of the four detectors were manufactured in a regular grid with exactly 220 \( \mu \text{m} \) spacing\(^4\). As long as the camera unit was carefully aligned during installation, this assumption held true.

The stitching algorithm is performed via two steps. The first step determines the canvas size of the projection image. Afterwards, the detector elements from each exposure are painted into the nearest pixel in the projection image.

### 3.2.4 Dead detector element inpainting

Inpainting is the process of filling in missing data in an image. In literature, the inpainting problem generally relates to photos where sections need to be removed and then filled in to rebuild a pleasant looking image \([43–46]\). In the papers listed, the results impressively replace large objects with a seemingly perfect image.

However, in the context of medical imaging, inpainting is a process to repair missing data with precise estimates. Filling in a dead detector element with false data can introduce significant errors into the reconstructed volume, even if the image appears correct. In the worst case, these errors could lead to false diagnosis.

The inpainting algorithm used by the scripts is a median filter over a 3\( \times \)3 neighbourhood to fill in the outer edge of dead regions. The inpainting requires at least 3 valid samples for the filter to proceed. The process is then repeated until all dead regions have been filled. For small dead regions (1-3 elements), the median is a reasonable approximation. However, the error quickly increases with the size of the dead regions.

On the other hand, even the best inpainting algorithms rely on neighbouring data. If a dead region is large enough to fully encase a feature of the subject, no

\(^4\) This is equivalent to 4 pixels with 55 \( \mu \text{m} \) detector elements, or 2 pixels with 110 \( \mu \text{m} \) detector elements.
inpainting algorithm can recover the missing data. It could potentially produce a pleasant looking image, but not a precise reconstruction.

A potential alternative would be a volumetric inpainting algorithm which includes data from exposures in other angles. However, depending on the location of the dead region, there may still be no available data for the inpainting process to function.

In this case, the simplest solution is to oversample the scan and cover the dead region with valid data from other exposures. Although this is not ideal, it offers a quick and reliable solution with minimal development time. Also, it is expected that manufacturing processes for detectors will improve. This should decrease the number of dead detector elements which, in turn, reduces the role of inpainting.

Lastly, there are other reconstruction algorithms that can handle the presence of dead detector elements unlike filtered back-projection. Given the ability to oversample, the potential for using better reconstruction algorithms, and the hope for better quality detectors, it was decided that the simple $3 \times 3$ median filter was sufficient at the time.

3.2.5 Sinogram conversion

Sinogram conversion is a simple change in coordinates. The stitched projection image uses the $x$ and $y$ coordinates. The angle between projection images adds a third coordinate $\theta$. A sinogram is an image using the $x$ and $\theta$ coordinates as illustrated in Fig. 3.2. For fan-beam CT, sinograms are convenient as they contain all the data needed to reconstruct a single slice from the volume.

3.2.6 Summary of algorithms

The algorithms forming the pre-reconstruction processing chain include darkfield masking, flatfield normalization, stitching, inpainting, and sinogram conversion. Each of these were pre-existing algorithms in the form of Python scripts. Fig. 3.3 shows the effects of each algorithm excluding sinogram conversion as it is just an alternative view of the same data.

An addition was made to the flatfield normalization code in response to the observations of temperature dependencies in Medipix 3.0. This change allows multiple sets of flatfield exposures to be linearly interpolated. The interpolation estimates
Figure 3.2: The three planes of projection data including the projection \((x,y)\), sinogram \((x,\theta)\), and another unnamed plane \((y,\theta)\). As projections are acquired at regular intervals around a subject, stacking the projections along the angle \(\theta\) results in an image with a sine wave-like appearance. The black parts of the image are bad pixels.

The true state of the MARS camera at the time of each exposure in the main scan.

3.3 MCTBuilder design

The focus when designing MCTBuilder was to present the necessary algorithms in a single user-friendly application. To this end there were a few required and desirable features as listed below.

- Allow the user to navigate and preview data in a clear and concise manner.
- Use a live preview to show the effects of the algorithms on the image.
Figure 3.3: The effects of the algorithms in the pre-reconstruction processing chain. These steps include darkfield masking (1), flatfield normalization (2), stitching (3), and inpainting (4). Dead detector elements are marked in red. In this example, only two exposures were stitched, and the darkfield masking and inpainting did not have much work to do.
• Allow batch processing of all the algorithms on all exposures in a scan.

• Rely on the scanner for geometric information instead of estimations.

The ideal pre-reconstruction processing chain is an automatic procedure. The scanner would know and maintain its state and pass on its state without any need for user intervention. However, at the time the MARS system did not know its own state as no geometric alignment or automated calibration procedures existed.

The appropriate solution was to provide user guidance for the image processing algorithms. To support this, the user needs to be able to navigate through the data, tweak the algorithms, and view the results in real time.

3.3.1 Navigation of data

Navigation of data contains two facets; how the data is presented and how the data is packaged.

The initial method for packaging the data was a single directory with names that contained metadata. “00001.3.0.2.0ff.npy” refers to the flatfield exposure at gantry rotation index 1, energy bin index 3, detector index 0 in the camera, camera translation index 2, and sample translation index 0. Locating files required searching a list which could easily contain 100000+ files. This is a tedious process.

Therefore, the scanner development team and I agreed to alter the packaging of the data by adopting a hierarchy of directories. The hierarchy was built up from the subject translation index, the camera translation index, the energy bin index, and the detector index in the camera. Lastly, the files in the end directory were labelled according to the gantry rotation index.

Conceptually, a directory hierarchy or filename metadata are equivalent. The advantage of the directory hierarchy is that it offers the files in a pre-sorted set of lists which allows users to locate data faster.

Relying on a directory hierarchy or on filename metadata is generally bad practice. This couples functionality of an independent piece of software together with the functionality and everyday use of the underlying operating system. Any user could disrupt the directory structure or rename a file.

However, the directory hierarchy was always intended to be a temporary solution. The true solution was to adopt the DICOM standard. This standard provides
protocols to package, store, and transfer data, and is used in clinical settings worldwide.

The DICOM standard is complex and took time to be fully implemented. The process involved designing new DICOM data structures as spectral CT data is not yet standardized. Afterwards, the new data structures were built into MARS software and the software was tied together with a picture archiving and communication system (PACS). This was completed at a later stage of my PhD as described in chapters 5 and 6.

As a temporary solution, the directory hierarchy offered easier navigation of data over the list of files with filename metadata. Also, the change was made to use TIFF images instead of numpy arrays to represent exposure images. This meant that the data could be viewed in any imaging tool. Together, easier searching and viewing were sufficient for the immediate needs of MCTBuilder and the MARS team.

3.3.2 Live preview of images in MCTBuilder

There are a few advantages to a live preview in image processing software. Firstly, it allows tweaking of parameters when user guidance is required. Secondly, a live preview allows for automatic testing and verification of the algorithms. Each user will test any changes made and can offer immediate feedback speeding up development and providing direction.

In MCTBuilder, the software workflow is divided into two parts. The first part is a wizard where each step of the wizard represents an algorithm in the image processing chain. The live preview updates the effects of current chain so that the wizard settings can be tweaked.

The second part is the batch process where the completed wizard settings are applied to a list of images. As no user interaction is required, the preview is not associated with batch processing.

Maintaining interactive speeds can be problematic for a chain of algorithms. Referring back to Fig. 3.1, the illustration codes each algorithm with a colour based on the required inputs and outputs. Note that the pre-computed steps are considered to be constants in the main chain.

Algorithms with a ratio of 1:1 do not affect each other. The only effect of such
algorithms is to add the time they take independently.

In contrast, algorithms with a ratio of $N:M$ represent potential choke points. Each of the $N$ inputs must complete the sub-chain of algorithms which quickly multiplies the overall time required for processing. In addition, for $M$ outputs, which do you choose to preview?

Luckily, in the case of stitching, the algorithm only produces a single output. Also, 30 is the largest number of stitched exposures in a MARS scan to date. As the exposures are all small images, 30 exposures can still be processed at interactive speeds (completed within a few seconds). However, it will not scale well for larger scans.

Sinogram conversion represents a critical choke point. In this case, the $N$ inputs and $M$ outputs are the complete dataset. Even for small scans, this process cannot be accomplished at interactive speeds, the bottleneck largely being the disk access.

On the other hand, a sinogram is simply an alternative view of the same data. Therefore, there is no necessity to include sinogram conversion in the live preview. This offers a nice solution where sinogram conversion is given as an option in the batch processing stage.

The other algorithms can still become bottlenecks if the procedure itself is slow. To this end I used OpenMP, a framework for multi-threading [47]. OpenMP provides a significant speed up by using all available computing power with almost no additional development time. The outcome is that for most MARS datasets, the live preview operated at interactive speeds.

3.3.3 Import/Export of data

The exchange of data between the scanning software and MCTBuilder was an important concern. Three changes were made including packaging the data in the directory hierarchy, packaging configuration files that characterize the scanner, and packaging a report describing the scan procedure.

The packaging of data into the hierarchical structure made the datasets compatible between the scanning software and MCTBuilder. This allows easy transfer of data by direct copy.

Each scanner has a set of configuration files which maintains the current state
of the scanner. By packaging these files with each scan, the state of the scanner at the time of the scan can be easily determined. Previously, the configuration files would need to be looked up on the scanner itself. This relies on the scanner strictly maintaining a complete history of all configuration files.

Before MCTBuilder, the scripts estimated the geometric position of each exposure by recalculating the path the scanner should have taken. The creation of a “scan report” means that the scanner directly informs MCTBuilder of the conditions of each exposure in the scan. Estimations were no longer required.

The scan report presents its data directly from the motors in steps. This means that there is a conversion process to obtain values in S.I. units such as millimeters and radians. To support this, the scan report contains a header which references the configuration files used during the scan.

It would have been possible to report the values in S.I. units directly. However, keeping the data as motor steps allowed the scan file to double as a diagnostic report. If any issues arise, the report provides data that can be directly verified on the scanner.

### 3.3.4 Graphical user interface

The design of MCTBuilder incorporated a simple GUI as shown in Fig. 3.4. The top left is a simplified Windows Explorer. The bottom left is a wizard containing the image processing algorithms. The right is the preview pane.

The explorer view has two parts. The left part shows the directory hierarchy while the right shows a list of files. This list includes all files in the selected directory as well as all sub-directories. This allows the list to show all files in a scan. The file list is also the input to the batch process.

The wizard offers each algorithm in the order that they are processed. Each algorithm has one or more parameters, the first being a toggle to activate the algorithm.

There are three wizard like tool chains. The first is the pre-reconstruction processing chain described above. The second was left empty for future filtrating and denoising algorithms. Lastly, the general chain offers a set of simple tools including pixel inversion, pixel scaling, resizing, and stretching. These are basic operations which are useful for general imaging but are not required for reconstruction.
Figure 3.4: A screenshot of MCTBuilder. The top left presents a windows explorer like view for navigation. The tools are presented as a wizard in the bottom left. The preview is shown on the right. Dead detector elements are marked as red.

A unique feature of the general chain is that the order is user defined. Each available tool can be added multiple times, reordered if need be, and removed.

The main view to the right shows a preview of the image. It also has the option to read the text of the configuration files, scan report files, and log files. The log files are created by MCTBuilder and record information from previous batch processing operations.

3.4 Summary

MCTBuilder unified the pre-reconstruction image processing chain. Algorithms from existing Python scripts were directly implemented into a single user friendly application.

To support the design of MCTBuilder the packaging of the scan data was altered so that exposure data was sorted into a directory hierarchy. The configuration data used during the scan was copied along with the exposure data. Lastly, the scan procedure was explicitly recorded in a scan report and packaged with the exposure data. Therefore, all information required to process the scan data was contained
in a single unit.

The design of MCTBuilder adopted a very basic layout. A windows explorer like interface presented the data for navigation. A wizard like interface presented the algorithms in the order of operations. A preview window shows the image with all the selected algorithms applied. In addition, the configuration and scan files could also be previewed.

With the completion of MCTBuilder, it became available to the MARS team. Although the algorithms were directly based on the original Python scripts, MCTBuilder contributed to the MARS project in three ways.

Firstly, a single pre-reconstruction processing application meant that the learning curve was much easier for users new to MARS technology. There was no longer any requirement to understand scripting languages to process data.

Secondly, the implementation of MCTBuilder, in conjunction with the scanning software, unified the way that data was packaged and transferred. This means that all scans can be accessed, viewed, and processed in a consistent way.

Lastly, the investigations into the behaviour of Medipix 3.0 led to the adoption of an interpolation scheme between sets of flatfield exposures. This improved image quality which was of benefit to all MARS teams.
Chapter IV

mART: An algebraic reconstruction application

This work described in this chapter started in December 2011 and it details the creation of mART, an application for reconstructing MARS datasets. The intention of mART was to replace Octopus CT while addressing issues directly related to the MARS system. The implementation of mART uses an algebraic reconstruction technique (ART). Initially, I chose the Kaczmarz solution and later switched to a simultaneous algebraic reconstruction technique (SART).

In order to complete the implementation of mART, I had to address the problem of modelling the geometry of the MARS prototypes. This model is made from simple rotations and translations to match the design of the scanner. With the completion of the geometric model, I was also able to improve the quality of the stitching algorithm in MCTBuilder (described in section 3.2) to sub-pixel precision.

As mART was completed, a study led by Raja Panta validated the value of mART to the MARS team [22]. His work on distinguishing biomarkers in human excised atheromatous plaques relied on data reconstructed through mART, not Octopus CT. I participated in the study by actively working with the image processing. This consisted of observing Raja’s work flow as well as providing support for any issues. The end results were the reconstructed images on which the study was based.

Alongside this, mART itself was published through IVCNZ 2012 [48]. This publication presented mART along with evidence that the intended goals for the first version had been obtained. These goals include; reconstructing all MARS datasets irrespective of the detector condition, and maintaining equal or better image quality than Octopus CT. The image quality assessment was conducted by Nate Tang, who also led the authorship of the paper.

Lastly, as with MCTBuilder, mART became available to the wider MARS team upon its completion. Around this time, the Mayo Clinic and Virginia Tech
University invested in MARS technology. This meant that MCTBuilder and mART were made available to them along with software support. The Mayo Clinic, in particular, had a different setup where files read from disk were not pre-sorted by the operating system. This required me to review and improve the GUI of MCTBuilder to correctly sort and display the data.

4.1 Summary of the condition of the MARS system

This section provides an overview of relevant changes to the MARS system as of December 2011. In this chapter, the relevant components include the state of the scanner hardware, the state of the scanner software, the state of the pre-reconstruction processing software, the state of the reconstruction software, the state of the material decomposition software, and the state of the visualization software.

4.1.1 MARS scanner hardware

By December 2011, scanners CT5 and CT6 had been completed. Their designs were similar to CT4 with the biggest change being a thicker lead shield in CT6. This meant that the x-ray source in CT6 could now operate up to 120 keV. CT4 and CT5 both had an operational limit of 80 keV.

Since the start of my PhD, a dual CdTe Medipix MXR had been used for testing along with the occasional study. A single scanner meant that only the best MARS camera was used (a Medipix 3.0 Si quad). With the completion of CT6, the CdTe detector was put into active use.

Unfortunately, the quality of the CdTe sensor crystal left much to be desired [31]. Bonding was poor and the crystal itself suffered from impurities and inhomogeneities. Fig. 2.1 in section 2.1 shows an exposure taken with the dual CdTe MXR. Notice the large dead regions, the wrinkle pattern, and also the large gap between detectors.

The previous multi-detector MARS cameras were perfectly aligned with a known gap of 220 µm. In contrast, the CdTe MXR had two detectors which were independently installed. Visual inspection showed that the detectors were skewed with respect to each other. Also the gap between the detectors was over 1.4 mm.
This means that the assumption of a well aligned system in the existing stitching algorithm was no longer valid.

### 4.1.2 MARS scanning software

The scanning software developed by Dr. Michael Walsh was on its first revision. It saved acquired exposures as TIFF images which were packaged in the format discussed in section 3.3. Together with MCTBuilder, the new image processing chain was now operational.

However, the manual transfer of data via external hard drive was an inconvenience. At first, a few team based external hard drives were available. As the team size grew over time this became unmanageable. At this point in time, members of the team have their own external hard drives. Still, this was a time consuming process which is neither efficient nor convenient.

### 4.1.3 MARS pre-reconstruction software

MCTBuilder was a welcome addition to the MARS system. However, the active use of the CdTe MRX camera meant that the algorithms were insufficient to produce high quality images.

The biggest issue was the size of the dead regions which exceeded the appropriate size of the inpainting algorithm. To compensate, scans were oversampled so that camera positions would overlap at least twice. This greatly increased scanning times.

The other issue was the alignment of the detectors in the new cameras. The assumption made in the stitching algorithm was that the alignment was sufficient to warrant stitching at the pixel level. This was not the case in the CdTe MXR.

In response to this, the stitching algorithm was updated to use the same geometric model of the MARS scanner that will be described in section 4.5. Also, the algorithm was switched to a sub-pixel resolution, brute force technique. This technique paints up to 200 points per detector element onto the projection image.

The choice of a brute force technique was to provide a quick implementation. An important goal of the upcoming mART was to eliminate the need for both stitching and inpainting. Therefore, as stitching would be used only for Octopus CT reconstructions in a legacy pre-reconstruction processing chain, the time penalty
of a brute force technique was deemed acceptable.

4.1.4 MARS reconstruction software

Octopus CT was still reconstructing data as before. However, the aforementioned issues about the size of the dead regions in the CdTe MXR caused a lot of problems. Octopus CT requires stitched projections that are fully inpainted. This forced users to either reconstruct data with inpainting related errors, or suffer increased scanning times through oversampling.

It became clear that the creation of a dedicated reconstruction algorithm for MARS datasets was now a high priority issue. A brief investigation into reconstruction techniques as well as consultation with my supervisor Dr. Peter Renaud revealed that algebraic solutions could ignore bad detector elements and still reconstruct a reasonable image.

4.1.5 MARS material decomposition software

Christopher Bateman started his PhD taking responsibility for the material decomposition software. At this point and time, the progress is limited to preliminary tests in Matlab. As the inputs to material decomposition are the reconstructed images, I maintained close collaboration with Christopher Bateman throughout my PhD.

4.1.6 MARS visualization software

Alexander Chernoglazov took on the responsibility to improve on the visualization capabilities of the MARS system. See his Masters thesis for some of his preliminary work [49]. The inputs to visualization are both reconstructed images and material decomposed images. Therefore, I also collaborated with Alexander Chernoglazov throughout my PhD. Lastly, as visualization was the topic of my Masters thesis, I made myself available as a consultant with the visualization team.

The new visualization software started as a branch of Exposure Render written by Dr. Thomas Kroes [50]. Exposure Render is an open source application which has an alternative rendering method using Monte-Carlo simulation. This approach provides realistic lighting from multiple sources while maintaining inter-
active speeds. The software code base is small and simple which makes it perfect for customization.

4.2 Motivation and goals

Having created a unified application for the pre-reconstruction processing algorithms, the workflow for the MARS team became easier. The users did not need any prior knowledge of programming in Python or Matlab and the live preview provided real time information on what each algorithm accomplished with their data. Together with Octopus CT, the users could produce reasonable images in a good time frame.

There are three main shortcomings with the current image processing chain. The first shortcoming is that the whole system treats energy bin data as independent CT datasets. Therefore, a lot of information is being discarded. We wanted to develop a reconstruction solution that better exploits the nature of the spectral CT data that the MARS system acquires.

The second shortcoming is the assumptions and requirements of Octopus CT. Octopus CT is designed for a CT system with a large detector array where each detector is of high quality. In contrast, the MARS prototype contains a small camera which is translated to cover the span of the subject. Also, the newer cameras had issues with large dead regions and poor alignment between detectors.

The final shortcoming was that multiple subject translations could not be reconstructed together. Octopus CT did provide complete reconstruction for helical paths. However, only one subject position could be reconstructed at a time for circular paths. At this time, the MARS system did not yet support helical scans. Therefore, long subjects had to be reconstructed in parts.

The immediate solution to all of these issues was to create an in-house reconstruction application. It is not expected that this application would be as fast or user friendly as Octopus CT. Instead there were a few general goals for the proposed software mART.

1. The algorithm must be customizable.

2. The algorithm should support all MARS datasets.
3. The software should be completed as soon as possible.

4. The quality should at least match Octopus CT.

As mentioned earlier, a mid term goal of the MARS project is to develop a reconstruction algorithm which better exploits MARS spectral CT data. Therefore, the reconstruction software should be customizable so that future developments can be easily implemented.

The new form of the MARS datasets can only be called fragmented. With large gaps between detectors and various skews, stitching and inpainting became more difficult. Therefore, an ideal reconstruction solution would reconstruct directly from the unstitched exposures in their respective positions and orientations.

In essence, I wanted to render the stitching, inpainting, and sinogram conversion tasks obsolete. This would also improve performance as the two primary bottlenecks in MCTBuilder would disappear.

As the new cameras were already in operation, it was important that the software be completed quickly. There are a variety of pre-clinical research teams that wished to use the high energy resolution of the CdTe MXR to further their research. Also, as mART will participate in active research, there should not be any compromise on image quality. Therefore, mART needs to at least match Octopus CT in quality and become available to the team as soon as possible.

### 4.3 A review of reconstruction algorithms

The first task in developing mART was to review reconstruction algorithms. I was already made aware of algebraic reconstruction techniques through my supervisors and former members of the MARS team. However, a more comprehensive review would allow for a more informed decision.

In the literature, reconstruction algorithms for CT come in a variety of flavours. These include back-projection \[38,51–53\], iterative methods \[54–56\], and optimization techniques \[57–60\]. Also, statistical models of the physics allows refinements to reconstruction algorithms \[61–65\].
4.3.1 Back-projection

The original model for reconstruction is the Radon transform introduced by Johann Radon [66]. This transform models the relationship between a set of projected rays (modelled as straight lines) and the function that the rays intersect.

The Radon transform has an inverse which allows for direct back-projection. However, the back-projection process results in blurred images with star shaped artefacts. These arise due to the spread of a single value over the whole ray.

In 1967, it was noticed by Bracewell et. al. that the blurring could be removed through convolution with a filter [51]. Originally, the filter was applied after back-projection as a 2D problem. However, this was a time consuming process.

In 1971, Peters and Bates published their work on the central slice theorem [52]. The primary observation was that the 1D Fourier transform of the projection corresponds to a line in the 2D Fourier transform of the slice.

This allowed the roles in filtered back-projection to be reversed. The convolution could operate on the 1D Fourier transform to filter the projection data. The filtered projection could then be back-projected to produce the final image. Computationally, this was much faster. I refer the reader to Kak and Slaney for a comprehensive overview of the early reconstruction techniques [67].

An alternative solution was also presented by the central slice theorem. Each line from the 1D Fourier transform can be interpolated together to build the 2D Fourier transform of the slice. Then the 2D inverse Fourier transform would recover the slice. However, as the number of projections is limited, large interpolation errors can result in poor reconstructions. De Francesco and Sliva cover some pure Fourier reconstruction techniques in their review [53].

The central slice theorem only applies to parallel beam geometry. With the rise of fan beam and cone beam, different approaches were needed. The central slice theorem could still be used for fan beam if the rays were re-binned to parallel beam geometry.

However, cone beam required more elaborate techniques such as the Feldkamp-Davis-Kress (FDK) approximation [38]. Variations of FDK dominate the cone beam filtered back-projection implementations today.

An ongoing issue with filtered back projection is the rigidity of the algorithm. Every new geometry requires a new approximation to allow filtered back-projection
to proceed. This is not a trivial process.

Also, filtered back-projection requires the sampling to match the Shannon Nyquist theorem [68] for accurate reconstructions. In modern CT, we wish to reduce dose further, and that means that we want fewer samples.

Lastly, one of the goals for mART is to remove the need for stitching. Instead, we want to reconstruct from discrete detector elements at arbitrary locations. There does not appear to be a simple method of achieving this with filtered back-projection.

4.3.2 Iterative techniques

Alongside analytical methods, iterative methods were also being developed. Originally, these were not as popular due to the high computational requirements. As computing power improved, algebraic methods became more favourable as the results were generally superior. Recently a study with radiologists confirmed that diagnosis was equal to, or improved with iterative reconstruction as opposed to filtered back-projection [69].

Iterative techniques are feedback loops. A current estimate is compared with measured data, and the error is fed back to create a better estimate. In the case of CT reconstructions, the estimate is of the volume which is then forward projected. The comparison occurs in the projection space and the error is then back projected onto the volume estimate. As long as the new estimate draws closer to the solution, convergence is guaranteed.

The first popular iterative technique for CT reconstruction is the Kaczmarz equation [54]. This equation performs the feedback loop on each ray independently in sequence. The new estimate is achieved by projecting the ray onto a solution hyperplane given by the measured data. Fig. 4.1 illustrates this concept with the simplest case of two solution hyperplanes (lines in 2D space) and a starting estimate.

The Kaczmarz equation guarantees convergence while being flexible with the scanner geometry. As long as the positions of the x-ray source and the detector element are known, the forward and back projection is a direct replication of the physical scan. This makes implementation easy.

An alternative to the Kaczmarz equation is the simultaneous iterative recon-
Figure 4.1: An illustration of the Kaczmarz process on the simple case of 2 solution hyperplanes in 2D space. The initial solution at the dot is repeatedly projected onto the hyperplane solutions until the common point is reached.

Construction technique (SIRT) proposed by Gilbert in 1972 [55]. SIRT considers all the rays in the system before updating the volume. This final update is a weighted average of the error from each ray which means that the volume is built up in a smooth, continuous fashion.

The downside of SIRT is the processing time. As all the rays must be processed for every iteration, the amount of work required increases dramatically.

A middle ground was found with the simultaneous algebraic reconstruction technique (SART) proposed by Anderson and Kak [56]. This time only a subset of all rays are considered. While the choice of subsets is arbitrary, it is common to consider all the rays that form a part of a single projection. The results presented by Anderson and Kak show that the performance is significantly better than SIRT while the resulting quality is comparable.

4.3.3 Optimization techniques

Optimization techniques are not solutions. They are alternative methods of approaching the problem. The solution may then be obtained via analytical or iterative means.

The basics of optimization techniques is to state an objective, consider con-
straints, and then build a model which represents the sum of the objective function under those constraints. Lastly, a method for solving the problem is identified.

Consider the equation \( A\vec{x} = \vec{b} \). We know that the measurements \( \vec{b} \) will have a noise component. Also, in the context of CT, \( A \) will probably be under-determined as we want to minimize dose.

Together, this means that we do not want to solve the equation. Instead we want a solution \( \vec{x} \) where \( A\vec{x} \) is close to \( \vec{b} \) under some conditions (positive, no noise, etc). A better objective would then be \( \| A\vec{x} - \vec{b} \|_2 \leq e \) where \( e \) is an error term which we consider an acceptable bound.

The advantage of optimization techniques is a reduction in dose. With filtered back-projection, accurate reconstruction requires sufficient samples based on the Shannon Nyquist theorem. However, the constraints posed in optimization techniques provide more details about the signal to be reconstructed. This means that fewer samples are required then what the Shannon Nyquist theorem suggests.

There are many constraints which are considered in practice for CT reconstruction. A popular constraint in modern reconstruction literature is the sparsity constraint used in compressed sensing techniques. While a CT image is generally not sparse, the total variation, wavelets, and the derivative, are all sparse representations of the volume.

Compressed sensing is a wide field with a vast range of approaches, and associated benefits. For brevity, these are not covered here. Instead, I refer the reader to the following articles which cover a range of implementations and uses of compressed sensing [57–60].

### 4.3.4 Statistical Approaches

There are no limits to the complexity of a statistical model. The article by Fessler provides a comprehensive review which covers a wide range of statistical approaches [65]. This includes using statistical denoising techniques to process data before or after reconstruction. More importantly, statistical techniques can be integrated into reconstruction algorithms.

Consider the previous example, where we wanted to find \( \vec{x} \) so that \( A\vec{x} \) is close to \( \vec{b} \). Now we can refine the idea of closeness. Instead of being close in magnitude, we can state the closeness in terms of likelihood. Which solution \( \vec{x} \) is most likely
to produce $\vec{b}$ given the relationship $A$?

Determining the likelihood requires knowledge of the physical system. For example, the x-ray source produces photons in a random process. This can be modelled as a Poisson distribution. Similarly, statistical models can be constructed for the detector response and the scatter. Altogether, the end result is a function describing the likelihood of acquiring the measured projection data given the current estimate of the volume.

Given a suitable statistical model, the general approach is to maximize the likelihood. Originally, maximum likelihood solutions were popular only for PET and SPECT data [61, 62]. However, in the past 10 years, solutions such as ordered subsets expectation maximization (OSEM) have become popular for CT reconstruction [63, 64]. When combined with optimization techniques, statistical models produce the most robust, clean reconstructions possible today [65).

4.3.5 Outcomes of the review

As can be seen in the previous sections, a wide range of reconstruction solutions exist today. The issue now, is to pick a method for development.

The ideal reconstruction algorithm would model the physics of the system accurately while having a low sensitivity to noise and outliers. In this case, the physics should be modelled statistically. Optimization techniques could then allow us to further impose noise limiting constraints. From the literature, such an algorithm should produce the best results.

However, my immediate task was not to develop the ideal reconstruction algorithm. Optimization and statistical techniques could take months if not years to develop. The process would involve characterizing the MARS system to properly complete the statistical model. This is a challenge by itself considering that the prototype is a changing system. New cameras and scanners have different properties which would change the statistics of an accurate model.

The mathematics needed to solve optimization techniques is also not trivial. Depending on the complexity of any objective function I develop, the solution may be difficult to find.

The MARS team needed to have reconstruction software as soon as possible which was well conditioned to MARS data. Current pre-clinical research teams in
the MARS project were using Octopus CT to reconstruct their data. Therefore, this was the baseline for the quality of the chosen solution in mART.

Therefore, my decision lay with the algebraic reconstruction techniques. I chose to first implement the Kaczmarz solution. This is a simple algorithm which could be quickly implemented. This made it perfect to validate the geometric model and projection algorithm that would be needed in the implementation.

Later, the first release of mART to the MARS team would adopt an implementation of SART. This was because SART has demonstrated in literature to produce good results in a reasonable time frame.

Also, the Kaczmarz solution and SART are similar algorithms. This means that the transition from one to the other is quick. Therefore, with Kaczmarz and SART, I could develop mART quickly to provide a reconstruction solution better than the current software Octopus CT. At a later stage, the option is always available to review the reconstruction algorithm and change to more elaborate solutions if needed.

4.4 The basics of algebraic reconstruction

The basics of algebraic reconstruction are not related to CT at all. Instead, it deals with the simple algebraic problem shown in Eqn. 4.1. Find \( \vec{x} \) given \( \vec{b} \) and the relationship \( A \). It so happens that with a few assumptions the CT reconstruction problem can be arranged in the same form.

\[
A\vec{x} = \vec{b}
\]  

(4.1)

In this case \( \vec{x} \) represents the values of the volume to be reconstructed. The volume represents linear attenuation, a measure of the absorption potential of a material. We treat each voxel in the volume as a member of the 1D array, \( \vec{x} \), irrespective of its position in the volume.

The measured values per detector element are the members of \( \vec{b} \). In this case, the measurements are converted into transmission values. Again, the order is arbitrary, although some solutions split \( \vec{b} \) into ordered subsets to improve performance.

Lastly, \( A \) represents the contribution of a set of voxels from \( \vec{x} \) to each value of \( \vec{b} \). In the context of CT data, the contribution is the average path length of the intersection of the ray through each voxel.
To derive the problem in the context of CT we start with the monochromatic Beer Lambert Law as shown in Eqn. 4.2. This is a good estimate of the number of photons which are absorbed by a material with linear attenuation \( \mu \) over a distance \( d \). \( C_0 \) represents the initial number of photons while \( C \) represents the remaining number of photons. In this form, the Beer Lambert Law assumes a homogeneous material and a monochromatic beam of photons.

\[
C = C_0 e^{-\mu d} \tag{4.2}
\]

In the case of the ray passing through numerous materials we can simply sum the effects of each material as in Eqn. 4.3 where \( i \) is the index for each material. For a volume divided up into discrete voxels, we can assume that each voxel represents a single material and sum over all the voxels that a ray passes through.

\[
C = C_0 e^{-\sum_i \mu_i d_i} \tag{4.3}
\]

The x-ray beam in a CT scan is not monochromatic. This could be modelled as shown in Eqn. 4.4 where \( C_{E_1-2} \) is the measured photons in the energy range from \( E_1 \) to \( E_2 \). Also, note that both the source photon count \( (C_0(E)) \) and the linear attenuation \( \mu(E) \) are energy dependent. The problem here is that now we have an infinite number of volumes, one for each value of \( E \).

\[
C_{E_1-2} = \int_{E_1}^{E_2} C_0(E) e^{-\sum_i \mu_i(E) d_i} dE \tag{4.4}
\]

To simplify the problem, we can break this up into a sum of discrete energy bins. For each energy bin we assume that \( C_0(E) \) and \( \mu_i(E) \) are constant over the spectral range of that bin.

Any inspection of linear attenuation curves would reveal that this is a bad approximation as the curves can have very steep gradients. However, by discretizing the problem, a solution is now possible. I started with the simplest form, a single energy bin over the range of the scan. This returns us back to the monochromatic form of the Beer Lambert Law.

To complete the derivation of the basic linear equation we look at the sum inside the exponential. We can define \( \bar{x}_i = \bar{\mu}_i, A_{ij} = \bar{d}_i, \) and \( \bar{b}_j = \sum_i \mu_i d_i \). Now the problem is structured in the linear algebra form from Eqn. 4.1.
4.4.1 Kaczmarz solution

The algebraic problem in Eqn. 4.1 rarely has an analytical solution. In CT, we usually want $A$ to be under-determined as we wish to limit the number of measurements required in $\vec{b}$ to minimize the x-ray dose. Under these conditions, there is an infinite number of solutions $\vec{x}$.

To find a solution Kaczmarz offered a simple, but effective iterative technique [54]. The technique is shown in Eqn. 4.5 where $\lambda$ is a relaxation factor, index $k$ is the current iteration, $i$ indexes each voxel in the volume $\vec{x}$ and $j$ indexes the pixels in the projection images $\vec{b}$. A current estimate of the volume is projected forward and compared with a measurement $(b_j - \vec{A}_j \vec{x})$. The resulting error is then projected back onto the contributing voxels equally. The magnitude of the back-projected value to each voxel is regulated by a relaxation factor $\lambda$.

$$x_{i,k+1} = x_{i,k} + \lambda \frac{b_j - \vec{A}_j \vec{x}}{\sum_i A_{i,j}^2} A_{i,j}$$  

As the process is repeated over and over for all measurements $b_j$ the volume slowly converges to a solution. In the presence of noise, the solution may tend to oscillate as various measurements pull the results away from the ideal. In this case, a small relaxation factor can lessen the effects of the noise by tightening the bounds. However, this comes at the cost of reducing the speed of convergence as each step contributes less to the solution.

4.4.2 SART

One of the big problems with the technique offered by Kaczmarz is that it tends to inject high frequency noise into the solution. As every ray is treated independently, after a single step is performed, the new volume is the same as before except for a single line. To the following steps, this line appears as an impulse in the volume. Fig. 4.2 illustrates this concept.

The problem is that errors in the form of impulses are almost transparent to any solver based on projections. Even if the errors were correctly identified, solvers such as the Kaczmarz solution divide the error equally over the whole ray. Therefore, the error will remain as a fixed structure during the reconstruction and manifest itself as salt and pepper noise.
Figure 4.2: An illustration of the first two steps of the Kaczmarz solution in two arbitrary directions. The blue ray is the result of the first step while the red is the result of the second step. The line profile of the red step on the right illustrates how the blue step appears as an impulse which remains in the solution after the red step has completed.

The solution offered by SIRT and SART improve on the Kaczmarz equation in this regard \cite{56}. Both SIRT and SART perform a weighted sum of the corrections resulting from a set of rays. In the case of SIRT, all available rays are considered while SART only considers a subset of the rays.

The driving factor is that all voxels in the volume should updated for each every step in the reconstruction. This lessons the introduction of foreign structures into the solution, in particular, high frequency impulses.

The equation for SIRT and SART is shown in Eqn. 4.6. Structurally, it is similar to the Kaczmarz solution except that multiple correction terms are combined as a weighted average using the values $A_{i,j}$ as the weights. This means that rays that pass through the center of a voxel will have a higher contribution then rays that pass through one of the corners.

$$x_{i,k+1} = x_{i,k} + \lambda \frac{\sum_j A_{i,j} b_j - \hat{A}_{j} \hat{F}}{\sum_j A_{i,j}}$$  \hspace{1cm} (4.6)

4.4.3 Summary of basics

The basics of ART is to solve Eqn. 4.1 in the context of the physics of CT. The chosen solvers for mART include the Kaczmarz and SART solutions. Both of these
are simple feedback loops where Kaczmarz considers one ray at a time while SART considers the weighted average of a set of rays. As SART is an average, the results are expected to be cleaner.

4.5 Geometric model of the MARS system

ART has very few requirements on geometry. To reconstruct accurate edges, ART requires the set of rays to sufficiently cover the angles tangential to the edges. The number of angles decides how well the curves of the edges can be recovered. Otherwise, ART has no restrictions on continuity, or on regularly spaced acquisition. As long as the x-ray source position and the detector element position are known relative to the volume, ART will function.

An accurate geometric model of the scanner is required to determine the x-ray source position and the detector element position. This section describes the model I developed for mART.

As mentioned in sections 2.2, there are a number of moving parts and mounts in the MARS system which need to be addressed. To simplify the process, only translations and rotations are considered. As these can be modelled using 4D matrices, the resulting geometric model is described as follows.

\[
\vec{p}_f = SGMCD\vec{p}
\]  
\[
\vec{x}_f = SG\vec{x}
\]

The final position of the detector element and the x-ray source is given by \( \vec{p}_f \) and \( \vec{x}_f \) respectively. \( S \) is the transform of the subject. \( G \) is the transform of the gantry. \( M \) is the transform of the camera motion. \( C \) is the transform of the camera mount. \( D \) is the transform of the detector mount in the camera. \( \vec{p} \) is the position of the center of a detector element relative to the center of the detector. \( \vec{x} \) is a point on the x-ray source relative to the x-ray source center.

The following subsections will address each of the transforms along with the assumptions made. The assumptions in the geometric model are made under two conditions. The first is whether or not it is possible for the installation to guarantee alignment. In this case, some parameters can be safely ignored.
The second condition is to simplify the model to a workable state. Although any geometric model can be easily adopted in ART, all parameters included need to be measured. It is impractical to measure some of these parameters physically, especially when components such as the detector are fragile and must not be touched.

This leaves measurement through observation of scanned data. However, the scanned data will be affected by the condition of all the parameters in the geometric model. The more complex the model is, the harder it will be to decouple the effects from the relevant parameters. Therefore, each parameter in the geometric model is present only if it can be proven to have a significant effect.

When designing the geometric model there are two coordinate systems to consider. The first is the scanner coordinates, while the other is the volume coordinates. Both coordinate systems use explicit terms to avoid confusion.

The scanner coordinates adopts the tangential, radial, and axial axis. These axis explicitly describe a cylindrical region without the complexity of the angular position. Fig. 4.3 shows the scanner coordinate system.

The volume coordinate system is also called the patient coordinate system. These use the coronal, sagittal, and axial axis. The coronal axes is defined as the going from the anterior (front) to posterior (back), the sagittal is from the left to right of the patient, the axial is from inferior (bottom) to superior (top).
4.5.1 Detector element transform

The detector element forms a part of a regular grid over the detector. It can also be assumed that the detector plane is flat. The aim of this section is to map a detector element index \((i,j)\) into a position in mm relative to the center of the detector.

In the scanner, the camera is mounted such that the rows of a detector are roughly in the direction of the axial axis of the subject. The columns are then in the direction that is tangential to the volume cylinder. Also, exposures are indexed from the top-left detector element. If we assign the axial axis to be the \(z\) axis, and the tangential axis to be the \(x\) axis then the position of a detector element is given in Eqn. 4.9.

\[
\vec{p} = \begin{bmatrix}
  s(j + \frac{(1 - r)}{2}) \\
  0 \\
  s(i + \frac{(1 - c)}{2}) \\
  1
\end{bmatrix}
\]  

(4.9)

Here, \(s\) is the span of a square detector element in mm, \(i\) and \(j\) index the columns and rows of the image respectively, and \(c\) and \(r\) are the number of columns and rows respectively. Also, note that a point in 3D is typically represented by a 4D vector where the 4th component has a value of 1. This matches up conveniently with the representation of rotations and translations as 4D matrices. Goldman’s paper on geometric representations thoroughly covers the affine space used here for the geometric model [70].

4.5.2 Detector transform

The detectors are mounted on a printed circuit board (PCB) which is mounted to the front plate of the camera frame. The goal of the detector transform is to account for any misalignments in the detector mount and to determine the position of the center of the detector relative to the center of the camera. The center of the camera is defined to be the pivot point of the camera mount. This is convenient when dealing with misalignments of the camera later.

The misalignment of the detector is modelled with two components. The first is
an in-plane skew while the other is an out-of-plane tilt. It is assumed that the out-
of-plane tilt is negligible due to careful installation of the detector. The in-plane skew is a rotation around the radial axis.

The position of the detector in the camera is modelled as a translation. The axial and tangential positions are relative to the center of the camera. However, the radial position is defined as the distance from the detector surface to the front plate of the camera frame. All three positions should be constant throughout the lifespan of the camera.

\[
D = \begin{bmatrix}
\cos(\theta_d) & 0 & \sin(\theta_d) & d_t \\
0 & 1 & 0 & d_r \\
-\sin(\theta_d) & 0 & \cos(\theta_d) & d_a \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\] (4.10)

When combining the in-plane skew and the detector position the result is as shown in Eqn. 4.10. The components \(d_t, d_r, d_a\) are the tangential, radial, and axial positions respectively while \(\theta_d\) is the in-plane skew. The model applies the skew first and then positions the detector.

### 4.5.3 Camera mount transform

The cameras are mounted directly to the gantry system. Ideally, the installation would line up the camera with the mounting plate. Also, the mount can translate tangentially across the span of the subject. The resulting transform is shown in Eqn. 4.11.

\[
C = \begin{bmatrix}
\cos(\theta_c) & 0 & \sin(\theta_c) & c_t \\
0 & 1 & 0 & c_r \\
-\sin(\theta_c) & 0 & \cos(\theta_c) & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\] (4.11)

Here, \(c_t\) is the translation of the mounting plate in the tangential direction, \(c_r\) is the distance from the front surface of the camera frame back to the surface of the mounting plate. Lastly, \(\theta_c\) is the in-plane skew of the camera relative to the mounting plate. The model applies the in-plane skew first and then positions the camera relative to the center of rotation.
The tangential translation is centered in line with the center of rotation. Therefore, in a perfectly aligned system with a single detector camera, the ray through the center of rotation should intersect with the center of the detector.

The second part of the camera mount transform takes into account the flaws of the gantry as well as the radial motion of the mounting plate. The tangential motion of the mounting plate may also be skewed. Also, the axial position of the mounting plate may not align with the x-ray source position. These transforms are combined as in Eqn. 4.12.

\[
M = \begin{bmatrix}
\cos(\theta_m) & 0 & \sin(\theta_m) & 0 \\
0 & 1 & 0 & m_r \\
-\sin(\theta_m) & 0 & \cos(\theta_m) & m_a \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (4.12)

Here, \(m_r\) is the distance from the center of rotation to the surface of the mounting plate, \(m_a\) is the axial offset between the mounting plate and the x-ray source position. Lastly, \(\theta_m\) is the in-plane skew of the tangential motion of the camera mount. The model applies the in-plane skew first and then positions the camera mount relative to the center of rotation.

So far, all of the out-of-plane tilts of the detector, camera, and the camera mount have been assumed to be negligible. This means that a simplification is in order. If we sum \(m_r\), \(c_r\), and \(d_r\) we get the object to detector distance. We could replace one of the three components with the object to detector distance and set the other two to be zero.

On the other hand, it would still be useful to store the original three values. The property \(m_r\) is a constant which is set for each scan. In contrast, \(c_r\) and \(d_r\) are properties of the camera and should not change over the lifetime of the camera. Therefore, installing a new camera should only require a lookup of the new values instead of new measurements.

### 4.5.4 X-ray source position

The x-ray source transform manages the radial translation of the x-ray source (source to object position). The radial axis of the x-ray source is defined to be parallel to the radial axis of the camera. However, the x-ray source may have a
tangential offset which means that the x-ray source is not in line with the center of rotation. Eqn. 4.13 shows the model when the x-ray source is treated as a point source.

\[
\vec{x} = \begin{bmatrix} x_t \\ x_r \\ 0 \\ 1 \end{bmatrix}
\]

(4.13)

The components \(x_t\) and \(x_r\) are the tangential and radial positions of the x-ray source respectively.

4.5.5 Gantry transform

The gantry transform accounts for the rotation of the gantry. It is also possible for the center of rotation to be misaligned with respect to the volume axis. However, such a tilt is extremely difficult to measure. For simplicity it is assumed to be negligible. The resulting transform is shown in Eqn. 4.14 where \(\theta_g\) is the gantry rotation.

\[
G = \begin{bmatrix} \cos(\theta_g) & \sin(\theta_g) & 0 & 0 \\ -\sin(\theta_g) & \cos(\theta_g) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

(4.14)

4.5.6 Subject transform

The subject transform aligns the subject with the volume space to be reconstructed. Physically, the subject can be translated axially. Virtually, it is useful to further translate the subject so that the object is centered in the volume space. This allows the volume space to be shrunk as small as possible saving memory and speeding up the reconstruction. Either way, the volume space axis is defined to match the axis of the subject. Therefore, there is no skews or tilts, only a single translation. Eqn. 4.15 shows the subject transform.
\[
S = \begin{bmatrix}
1 & 0 & 0 & s_s \\
0 & 1 & 0 & s_c \\
0 & 0 & 1 & s_a \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (4.15)

The components \(s_s\), \(s_c\), and \(s_a\) represent the subject position. Note that we do not work with the tangential and radial axis anymore. In the volume space, the axis are the sagittal, coronal, and axial axis.

### 4.6 Sampling techniques for algebraic reconstruction

The second part of the geometric problem of algebraic reconstruction is determining the relationship between the voxels and the detector elements. Using the geometric model of the scanner we know the location of the source and detector element relative to the volume. This section presents the design decisions in choosing a routine for projection.

There are two common approaches to performing the projection. The first are ray driven techniques. These follow a ray from the source to a detector element. The second are voxel driven techniques. In contrast, these consider all rays that pass through a single voxel from the source.

#### 4.6.1 Ray driven techniques

As mentioned above, ray driven techniques focus on a single ray at a time and follow through all voxels between the source and a detector element. There are two common ray driven techniques that are used in practice. These are raytracing and raycasting.

Ray driven techniques are usually the most important for forward projection when considering parallelization. Forward projection writes to the detector elements after reading the volume voxels. In a parallel process, reading the same location is safe but writing is not. Ray driven techniques treat the detector elements independently and are therefore the best option for a parallel implementation of forward projection.

In contrast, back-projection is the opposite problem. A voxel is updated using the errors from numerous detector elements. Therefore, an appropriate process
would only read the detector elements and write to each voxel independently. This
is not the case with ray driven techniques.

Still, considering parallelization is premature at this stage. To start with, a sin-
gle technique will be implemented for both forward and back projection in mART. For more
details on the challenges of parallelization, see the work by Nguyen et al. [71].

Raytracing

Raytracing is an exact approach to projection first used in radiology by Siddon [72]. It treats each voxel as a discrete, homogeneous unit and calculates the exact
path length of the ray through the voxel.

At first, this could be considered a slow technique. However, some clever tricks
by Bresenham allow fast navigation through a regular grid [73, 74]. The basic
technique is to determine the length of the ray to cross grid lines in each axial
direction $x$, $y$, and $z$. Then as you step through the voxels you can easily determine
which grid line is hit next in the 3 directions. The process is outlined below.

1. Calculate the length $\vec{s}$ needed for the ray to cross grid lines in each axis.
2. Calculate the distance $\vec{i}$ to the first grid line in each axis.
3. Initialize the current distance $p = 0$ and the potential next points $\vec{n} = \vec{i}$.
4. Find the index $e$ of the nearest next point i.e. the minimum value in $\vec{n}$.
5. Process the current voxel with the calculated path length $A_{i,j} = n_e - p$.
6. Update the current distance $p = n_e$ and the next point $n_e = n_e + s_e$.
7. Repeat process from step 4 until the end of the ray is reached.

In this process, the distance to the next grid lines are maintained along with
the current distance travelled using basic addition and comparisons. This allows
the algorithm to step through the relevant voxels in sequence.

While this process is simple, the fact that it must be repeated for all voxels
contained in all rays can not be ignored. Depending on the size of the volume, this
process can easily become the bottleneck in the algorithm.
Raycasting

As opposed to raytracing, raycasting treats the volume as a set of discrete points. When the ray proceeds through the volume, it does so at regular intervals and samples the volume with some form of interpolation. The interpolation scheme controls the level of precision of the process. Nearest neighbour interpolation will naturally be inferior to trilinear interpolation although the latter will be much more expensive.

The process of stepping through the volume with a constant sampling distance means that raycasting is simpler than raytracing. However, the interpolation process can introduce errors.

4.6.2 Voxel driven techniques

Voxel driven techniques are about considering the effect of each voxel independently. The effects of a single voxel extends to multiple detector elements. Therefore, voxel driven techniques naturally fit the back projection process, especially if parallelization is desired.

However, projecting a 3D voxel onto a 2D projection image is not a simple process. Although a precise projection could be achieved and the contribution of the voxel to each detector element accurately determined, the process is much slower than tracing or casting a ray. Also, it must then be repeated for all voxels. This repetition can easily cripple performance.

Splatting

Therefore, in practice, many implementations use approximations such as splatting. Splatting takes a simple geometry and projects it on to a 2D projection image under a variety of conditions to create a lookup table. Then, the lookup table provides a reasonable estimate of each voxel projection by looking at the splatted result under similar conditions.

Issues with goals of mART

During my investigation into voxel driven techniques, an unforeseen issue emerged with MARS datasets. One of the driving motivators for mART was the elimina-
tion of the stitching and the inpainting routines. This meant that mART would reconstruct directly from the individual normalized exposures.

However, with a technique that starts from the voxel there is no prior information about which exposure in a projection the voxel will influence. This means that all exposures must be tested for each voxel which introduces a significant delay. If the exposures were stitched into a single projection this would not be an issue. However, this would defeat the goal of making stitching obsolete.

Therefore, voxel driven techniques will not work well with the intended form of mART. Still, if GPU parallelization were to be investigated, voxel driven techniques might still offer the best solution for back-projection under these circumstances.

4.6.3 Implementation

When considering which sampling technique to implement in mART, I based the decision on two requirements. The first was precision, while the second was performance. For the first implementation, precision takes precedence over performance.

There are many sources of errors in the MARS system. These can include issues in the sensor crystal, in the bonding between the sensor crystal and the ASIC, the circuitry of the ASIC, the condition of the x-ray source, the circuitry of the gantry, the physical condition of the gantry (vibrations and impacts), the geometric measurements, etc. I did not want to introduce more potential errors through the reconstruction process.

To this end the raytracing technique seemed the most appropriate. There is no interpolation or approximations. Instead the path length is accurately determined. Also, with the Bresenham technique, raytracing is almost as fast as raycasting. Therefore, it would serve as a good scheme for the first version of mART. Later, the projection scheme could be reconsidered if necessary.

4.6.4 Summary of sampling techniques

The problem of projecting the volume on to the exposures and vice versa is a problem of navigation and sampling. The three commonly implemented techniques, raytracing, raycasting, and splatting each offer their own advantages. Raytracing accurately measures the path length of a ray through a voxel. Raycasting uses regular sampling to quickly navigate over a volume. Splatting, as a voxel driven
technique, offers the potential for good parallelization with back projection.

In the case of MARS data, voxel driven techniques are crippled due to the fact that we no longer wish to stitch exposures together into single projections. But the deciding factor for the first implementation of mART was the accuracy offered by raytracing. As a simple, precise technique, it is hoped that at least the projection routine can be ruled out as the cause of any issues that are observed in reconstructed data from mART.

### 4.7 Implementing mART

With all of the design decisions made, the implementation of mART was a relatively straightforward endeavour. This section covers the first two major versions. The first version was an implementation using the Kaczmarz solution. The second version adopted SART. In both cases, the GUI, the geometric model, and the projection routine were identical.

#### 4.7.1 mART version 1

The goal of a first version is generally proof of concept. Every aspect of a piece of software needs to come together and function correctly. In the case of mART, I knew which algorithms to use, and I knew that mART should reconstruct from unstitched, normalized exposures without inpainting.

At the early stages, another nice feature came to light. As Kaczmarz treats every detector element independently, there is no dependency on how the detector elements are packaged. This meant that unlike Octopus CT, mART has no requirement for the input to be sinograms.

Recalling the pre-reconstruction processing chain from Fig. 3.1, this is a very nice result. Both stitching, and sinogram conversion are the biggest bottlenecks in the chain of algorithms. So with the dependency on them gone, MCTBuilder can be simplified together with a solid improvement in performance.

**Graphical User Interface**

Altogether, the first version of mART looked as shown in Fig. 4.4. The GUI offers a modest interface with a few settings on the left, and a preview on the right.
The parameters on the left hand side represent errors in the geometric variables. The intent is that they should never be used as geometric errors should be updated on the scanner so that future scans are not affected.

In the case that these parameters are wrong the GUI offers corrections to them as offsets. The MARS scanner is different from a commercial design in that every exposure can be uniquely placed. If desired, the tangential and radial position can change for each exposure in a scan. While such a scan would likely be more trouble then its worth, I felt that it was important that mART be capable of handling any possible dataset from the MARS system.

Therefore, in such an odd scan, the source to object and object to detector distances would be unique to every acquired exposure along with all other parameters. However, the error in any parameter would be the same for all exposures. Therefore, the GUI presents the measurement error and not the absolute values.

The measures presented are directly from the geometric model and include those which are most commonly wrong. These include the object to detector distances.
distance (sum of detector, camera, and camera mount radial positions), the source
to object distance (x-ray source radial position), the circumferential offset (the
term I adopted before tangential, it is more explicit but also a mouthful), and the
skews of the camera, and the camera mount.

The other settings control aspects of the reconstruction. The voxel scale was a
multiplier on the size of the voxel. 2x would mean that the voxel would be twice
the size of the default value. The number of threads controls the threads available
to openmp. In modern systems, the maximum is typically 8 threads.

OpenMP was a fast method of improving performance in mART. It means that
up to 8 rays are traced simultaneously. As these rays are spread apart there is no
risk of synchronization issues regarding reading and writing to the same location.
Therefore, it was a simple, safe method of improving a CPU implementation.

Lastly, a basic summary of the reconstruction would be shown including the
directory paths, the volume dimensions, and the volume size.

The preview on the right shows the progress of the reconstruction. The updates
to the image occur every time all rays in a projection have been processed. At the
bottom, a different slice may be requested. Also, the option to request a different
energy is provided, but the first version of mART only reconstructed energy bins
independently in sequence. Lastly, some simple window/level tools allowed for
better viewing of the reconstructed estimate.

**Characterizing the volume**

Behind the interface are some other important components of mART. The first
being the procedure to estimate the volume dimensions. At the time, the scanner
did not pass on the details of the region of interest that it was scanning. Therefore,
I had to extract a relevant region from the geometric positions of each exposure in
the scan.

There were four important measurements which needed to be calculated. These
include the voxel size, the volume length, the volume radius, and the volume offset.
The volume offset moves the volume position into relative coordinates. The subject
position is given in motor steps relative to an arbitrary datum. For reconstruction,
we want the origin to be at the bottom of the volume. Therefore, the subject
position needs an offset per exposure.
Figure 4.5: The model for calculating the radius. The algorithm uses two right angled triangles to calculate the radius.

During this calculation the basic process was to position the x-ray source and the exposure using the geometric model. Then four rays would be cast from each corner of the exposure. From these rays a bounding box around the exposures could be generated for each projection. This bounding box then gives the final measurements.

The first measurement is the radius of the volume. This situation is shown in Fig. 4.5. A quick observation will note that this setup is simply two right angled triangles. Therefore, the calculation becomes as shown in Eqn. 4.16. Here, $r$ is the desired radius, SOD is the source to object distance, $p$ is the tangential position of the exposure, and SDD is the source to detector distance.

$$r = \text{SOD} \times \sin \left( \arctan \left( \frac{p}{\text{SDD}} \right) \right) \quad (4.16)$$

The calculation for the volume length has three components. The first is the axial coverage of the scan, the second is the cone at the top of the volume, the third is the cone at the bottom of the volume. Fig. 4.6 illustrates the model used.

In this case the calculations operate from the back of the volume which means that the radius must already be known. From there, basic triangle geometry will
Figure 4.6: The model for calculating the length. The algorithm calculates three components using the back of the volume. Therefore, the radius must be already known.

give the sizes of each of the three components. Lastly, the axial offset can also be calculated here. As mentioned earlier we want the origin to be at the bottom of the volume so the offset will be the axial position of the lowest x-ray source position minus the bottom section of the volume.

The final calculation is to estimate the size of the voxels. The choice of voxel size is mostly arbitrary. However, an issue arises with the fact that the projection routine treats rays as lines. If the voxel size is too small, then the rays may skip over voxels. This leads to large gaps in the volume as shown in Fig. 4.7.

To guarantee that this is not possible, some simple observations can help. Firstly, the ray spread will be furthest at the back of the volume. Also, the ray spread directly matches the detector element spacing. Therefore, magnifying the
detector element spacing onto the back of the volume guarantees that no voxel will be skipped through the projection routine.

**Weighted distance scheme**

A lot of literature on algebraic reconstruction has a focus on how to improve the rate of convergence. A big section of this work relates to the order of operations. The basic principles being that the projection between two orthogonal hyperplanes will bring the result closer to the optimum then hyperplanes which are parallel (in this case the solution does not change). So it is beneficial to consider processing projections in an order that favors orthogonality.

However, as Klaus Mueller rightly points out in his thesis [75], a pure orthogonal approach is also not a sufficient criteria. It is also desirable to spread out the
angles so that the reconstruction is not biased towards particular angles.

The problem here is related to the way that algebraic reconstruction reaches convergence. In every step, the routine looks for the solution which is closest to the current form of the volume. At the start of the volume, the change needed to reach a solution is large. As the reconstruction proceeds the change decreases at every step.

This means that the influence a projection has on the volume will be larger at the start of the reconstruction then at the end. Therefore, if the angles are clustered together, the reconstruction will heavily favour the clusters at the start of the reconstruction. Therefore, the criteria of spread is an attempt to balance out the influence of each projection which contributes to the reconstruction.

To this end Klaus Mueller introduced his weighted distance scheme. This scheme considers two components including orthogonality and the distance to the previous selections. What this means is that the system will try to spread out the angles while still favouring orthogonal projections to improve convergence. He describes his scheme along with comparisons to other common ordering schemes including sequential access, random selection, and fixed angle selections. His results clearly show the improvement that the weighted distance scheme provides.

Therefore, mART adopted the weighted distance scheme in choosing the order in which projection angles are selected. It is a very simple tactic that offers a good improvement in quality and the overall convergence rate.

**Testing mART version 1**

As a first version, the testing phase of mART version 1 was mainly to ensure that the volumes were reconstructed correctly. Performance and quality were premature at this stage.

The first tests were done with a Shepp Logan phantom. The projection data was arranged in the same manner as a MARS dataset from a perfectly aligned camera. There were 360 projections with one exposure per projection. Also, a slight blur was applied to the phantom to give the edges a more natural curve and better condition the problem. This represented the simplest possible case for a MARS dataset.

Fig. 4.8 shows a single slice from the original phantom compared with the
reconstruction using mART. As can be seen the volume structures were correctly recovered.

For verification of real data, an old dataset called FatCaFe was reconstructed using mART. This was the dataset from Dr. Paul Ronaldson’s tests on material decomposition in his paper [19]. The results are shown in Fig. 4.9.

Once again, the structures of the dataset were correctly reconstructed which validates the behaviour of mART. This means that the next version can focus on improving the quality of the reconstructions.

4.7.2 mART version 2

From the earlier investigations into reconstruction techniques, it was always intended to complete a SART implementation. All the literature on ART agrees that the quality of SART was superior. Now that the geometric model and the projection routines had been verified it was time to complete the true implementation of mART.
Figure 4.9: The second test was of real data on a phantom scan called FatCaFe. The vials contain solutions of water, lipid, calcium, and iron.

*Combining energy bins*

Before implementing SART, another observation warranted immediate attention. The first version reconstructed all the energy bins in sequence. This meant that for 8 energy bins, 8 reconstructions needed to be performed.

However, the energy bins are acquired simultaneously. This means that the geometric position and orientation of the exposures for each energy bin are identical. In that case, both the geometric model and the projection routine can be shared with all the energy bins.

Therefore, a quick optimization was to reconstruct all the energy bins simultaneously. Each voxel in the volume would represent a vector containing all the energy bins. Therefore, when the projection routine determines a path length through the voxel, the same length may be used for all energy bins. For 8 energy bins, this resulted in a near 8 times speed up.

The energy bin data is still treated as independent data. However, exploiting the geometric similarities between energy bins over the scan was a good first step.
in exploiting the characteristics of spectral CT data.

**Dead detector elements**

One of the motives of selecting algebraic reconstruction was the fact that dead detector elements can simply be removed from the equation. The remaining system is just a little more under-determined than before. So mathematically, the effects should be minor as long as enough data remains.

However, removing dead detector elements does have unexpected drawbacks. It suffers from the same issue as the Kaczmarz equation as line structures are injected into the solution. These discontinuities can add noise into the reconstruction.

In the case of isolated dead detector elements this is problematic because the only ray which can fill the void is the one which was removed. All other rays either miss the void completely or only intersects at one point as an impulse error. This means that the reconstruction will contain the discontinuity for a long time, or it may even become permanent.

To lessen the effects of this issue, mART version 2 a single step in the reconstruction included all rays from both the target projection as well as the opposite projection. This means that all holes are filled in except the uncommon case where both sides of the scan have matching dead regions.

**Implementing SART**

Implementing SART was not as straightforward as the algorithm would suggest. Because the update to the volume is a weighted average, it meant that 3 copies of the volume data had to be stored. The first was the volume itself, the second was the update volume, the third was the sum of the weights per voxel.

This was problematic because for 8 energy bins, this meant storing 24 volumes. An initial glance at the weight volume suggests that it could be shared over the energy bins. However, due to the fact that mART ignores dead detector elements, this was not possible. Each energy bin has a unique pattern of dead detector elements as each energy bin is acquired with its own counter. Therefore, the sum of the weights was different for each energy bin.

It was also possible to store the volume estimate on disk and load it each time an update was necessary. However, this would cause a major performance hit
which should be avoided. Instead, the adopted solution was to order workstations for the MARS team with 48 GB of memory.

**Testing mART version 2**

A good example of the results that mART version 2 produces is shown in Fig. 4.10. This test used the same FatCaFe data from the second test of mART version 1. In this case reconstructions from mART and Octopus CT are compared. The images clearly show an improvement in the signal to noise of the image.
Also note that the Octopus CT images have some minor distortion around the calcium tube (brightest tube). This was due to the geometric model in Octopus CT which does not match the MARS system. This distortion is not present in the mART images.

To further verify this, the standard deviation, minimum, and maximum linear attenuation values are shown in table 4.7.2. These were acquired from a circular region in the calcium tube of the phantom. As expected, the standard deviation and range of the values are significantly smaller in mART than Octopus CT.

The results obtained from tests of mART version 2 were published at the IVCNZ conference [48]. Also, a study of human excised atheromatous plaques was led by Raja Panta which used mART to produce the reconstructions for his study. This was published in New Zealand Medical Journal [22].

### 4.8 Summary of mART

The software mART is the replacement for Octopus CT in reconstructing MARS data. Initially, mART adopted the Kaczmarz solution to algebraic reconstruction but this quickly changed to an implementation of SART. The choice of Kaczmarz and SART was based on their simplicity, the ability to model custom geometry, and the ease with which they can be implemented for the MARS team.

To complete the implementation, the MARS scanner was modelled using a set of rotations and translations so that the positions of the x-ray source and the detector element are known relative to the volume.

The projection routine adopts raytracing for both forward and back projection. Raytracing follows a line through the volume to determine the path length of the ray through each voxel. The path length values were shared between energy bins so that all energy bins are reconstructed simultaneously.

The outcomes of the tests show that mART can reconstruct MARS datasets
with quality that is equal to or better than Octopus CT. With these positive results, I concluded that the original goals for the design of mART had been achieved. Therefore, mART was released to both the MARS team and our external partners.
Chapter V
Digital Imaging and Communications in Medicine

The work described in this chapter started in July 2012 and it covers the adoption of the DICOM standard in the MARS molecular imaging system. The DICOM standard provides protocols for storing, managing, and transferring data. The latest revision of the standard (2014b) is divided into 20 parts where some parts are obsolete from previous revisions and have been removed [76–93].

DICOM is also compatible with radiology information systems (RIS) which allow database management of patient data, as well as requesting and managing scans from a variety of modalities. It can be said that DICOM and RIS systems form a critical component of modern medical practice.

Currently, there is no DICOM standardization for spectral CT. This meant that the shift to DICOM for the MARS molecular imaging system required some massaging of existing DICOM definitions to find a balance between supporting the data and compatibility with DICOM software. This chapter addresses the problem of designing a spectral CT DICOM data structure. Two new DICOM data structures were developed, one for projection data, and one for reconstructed data.

During this time period, CERN released a revision of the Medipix 3.0 ASIC designated Medipix 3.1. This revision came as a response to the observed instabilities in Medipix 3.0. However, the faulty charge summing mode was not yet completed. A further revision called Medipix 3.2 was promised later.

As Medipix 3.1 was integrated into MARS cameras, we first conducted a study into its characteristics. This study was led by Dr. Michael Walsh and published in the Journal of Instrumentation [94]. I participated in the preparation of the data for analysis. The most important observation of the study was that Medipix 3.1 does not suffer from a temperature dependence as with Medipix 3.0. This means that interpolating between sets of flatfield exposures is no longer necessary.
5.1 Summary of the condition of the MARS system

This section provides an overview of relevant changes to the MARS system as of July 2012. In this chapter, the relevant components include the state of the scanner hardware, the state of the scanner software, the state of the reconstruction software, the state of the material decomposition software, and the state of the visualization software.

5.1.1 MARS scanner hardware

By July 2012, the initial designs for CT10 were in active development. An important design feature was the adoption of slip ring technology. This would allow free rotation of the gantry as opposed to the single revolution permitted by the cable management technology in previous prototypes.

Also, Dr. Stephen Bell had became the project leader for both the hardware and software development of the MARS scanner. This meant that the development of the spectral CT DICOM definitions in this chapter was a collaboration between Dr. Stephen Bell and I.

Another important change was the emergence of Medipix 3.1. The initial tests for the 3.1 series showed vast improvements in stability. In fact, with version 3.1, it was no longer needed to interpolate between before and after sets of flatfield exposures. This was confirmed in our study led by Dr. Mike Walsh [94].

Along with Medipix 3.1, the MARS team received new detectors with GaAs sensor crystals. It was hoped that the ability to manufacture GaAs would make it a good middle ground between quality and detector efficiency. Unfortunately, the quality of the first GaAs sensor crystals was as poor as the existing CdTe Medipix cameras. Fig 5.1 shows an exposure from a dual GaAs MARS camera which suffers from large dead regions and visible distortion.

5.1.2 MARS scanner software

In preparation for CT10, a continuous rotation scan mode had been added to the scanning software. Even with the lack of a good lead in/lead out period the scanning times were over 10 times faster. This was a welcome addition together with the improved stability of Medipix 3.1.
However, continuous rotation suffered from synchronization issues. This meant that two exposures from a projection taken at different camera positions, would probably not be acquired at the same angle. This broke the assumptions made in stitching that all exposures are on the same plane.

Luckily, mART does not require stitched exposures and was unaffected. This was a good validation of the original design decisions made when creating mART. Still, better synchronization will ensure more consistent angular coverage during the scan so improvements were necessary.

A big issue arose with the geometric model developed in section 4.5. While the model is precise, there were no standard geometric alignment procedures on the scanner to measure the variables. Up until this point, Octopus CT was used to determine geometric characteristics such as the center of rotation. With the inability to stitch exposures, Octopus CT could not be used for continuous rotation scans. This meant that there was still a dependence on the start/stop scans for geometric alignment.
Figure 5.2: A comparison of performance when reconstructing with 1, 2, 4, 8, 16, and 32 cores on a single node of a Power 7 based supercomputer. The black line, shows a prediction for the expected performance with 64 cores by fitting a power function to the acquired data.

5.1.3 MARS reconstruction software

The use of ART allowed better reconstructions under the poor imaging conditions of the current CdTe and GaAs Medipix detectors. A brief study was performed with the Bluefern Power 7 system to test the scalability with more computing cores.

The test consisted of reconstructions of a small phantom dataset with $128 \times 128 \times 64 \times 8$ voxels. The reconstructions were performed using 1, 2, 4, 8, 16, and 32 cores on a single node of the Power 7 supercomputer.

The reconstruction times are shown in Fig. 5.2. As can be seen, the advantage of new cores tapers off over time but 32 cores is close to 9 times faster then with a single core.

The other outcome of this test is an assessment for the potential to use Bluefern as a platform for high performance computing for the MARS team. Although the test results show promise, the time taken before the job started was over two days for 32 cores. This was due to competition with jobs from other groups. This shows
that while the computing power of Bluefern is perfect for our needs, we do need hardware dedicated to the MARS project if we want fast reconstructions. For now, dedicated workstations with 8 cores will suffice for the needs of the MARS team.

5.1.4 MARS material decomposition software

A functional version of Christopher Bateman’s material decomposition algorithm was being tested which decomposes a single slice from a volume. His results showed promise, but it was clear that beam hardening artefacts posed critical issues for material decomposition. The energy signal is too distorted for accurate decompositions.

Discussions had started on the potential to combine reconstruction and material decomposition into a single process called simultaneous material reconstruction.

Material reconstruction would adopt the polychromatic form of the Beer-Lambert law. This would have the advantage that beam hardening becomes valid signal rather than an image artefact. Also, a true statistical model would be developed for the MARS system. This is the ideal reconstruction solution for the MARS team.

The investigation was led by Christopher Bateman, while my involvement was as a consultant. This left me free to continue supporting the MARS team while still being kept in the loop on the material reconstruction algorithm.

5.1.5 MARS visualization software

The data format adopted by Exposure Render is the raw image format which is a simple dump of image data. As both Octopus CT and mART produce TIFF images, a mini python based application was developed by Alexander Chernoglazov to convert a list of TIFF files into the raw format together with some descriptor files. The conversion script meant that the image processing chain from the scanner to visualization was now complete.

5.2 An overview of DICOM

DICOM is a singular, large, complex standard. In fact, a better description would be to call DICOM a collection of standards with a single common goal; to unify
data management in medicine. However, from a global perspective, the DICOM standard could be split into two parts; the definitions for data structures, and the definitions for network communications.

5.2.1 Data structures in DICOM

A DICOM instance is simply a collection of data structures. This collection is called an information object definition (IOD). A DICOM IOD contains a set of DICOM modules. Each module represents a group of data structures with a common goal. The data structures themselves are called DICOM tags.

DICOM tag definitions

A DICOM tag represents a single piece of information. This could be image data, video data, audio data, metadata, patient information, etc.

Each DICOM tag is identified by a group number and an element number. Generally, groups combine data with a common theme. In contrast, elements simply register the tags with a unique number.

Every DICOM tag has a value representation (VR) which describes the data type. This might be a string (text), or a float (real number), or a signed or unsigned integer. Each of these types also come in various sizes e.g. a 16 bit unsigned integer (unsigned short) is designated by VR = US. There are also special types for codes and unique identifiers. The VR allows software to interpret DICOM tags correctly.

Along with the VR is the value multiplicity (VM). The VM represents the required number of values in the tag. A tag might contain a fixed number of values e.g. VM = 16 for a 4×4 matrix. Alternatively, the tag might contain an unspecified number of values e.g. VM = 3 − n means that the tag must contain at least 3 values. Normally the limit n is either defined by other tags e.g. the number of pixels is given by rows × columns. Otherwise, the DICOM module might explicitly define the limits e.g. the “image type” tag has a VM of 2 − n but many DICOM modules require 3 values to be present.

DICOM module definitions

A collection of DICOM tags is called a DICOM module. The DICOM module combines tags which define a specific characteristic. For example, the image pixel
module contains tags which state the pixel format (greyscale, coloured), the pixel dimensions (rows, columns) and spacing, etc.

DICOM tags are not unique to a single DICOM module. In fact, a single DICOM tag may be present in a variety of DICOM modules but have a different purpose in each. The purpose of a DICOM tag would be defined explicitly in each DICOM module.

Depending on its purpose, a DICOM tag would be given type designating its importance to the DICOM module. There are 5 different DICOM tag types which may be used as listed below.

- **Type 1**: A mandatory tag which must be present and contain valid data.
- **Type 1c**: A type 1 tag which is only present under given conditions.
- **Type 2**: A mandatory tag which must be present but does not necessarily contain valid data.
- **Type 2c**: A type 2 tag which is only present under given conditions.
- **Type 3**: An optional tag which may or may not be present. When present, it is expected to contain valid data.

The type 1 DICOM tag is a tag which can be relied upon. For example, a DICOM instance which represents an image should always have the tag containing pixel data. As both presence and validity are guaranteed by the standard, it is type 1 tags which may be used to implement functionality.

The type 1c DICOM tag is a type 1 DICOM tag except that its presence is conditional. The conditions for presence are explicitly defined in the DICOM module. As with type 1 DICOM tags, type 1c can also be relied upon for functionality. However, it is the responsibility of the implementing software to account for the necessary conditions.

An example of a type 1c tag is the rescale intercept and slope tags which implements the linear pixel scaling \( y = sx + i \) where \( y \) is the scaled value, \( x \) is store pixel value, and \( s \) and \( i \) are the rescale slope and intercept values respectively. The rescale intercept \((i)\) is only mandatory if its value is not 0. The rescale slope \((s)\)
is only mandatory if its value is not 1. Otherwise, the two tags may be omitted because no scaling is performed i.e. $y = x$.

The type 2 DICOM tag is an informative tag which represents critical information. As the validity of the data is not guaranteed, type 2 DICOM tags cannot be relied upon for the software’s functionality.

However, type 2 tags do contain essential information. Consider the name and identification number of the patient. Depending on the conditions, this data may not be immediately known. However, both the name and identification number of the patient are critical when linking the DICOM instance to the patient. Therefore, the software will create the DICOM instances with empty tags for the patient name and identification number. As the information becomes available, the tags will be filled in correctly. This allows the patient to be treated immediately while still guaranteeing the presence of critical information.

The type 2c DICOM tag is simply a type 2 DICOM tag which is required under certain conditions. As with type 1c DICOM tags, the conditions are defined by the DICOM module.

Lastly, the type 3 tag is an optional tag. This can contain any supplementary and informative data. Information such as the manufacturer of the scanner is useful but is certainly not critical information.

**DICOM information object definitions**

A DICOM information object definition is a collection of DICOM modules. For example, the CT DICOM IOD represents an image acquired with the CT modality. The modules required will include image and pixel modules, patient descriptor modules, study descriptor modules, system descriptor modules, etc. This means that a single DICOM instance not only provides data, but it links the data to the patient, describes the conditions that created the data, and catalogues the data so that it may be accessed worldwide if necessary.

As with DICOM tags inside a DICOM module, a DICOM module inside a DICOM IOD has a predefined type. The type may be mandatory, conditional, or optional.

Referring back to the CT DICOM IOD, the image specific DICOM modules will be mandatory as the data is meaningless otherwise. DICOM modules such as
the contrast bolus module are conditional when a contrast agent is applied. Lastly, DICOM modules such as the manufacturer modules are optional. They are useful for logging purposes and quality assessment but do not normally affect medical practice.

**Private DICOM definitions**

The DICOM standard also allows vendors to create private DICOM modules and tags. Naturally, vendor specific data will suffer from compatibility issues. Therefore, in most cases, private tags generally represent supplementary information which is not required to view the data correctly. Still, major vendors do publish their private definitions so that DICOM software can fully support their tags.

**DICOM data summary**

To summarize, a DICOM instance represents a DICOM IOD. A DICOM IOD contains a set of DICOM modules which contain, describe, and catalogue the instance data. Every DICOM module contains a set of DICOM tags. Each DICOM tag represents a single piece of information. Together the DICOM tags in a DICOM module represent information with a common theme.

Both DICOM tags and DICOM modules have rules which describe their importance. DICOM modules may be mandatory, conditional, or optional. DICOM tags may be type 1, type 1c, type 2, type 2c, or type 3. Together the DICOM tag, module, and information object definition provides a flexible way to package data.

**5.2.2 Networking in DICOM**

DICOM also provides protocols for communicating and transferring data over networks. The networking standards are built on top of existing standards such as TCP (transmission control protocol) and TLS (transport layer security) for basic networking and secure networking respectively.

**DICOM network services**

The basic building blocks for the networking protocols are a series of communication services. These are listed below.
• C-ECHO: A verification service to establish if both parties are available.

• C-FIND: A querying service to search for DICOM instances.

• C-GET: A retrieval service to fetch DICOM instances.

• C-MOVE: A transfer service to send DICOM instances from one server to another (including yourself).

• C-STORE: An upload service to store a DICOM instance on a server.

• N-CREATE: A service to create a DICOM instance on a remote server.

• N-DELETE: A service to delete a DICOM instance on a remote server.

• N-GET: A service to fetch a set of DICOM tags from a remote server.

• N-SET: A service to update a set of DICOM tags on a remote server.

• N-ACTION: A service to stimulate a task on a remote server.

• N-EVENT-REPORT: A service to update other servers when events occur.

Each of the DICOM services use specific DICOM instances to represent their tasks. For example, a C-FIND instance would contain the query search terms.

DICOM network protocols

DICOM networking protocols execute specific tasks. This could be locating and retrieving data, requesting scans or image processing, verifying data integrity, etc.

The protocols are implemented using the DICOM networking services listed above. For example, the protocol to download a set of DICOM instances requires the C-FIND service to locate the instances, the C-GET or C-MOVE service to request that the DICOM instances be transferred, and the C-STORE service to receive the incoming DICOM instances.

The DICOM networking protocols can also impose constraints on the DICOM networking services. An N-ACTION service represents a request for a task to be
performed. The specifics of the task would be defined explicitly in each relevant DICOM networking protocol. For example, the modality worklist protocol can request that a scan be performed. An N-ACTION service is defined in the modality worklist to change the status of the request between pending, working, failed, or completed.

**DICOM network association**

Naturally, given the large number of protocols, it is impractical to expect applications to support every networking feature in the DICOM standard. In fact, DICOM applications generally choose which subset of the DICOM standard to support and then publish a conformance document to inform potential users.

To assist with the differences in software support, a DICOM networking session first establishes an association. A DICOM association is a formal agreement between the server and the client (note that the server/client roles may switch depending on the circumstances). The agreement is negotiated based on the computer identity, the user identity for secure networking, the list of DICOM protocols which each side supports, and the networking performance each side is capable of. Once completed, both parties can safely begin communications.

**DICOM network summary**

To summarize, the DICOM networking standard provides a set of DICOM networking protocols for specific tasks. The DICOM networking protocols are implemented using a set of DICOM networking services. Each DICOM networking service performs a single task over a network connection.

Before the DICOM networking services can begin operating, a DICOM association must be established between the two parties. This is a formal agreement on what operations may be performed. The agreement is based on the DICOM compliance, security, and identity of each party.

### 5.3 DICOM information object definitions for the MARS system

The design of the MARS molecular imaging system is spread over a variety of computer systems. The scanner deals with data collection. Then there is a DICOM
Due to the design of the MARS molecular imaging system, two DICOM IODs are required. The first is for projection data acquired from the scanner while the second is for the reconstructed data.

Of these data formats, the reconstructed data is the most important. Typically, medical imaging devices perform all the image processing locally with custom formats. Only the final outcomes i.e. the reconstructed data, are stored on the DICOM database which is called the picture archiving and communication system (PACS).

However, in the current state of the MARS system there are benefits to including the raw and normalized data on the PACS. Some of our external partners who use the MARS system have their own reconstruction techniques which they are developing. Therefore, for compatibility reasons, there is an advantage to presenting DICOM formats at all major stages of the image processing chain. This modular design allows any component in the system design to be replaced as desired.

The DICOM standard does offer an existing IOD for raw data formats. This is an empty IOD which is populated by private tags. While this is effective in
packaging vendor specific data, it is not viewable in standard DICOM viewers (software to display DICOM images).

This is not an issue if the data is intended to be processed immediately. However, in the current state of the MARS system, we wish all stages of the image processing chain to be viewable. This is both for diagnostic purposes as well as for compatibility with our external partners. As the MARS system matures into a commercial venture, the conditions will naturally change and the raw data IOD will most likely become the better choice.

5.4 DICOM IOD for raw projection data

The raw data represents exposures measured in counts. An exposure from a MARS camera contains measurements from a set of Medipix detectors. Each Medipix detector will acquire a number of measurements per detector element per energy bin.

In terms of data we can divide the information into groups. For example, the energy bins could be stored as separate DICOM instances. However, as the geometric and exposure settings are identical for the energy bins, it is more efficient to store them as frames in a single DICOM instance. This saves space as well as limiting the number of instances that will be produced. Note that a large scan will easily have 100000+ exposures which will have significant latency issues when transferring and processing the data.

To this end we can look at the nuclear medicine (NM) IOD for guidance. This IOD contains a flexible multi-frame system where the multiple frames can represent a variety of parameters such as energy and time. This is achieved by using the “frame increment pointer” tag to refer to a list of indexing tags.

For example, a “vector of energies” tag would index the energy for each frame while a “vector of time steps” tag would index the time for each frame. Together, these vectors allow both time and energy to be stored as multiple frames as illustrated in Table 5.1.

By inheriting the multi-frame modules from the nuclear medicine information

---

1 While the MARS system does not collect time varying data yet, it is highly likely to be a future development. Therefore, it is better to consider supporting time varying data at an early stage.
Table 5.1: An example of the indexing scheme of the nuclear medicine multi-frame module. This example contains two energies and two time steps.

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Group</th>
<th>Element</th>
<th>V.R.</th>
<th>V.M</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detector rotation angle</td>
<td>0x0023</td>
<td>0xXX09</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Detector x index in image</td>
<td>0x0023</td>
<td>0xXX0A</td>
<td>US</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Detector y index in image</td>
<td>0x0023</td>
<td>0xXX0B</td>
<td>US</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Detector axial position in camera</td>
<td>0x0023</td>
<td>0xXX0C</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Detector tangential position in camera</td>
<td>0x0023</td>
<td>0xXX0D</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
</tbody>
</table>

Table 5.2: The nth detector module which contains the geometric and indexing tags for each detector in the image. Only the relevant tags are shown. The indexing is achieved with the element number. The first two digits (XX) start from 10 and increase per detector module. Note that the group and element numbers are given in hexadecimal form. FD stands for double precision floating point. US stands for unsigned short.

object definition we gain the advantage of borrowing its fully standardized functionality. All the tags required to index the multiple frames already exist. This means that any DICOM viewers that fully support nuclear medicine instances should have built in functionality to properly display our data.

It would also be convenient to treat all the detectors in a camera as a single unit. However, in this case, the geometric parameters are unique to each detector. Therefore, it is not appropriate to add the detectors as components of the multi-frame system.

Instead, our design simply concatenates the detectors into a single pseudo-stitched image. This allows any user to view the complete detector array in the camera in a single instance. Unfortunately, to index the detectors and store their geometrical properties, we have to rely on private tags for functionality.

Table 5.2 shows the relevant tags in the private detector module. This module is a generic definition which is repeated per detector in the camera. For example, a camera with two detectors will contain two instances of the private detector module.
In addition to the geometric parameters of the detectors, other geometric parameters also need to be stored to properly position the exposure. To this end, there are currently two private modules in use. These are shown in Tables 5.3 and 5.4.

The image information module provides a simple indexing scheme while the gantry motor module provides the precise placement of the camera and x-ray source during the time of the exposure. Unlike the previous scan report files, the DICOM tags now use the S.I. units of millimeters and degrees. The MARS system has matured enough that the diagnostic value of motor steps is obsolete.

In the case of the raw projection IOD, the use of private tags for functionality is appropriate. Standardized tags do not exist which match the specific design of the MARS scanner. Also, the geometric properties are only required by reconstruction and stitching algorithms. The geometric private tags do not affect functionality when visualizing the data. The functionality of these tags is only needed by MARS software and won’t cause issues elsewhere.

Other than the private modules mentioned above, the MARS scanning software

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Group</th>
<th>Element</th>
<th>V.R.</th>
<th>V.M</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gantry rotation angle</td>
<td>0x0019</td>
<td>0x1001</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Camera tangential position</td>
<td>0x0019</td>
<td>0x1002</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Subject axial position</td>
<td>0x0019</td>
<td>0x1003</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Source radial position (Source to object)</td>
<td>0x0019</td>
<td>0x1004</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Camera radial position (Object to detector)</td>
<td>0x0019</td>
<td>0x1005</td>
<td>FD</td>
<td>1</td>
<td>Type 1</td>
</tr>
</tbody>
</table>

Table 5.3: The gantry motor module which contains the gantry, camera, subject, and source positions. Only the relevant tags are shown.

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Group</th>
<th>Element</th>
<th>V.R.</th>
<th>V.M</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requested energy bin vector</td>
<td>0x0019</td>
<td>0x1306</td>
<td>FD</td>
<td>1-n</td>
<td>Type 1</td>
</tr>
<tr>
<td>Subject position sequence number</td>
<td>0x0019</td>
<td>0x1309</td>
<td>US</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Camera position sequence number</td>
<td>0x0019</td>
<td>0x130A</td>
<td>US</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Number of detectors in image</td>
<td>0x0019</td>
<td>0x130E</td>
<td>US</td>
<td>1</td>
<td>Type 1</td>
</tr>
<tr>
<td>Gantry angle sequence number</td>
<td>0x0019</td>
<td>0x130F</td>
<td>US</td>
<td>1</td>
<td>Type 1</td>
</tr>
</tbody>
</table>

Table 5.4: The image information module which contains the indexing scheme for the image. Only the relevant tags are shown.
defines other private modules for diagnostic and informative purposes. However, as these are purely optional modules for the scanner development team, they are not included in this thesis.

The final tag of importance is the “image type”. This is an array of values where the first value indicates if the data is acquired or derived, and the second value indicates if the image is created during or after the patient examination. Other values may be defined by DICOM modules.

In the case of the MARS raw projection IOD, the third value is now defined to be either “DARKFIELD”, “FLATFIELD”, or “RAWIMAGE”. This refers to the three types of scans which are currently performed on the MARS system. As the “image type” is a type 1 tag, it may be used to verify if the data is the correct type.

The final information object definition is illustrated in Table. 5.5. The basis for the IOD is the computed radiography (CR) IOD. As the raw data represents a radiograph, the CR IOD is more appropriate then the CT IOD which only represents reconstructed slices.

5.5 DICOM IOD for normalized projection data

The normalized projection data reuses the DICOM IOD defined for the raw projection data. Structurally and geometrically, raw and normalized projection data are identical. The only difference is the unit of measure that each represents. Therefore, the required tags are the same with a slightly different use. To mark this difference we introduce a fourth option for the third value of the “image type” tag as “PREPROCESSED”. This informs software that the data is now normalized projection data.

The first consideration is which unit of measure to adopt for the normalized projections. With the previous TIFF images, the normalized projection data represented the intensity ratio $\frac{I}{I_o}$ due to the requirements of Octopus CT. However, as DICOM data is incompatible with Octopus CT, this is no longer a constraint.
Table 5.5: The information object definition for the raw data. The three private modules defined in this chapter are shown in bold at the bottom. Also shown in bold is the multi-frame system from the NM IOD. The other modules are straight from the definition of the CR IOD. Optional private modules are not shown here. M stands for mandatory, C stands for conditional, and U stands for user option.
5.5.1 Data types, binning, and precision

There are restrictions in the current DICOM standard which mean that $\frac{I}{I_0}$ is still the best format. The pixel data in the DICOM standard may be either 8 bit or 16 bit. There is no standardized format for 32 bit data streams. While it is possible to force DICOM instances to use 32 bit data streams, there is no guarantee that DICOM software will interpret the data correctly. Some may default to 16 bit, others may translate the endianess\(^2\) of the data incorrectly. Either way, the results will most likely be garbage.

In addition, the only supported pixel data types are signed and unsigned integers. There is no standardization for floating point values. Instead, the “rescale intercept” and “rescale slope” tags allow software to restore the original real values from the pixel data.

The rescale tags implement a linear pixel scaling which is automatically applied in DICOM software. However, since the scaling is linear, there may be issues with precision in derived data.

For example, if the chosen representation were to be transmission values ($\tau = -\ln \left( \frac{I}{I_0} \right)$), the logarithm would stretch the histogram towards low density materials.

However, the binning required to convert real numbers to integer values has constant spacing which means that the precision in low density materials would be reduced. This effect would worsen depending on the transmission value of the most dense material in the exposure. Luckily, $\frac{I}{I_0}$ represents a linear transformation of the original acquired data. Therefore, no loss in precision should be observed.

Another convenience with $\frac{I}{I_0}$ is that the values are limited within the range $0 < \frac{I}{I_0} \lesssim 1$. This means that constant scaling can be applied (I used 65535/1.5 where 1.5 is a weak threshold. See chapter 6 for more details). The advantage of constant scaling is that the binning width during conversion is consistent between datasets.

To summarize, $\frac{I}{I_0}$ does not distort the precision of the original acquired data. Therefore, it remains the best choice in representing normalized projection data.

\(^2\)Endianness defines how the bytes are ordered in memory. As an analogy in language, consider twenty four as opposed to four and twenty. The same number is represented in opposite ways.
5.5.2 Dead detector element representation

The next complication is the presence of dead detector elements. In raw projection data, dead detector elements have not yet been identified. In the reconstructed data, there are no dead voxels. However, in the normalized projection data, the dead detector elements must be represented.

A simple solution emerges from the range of $\frac{I}{I_0}$. A value of 0 is unusable data in the context of the current reconstruction algorithm due to the logarithm in the Beer-Lambert law. Therefore, we can use value 0 as a marker for dead detector elements.

The DICOM standard offers the “Pixel Padding Value” tag which references pixels which are to be ignored. In CT images, the “Pixel Padding Value” is used to ignore the image corners outside the circular region of interest. Using the tag in the same manner, the “Pixel Padding Value” tag becomes a valid marker for dead detector elements in the image.

5.5.3 Normalized projection data summary

The normalized projection data is represented by the same DICOM IOD as the raw projection data. However, the data now represents $\frac{I}{I_0}$ which is useful as it provides a format without distorting the precision of the original data. In addition, it offers a convenient method of marking dead detector elements by using the value 0.

5.6 DICOM IOD for reconstructed volume data

The IOD for the reconstructed spectral CT volumes is the most important of the IODs introduced in this thesis. It represents the data which will be visualized and analysed. In addition, it may form part of an official standardized spectral CT IOD in the near future.

Spectral CT is an extension of CT. Therefore, it is only natural that the new IOD is built up from the CT IOD. However, there are some fundamental differences with spectral CT data which means that the CT IOD can not be used directly. These differences include the multiple energy bins, but more importantly, the unit of measure that the data will represent.
5.6.1 Multi-energy and multi-frame images

The multiple energy bin issue is simple to deal with. In the projection data IOD, the multi-frame module from the nuclear medicine information object definition was adopted. It will also work well in this circumstance as the spatial information of the energy bins are still identical from one bin to another. In addition, the multi-frame module already provides support for time varying data for when it will be needed in the future.

5.6.2 Hounsfield units, linear attenuation, and density

The unit of measure for the spectral CT IOD is the greatest concern. The CT IOD has a restriction which states that the unit must be in Hounsfield units. The Hounsfield unit is a non S.I. unit which scales the linear attenuation values such that water becomes 0 and air becomes -1000. Eqn. 5.1 shows the conversion algorithm where HU is the resulting Hounsfield unit, and $\mu$, $\mu_{\text{air}}$, and $\mu_{\text{water}}$ are the linear attenuation values for the target voxel, air, and water respectively.

\[
HU = 1000 \frac{\mu - \mu_{\text{water}}}{\mu_{\text{water}} - \mu_{\text{air}}}
\]  

(5.1)

In the past, Hounsfield units were convenient for two reasons. Firstly, patients are mostly water and are scanned in air. Therefore, the Hounsfield unit shifts the histogram of the two most common substances to known values. Secondly, the conversion provides a scale which is easy to bin to integer values. This means that the Hounsfield unit works well with the design of DICOM image instances.

However, Hounsfield units are energy dependent. This means that in the context of spectral CT, the conversion is not a trivial process as it requires knowledge of $\mu_{\text{water}}$ and $\mu_{\text{air}}$ for every energy bin. While it is possible to use known data such as the tables from NIST (National Institute of Standards and Technology), this requires intimate knowledge of the source spectrum, the detector response, and any filtration used to match the conditions of the values from the tables. Also, a polychromatic reconstruction algorithm would be needed to eliminate any distortions to the signal. Only then would the use of NIST tables be accurate.

Alternatively, the parameters could be determined experimentally if a sample of water and air were present in every scan. However, this does pose an unnecessary
constraint for every scan acquired.

Another consequence of the energy dependence is the loss of presets. With current CT systems, radiologists can memorize ranges of Hounsfield units for common materials and organs. However, the energy dependence of the Hounsfield unit means that this is no longer possible. If the scanning protocols were limited to a small set of energy bins, then a table could be drawn up. However, this again poses an unnecessary constraint for every scan acquired.

Before making a final decision, let us consider what the final goal of medical imaging is: better diagnosis. Spectral CT offers a significant improvement through material decomposition to quantify the materials in the subject. This means that the current reconstructed data is not the final step in a spectral CT system. Once material decomposition algorithms are complete, the final unit would be some measure of density to quantify materials directly.

Considering the issues mentioned above, it becomes obvious that Hounsfield units are not appropriate for representing spectral CT data. It adds complexity without providing any real benefits.

Instead, the S.I. unit of linear attenuation (cm$^{-1}$) was chosen for MARS datasets. Linear attenuation is the direct output from reconstruction algorithms, and is the input for material decomposition. Also, Hounsfield units are derived from linear attenuation. This means that Hounsfield units may be obtained if desired.

5.6.3 Final IOD design

Therefore, the final IOD is summarized in Table 5.6. Notice that the definition does not depend on any private modules. Some optional private modules may be created and included to list the image processing steps and parameters. However, this is purely logging data and will not affect the function of the resulting DICOM instances.

5.7 Summary of DICOM design

To support DICOM in the MARS system, two unique IODs were designed. One IOD represents projection data, and the other IOD represents reconstructed data.

The projection data IOD can represent darkfield exposures, flatfield exposures, scan exposures, and normalized projection data. The IOD is based on the CR
<table>
<thead>
<tr>
<th>IOD Modules</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient</td>
<td>M</td>
</tr>
<tr>
<td>Clinical Trial Subject</td>
<td>U</td>
</tr>
<tr>
<td>General Study</td>
<td>M</td>
</tr>
<tr>
<td>Patient Study</td>
<td>U</td>
</tr>
<tr>
<td>Clinical Trial Study</td>
<td>U</td>
</tr>
<tr>
<td>General Series</td>
<td>M</td>
</tr>
<tr>
<td>Clinical Trial Series</td>
<td>U</td>
</tr>
<tr>
<td>Frame of Reference</td>
<td>M</td>
</tr>
<tr>
<td>General Equipment</td>
<td>M</td>
</tr>
<tr>
<td>General Image</td>
<td>M</td>
</tr>
<tr>
<td>Image Plane</td>
<td>M</td>
</tr>
<tr>
<td>Image Pixel</td>
<td>M</td>
</tr>
<tr>
<td>Contrast Bolus</td>
<td>C</td>
</tr>
<tr>
<td><strong>Multi-frame</strong></td>
<td>M</td>
</tr>
<tr>
<td><strong>NM Multi-frame</strong></td>
<td>M</td>
</tr>
<tr>
<td>Display Shutter</td>
<td>U</td>
</tr>
<tr>
<td>Device</td>
<td>U</td>
</tr>
<tr>
<td>Specimen</td>
<td>U</td>
</tr>
<tr>
<td>Overlay Plane</td>
<td>U</td>
</tr>
<tr>
<td>VOI LUT</td>
<td>U</td>
</tr>
<tr>
<td>SOP Common</td>
<td>M</td>
</tr>
</tbody>
</table>

Table 5.6: The information object definition for the reconstructed data. Currently, there are no private modules used. The bold modules are from the NM IOD. The other modules are from the CT IOD. M stands for mandatory, C stands for conditional, and U stands for user option.
IOD with the addition of the NM multi-frame system. Geometric information is
provided using private modules. This IOD creates images which can be viewed in
any DICOM viewer while also providing all the necessary data for reconstruction
and visualization.

The reconstructed data IOD extends the CT IOD. The NM multi-frame system
is used again, to group energy bins together.

The only other change to the CT IOD is the unit of measure to adopt. The
reconstructed data represents the S.I. unit of linear attenuation which is the direct
output from reconstruction algorithms. This is also more appropriate for material
decomposition which requires linear attenuation and mass attenuation to produce
a measure of the density of materials.
Chapter VI

Evolution of MCTBuilder into mPPC

The work described in this chapter started in January 2013 and it explains the additions and adjustments made to MCTBuilder during its transformation into mPPC (MARS Pre-Processing Chain). This transformation included the following changes.

• Adoption of the developed DICOM IODs.
• Characterization of detector element behaviour.
• Thresholding raw and normalized data.
• A Poisson distribution test.
• A dilation algorithm.
• Alterations to the inpainting algorithm.
• A ring filtration algorithm.

The majority of these changes were motivated by the poorer quality of the CdTe and GaAs based detectors. As a result I conducted a study to characterize what kind of signals were produced in the acquired images. From this characterization the new algorithms were developed and added to mPPC. The research questions address are summarized as follows.

• What kind of signals can the Medipix detector produce?
• How should isolated detector elements (surrounded by dead detector elements) be dealt with?
• Is there still a use for the inpainting algorithm?

• How should ring and windmill artefacts be dealt with?

Around April 2013 a team from the Dubna branch of the Joint Institute of Nuclear Research (JINR) became interested in MARS technology. Both mPPC and mART were provided to them after their purchase of a MARS scanner.

6.1 Summary of the condition of the MARS system

This section provides an overview of relevant changes to the MARS system as of January 2013. In this chapter, the relevant components include the state of the scanner hardware, the state of the scanner software, the state of the pre-reconstruction processing software, and the state of the reconstruction software.

6.1.1 MARS scanner hardware

By January 2013, the new Medipix 3RX detectors had been introduced. Initial testing suggests that this design is more stable than the previous Medipix 3.1. More importantly, the long-awaited charge summing mode was now fully functional.

However, for the image processing this did introduce new challenges as detector elements in acquired images are now coupled. Unfortunately, this coupling means that noise resulting from one detector element can bleed into neighbouring elements.

Another development was a helical scanning mode. This allowed for scans with better uniformity as shown back in Fig. 2.5.

As with the introduction of continuous rotation, the new scanning mode was successfully reconstructed with mART the first time it was tried. Again, this validates the robust design of mART which allows it to reconstruct any MARS dataset. The only requirement is that the detector and x-ray source position and orientation are known.

6.1.2 MARS scanning software

The MARS scanning software was changed to support the new design of the scanner along with Medipix 3RX. Also, the raw DICOM IOD is now supported in the
scanning software. The acquired data is immediately uploaded to a PACS for data storage.

The chosen PACS software is called Conquest DICOM. This is a freeware, open source application which has good DICOM compliance. The only downside is that remote deletion with the N-DELETE service was not possible.

In a research setting, it is natural that a lot of data is collected that will not be used. Scans are often cancelled for various reasons which produces numerous incomplete datasets. The inability to delete these incomplete datasets meant that the PACS needs regular maintenance to clean up the database.

### 6.1.3 MARS pre-reconstruction processing software

After completing the design of the DICOM IODs mentioned in Chapter 5, I added support for these IODs into mPPC. This meant that data could be transferred via the network instead of relying on the manual copies previously achieved using external hard-drives.

mPPC operates on an independent workstation together with mART. To bridge the gap via the network, K-PACS was installed to transfer the MARS datasets. K-PACS is another free DICOM application which offers a better user interface than Conquest DICOM. Therefore, Conquest DICOM acted as the primary PACS while K-PACS acted as a downloading tool.

### 6.1.4 MARS reconstruction software

Along with mPPC, I also added support for DICOM to mART. This means that the reconstructed data can be uploaded to the PACS and used in the same way as a standard CT dataset.

During testing, there were some issues with a few freeware DICOM viewing applications. CT datasets are normally expected to use Hounsfield units. This led to some applications cutting corners by representing the rescaled data as integer values\(^1\).

The MARS spectral CT IODs use linear attenuation which often does not exceed 3 cm\(^{-1}\) in a scan. As integers, this range only gives 3 valid values. Therefore, a

\(^1\) Remember that DICOM only stores integer values and a set of rescale tags converts values back into real numbers.
few DICOM applications could not display reconstructed MARS datasets correctly. After careful re-examination of the DICOM standard, I confirmed that the design of the MARS spectral CT IODs are acceptable. In fact, Ginko CAD X, K-PACS internal viewer, and Amide are all examples of better DICOM viewers which can display MARS spectral CT data correctly.

6.2 Transition to DICOM

The first major change to mPPC (formerly MCTBuilder) was the adoption of DICOM. This change has a few major consequences for mPPC.

All supplementary information including the geometry and indexing scheme are now stored with the image data. This meant that the scan report, the scan configuration files, and the directory hierarchy were now obsolete.

This allowed mPPC to be simplified. The new GUI is as shown in Fig. 6.1. Instead of a Windows Explorer interface with the directory structure, mPPC simply shows the DICOM instances in the order of acquisition.

Figure 6.1: The new interface for mPPC. The file list shows all the files in a scan in the order of acquisition. The algorithms are presented in a simple list. Lastly, the preview shows the image as well as the tags.
With the removal of stitching and sinogram conversion the chain was short enough to present in a single list instead of a wizard. This means that there was no more jumping between the pages from the old wizard-style interface to fill in the steps of the pre-reconstruction processing chain.

The preview can show either the image data or the DICOM tag information. Also, the user can switch energies as our DICOM IOD design. The view of the DICOM tag information is a basic dump of all tags which means that diagnostic tags written by the scanning software may also be viewed if desired.

To summarize, the change to adopting the DICOM standard simplified the import/export of data between the scanning software and mPPC. It also simplified the user interface which allowed mPPC to become more user-friendly.

6.3 Characterization of Medipix signals

Over the various steps of the evolution of the Medipix detector, the pre-reconstruction processing software has had to adapt to the condition of the data. The first of such changes was the interpolation of sets of flatfield exposures to compensate for the instabilities of Medipix 3.0 (see Chapter 3).

Other changes included the necessity of thresholding both raw data and normalized data. In the case of raw data, the counts might be too low to be statistically significant. However, thresholding low counts can limit the ability to reconstruct scans with dense objects. In the end, a requirement to have a minimum of 2 counts was applied. This was chosen as 0 counts can not pass through the logarithm in the Beer Lambert Law, and 1 count is too ambiguous to consider. It could be a relevant photon, or may be scatter, or even a stray photon from the environment.

There are two conditions that lead to a saturated detector element: exceeding the maximum value of the counter, or exceeding the possible count rate. In both cases the detector element value is set to a saturation value (4095 for a 12 bit counter). This makes removal easy as the saturated value is a constant.

After flatfield normalization, unexpected noise can still cause problems. The normalized values should range from $0 < n \lesssim 1$, where the normalized value for air should average at 1. However, some outliers greatly exceed the expected value of 1. A further complication arises as the average normalized value can fluctuate for long scans despite interpolating sets of flatfield exposures to correct for thermal
instability. If this new threshold is set lower then the average normalized value for air, then the resulting data becomes unusable as too many detector elements are removed. The final threshold value was set to 1.5. This imposes a weak constraint on the outliers while providing some tolerance for the fluctuations of the average normalized values.

With the improved stability of Medipix 3.1 and 3RX, some of these issues disappeared. Only a single set of flatfield exposures was now required. However, despite the thresholding of outliers, many detector elements still remained with obvious errors as shown in Fig. 6.2.

In this image, even though darkfield masking, flatfield normalization, and thresholding was applied, many abnormally bright and dark detector elements still remain scattered over the image. The next step was to characterize the detector elements to better identify bad data.
6.3.1 Ideal detector element

While the first instinct would be to say that the ideal detector element has no noise, this is most certainly not the case. The emission of photons from the x-ray source is a random process. Therefore, the physics of the problem tells us to expect some noise in the measurements. In fact, a random counting process can be modelled as a Poisson distribution.

On the other hand, it is hoped that any sources of noise resulting from electronics in any Medipix detector or inhomogeneities in the sensor crystal are negligible. Therefore, we can say that the ideal detector element would follow a pure Poisson distribution. Fig. 6.3 shows a detector element from a set of 1000 flatfield exposures which is well behaved.

6.3.2 Dead detector elements

Previously, we defined a dead detector element as one which yields a fixed value. However, a better definition of a dead detector element is one which does not produce signal related to the exposure of photons to the element.
Figure 6.4: A plot of the repeated values of a detector element with erratic behaviour. The bottom right shows the histogram of these values. Note how it is impossible to determine where the expected value should be as different regions of the plot tend towards different values. The histogram shows no obvious peak with random spikes over a wide range of values.

The importance of this new definition is that a dead detector element may reveal itself as a fixed value element or may simply produce random garbage.

6.3.3 Erratic detector elements

Erratic detector elements are cases where the detector element produces garbage as a response to exposure to photons. This means that there is still an underlying signal which is lost behind the noise of the detector element. Fig. 6.4 shows a detector element from a set of 1000 flatfield exposures which behaves erratically. The signal is completely lost under noise which means that a reasonable expected value can not be determined.

An important distinction must be made between dead and erratic detector elements. In both cases, the response can be garbage. However, because erratic detector elements only respond to photons, they are harder to detect. A dead detector element may count during a darkfield exposure while an erratic detector element will not. This means that other methods of identification are required.
6.3.4 *Unstable detector elements*

Unstable detector elements are ones which record valid data for some exposures, and erroneous data in others. If the frequency of the switch between valid and invalid is high enough, these may appear as erratic detector elements. The difference is that despite appearances, these detector elements make valid measurements at times. Determining which exposures are valid and which are invalid can be very problematic.

For example, if the frequency of the switch is very low, then all the collected flatfield exposures might contain valid data while the scan contains invalid data. In this case, the problem of detection becomes much more difficult. Fig. 6.5 shows an example of an unstable detector element which is switching on and off.

6.3.5 *Truncated detector elements*

There are a few conditions which could lead to a truncated response. The first is when there is insufficient exposure to penetrate the subject. In this case, the starved detector element may yield 0 counts. Even in the case where a few counts
Figure 6.6: A plot of the repeated values of a detector element with a truncated response behaviour. The bottom right shows the histogram of these values. In this case the truncation is moderate, the pixel has tolerable behaviour otherwise.

are registered, these may simply be due to scatter.

The other possibility is when the exposure is too long, or the tube current is too high. In the former case, the counts may exceed the limits of the 12 bit counter which stops at the maximum value. In the latter case, the rate of photons absorption events is too high which means that counts are missed as the pulse shapers, the sensor, etc., become saturated.

The impact of truncated detector elements differs depending on the exposure type. In the case of a scan, each exposure is a unique view of the subject. This means that each exposure can be considered independently. Removing a few truncated values will not affect the exposures which register appropriate values.

However, in the case of flatfield exposures, we have to consider the effect the truncation has on the average and variance calculations. In both cases, these will deviate from the true values. The amount of deviation depends on how often the truncation occurs. Therefore, in the case of the flatfield exposures, if the number of truncated values exceeds 95% of the exposures, the detector element is removed. 95% was an arbitrary choice which imposes a weak constraint on the data.

Fig. 6.6 shows an example of a truncated detector element. In this example
Figure 6.7: A plot of the repeated values of a detector element which drifts over time. The top right shows the histogram of these values. In this case the drift is severe along with excessively noisy behaviour.

6.3.6 Drifting detector elements

Drifting detector elements are due to instabilities in the detector with the result that the average value shifts over time. In the previous situation with Medipix 3.0, the whole detector suffered from drift. Even though Medipix 3RX is stable as a whole, there are still isolated detector elements which drift.

With Medipix 3.0, the origin of the drift was a temperature dependency. A constant amount of energy was continuously applied to the system which meant that the change was roughly linear. However, in the case of isolated detector elements with no temperature sensitivity, the cause is unknown. This means that unlike before, a simple correction scheme is not possible. Therefore, the best response is removal. Fig. 6.7 shows such a drifting detector element with no hope for correction.
6.3.7 Identification and characterization

The problem of identification relates to the available data. Consider the exposures in a scan. In truth, the detector elements are not in a regular grid. Some detector elements are larger than others. This means that a perfectly calibrated exposure appears noisy as larger detector elements will naturally count more. Therefore, it is not appropriate to compare raw values with their neighbours.

A normalized detector element can be compared with its neighbours. Dividing the scan data by the flatfield data will balance the differences due to the detector element size. However, even though we can compare a normalized value with its neighbours it is better not to.

The important point is that projection data does not have the same properties as reconstructed slices. In reconstructed slices, parts of the signal can be assumed to be sparse e.g. total variation, derivative, etc. This is basis for using compressed sensing in reconstruction.

However, this can not be equally assumed with projection data. A homogeneous cylindrical object in a scan will generate a projection with gradients as rays pass through the different depths of the cylinder. Therefore, assuming that the signal in one detector element is the same as in another is a weak, high risk assumption which should be avoided.

Another issue with comparisons to the neighbourhood are the bad detector elements themselves. The unidentified bad detector elements will skew results, while the identified bad detector element will bias the sample set as less samples are collected from one region then another. Either way, the results will likely not be as intended.

On the other hand, sets of flatfield exposures are perfect for measurements. The repeated exposures are guaranteed to have the same underlying signal. Therefore, we can measure the resulting data over the repeated samples per detector element.

Of course, this assumes that the condition of each detector element is the same between the flatfield and the scan. Unfortunately, this is not always the case which means that flatfield based methods are not sufficient to characterize the detector by themselves.
6.3.8 **Algorithms to remove bad detector elements**

To better identify and remove bad detector elements, I added a set of new algorithms to mPPC. These include the thresholding mentioned earlier, a Poisson distribution test, bad detector element dilation, user guided masking, revised inpainting, and a ring filter. The old algorithms such as darkfield masking are still included\(^2\).

**Poisson distribution test**

With the knowledge that the ideal detector element has a Poisson distribution a simple test is possible. A basic property of the Poisson distribution is that the variance equals the average. The algorithm is given in Eqn. 6.1.

\[
\text{threshold} \geq 100 \times \begin{cases} 1 - \frac{\text{variance}}{\text{average}} & : \text{variance} \leq \text{average} \\ 1 - \frac{\text{average}}{\text{variance}} & : \text{otherwise} \end{cases}
\]  

(6.1)

This test is a variation of the dispersion index which guarantees that the measure ranges from 0 to 100. This is convenient because irrespective of the scan exposure settings, we can present the measure as a percentage to the user. This gives a solid, consistent measurement to which we can apply a threshold.

The importance of a consistent measurement goes beyond mere convenience for the user. It also allows the Poisson distribution test to operate as a quick and dirty quality assessment tool. For example, consider a MARS camera which typically loses 1% of its detector elements with a dispersion of 15%. Then, a new scan suddenly results in a loss of 10% of the detector elements with the same dispersion of 15%. Obviously, some condition has changed the MARS camera for the worse.

This quick and dirty quality assessment has already had some uses. Before a scan takes place, there is a warm up period for the x-ray source. In some scans where the flatfield was acquired before the main scan, the Poisson distribution test revealed that the conditions were bad in only the first camera position. This allowed the problem to be quickly identified. The x-ray source was still warming up during the first camera position acquisition of the flatfield exposures. Lengthening

\(^2\)Remember that darkfield masking successfully identifies the majority of dead detector elements.
the warm up period solved the issue.

It is important to note that the Poisson distribution test does not say if a detector element follows a Poisson distribution. In fact, many erroneous distributions may pass this test. Instead, this test identifies distributions which are highly likely to not be a Poisson distribution. Fig. 6.8 demonstrates the results of the Poisson distribution test. It removes some of the bright and dark patches, but not all.

**Dilation of isolated detector elements**

The dilation algorithm is a process which removes detector elements if the number of good neighbours is less than a threshold. This is an important algorithm as isolated detector elements within a dead region cannot be easily validated.

The problem is that isolated detector elements have no neighbouring data. This means that only the set of flatfield exposures offer information about the detector element. As mentioned earlier, this is not sufficient to characterize detector elements as the most problematic ones behave differently between the scan and the flatfield exposures. Therefore, it is better to remove such detector elements, just in case.
User guided masking

User guided masking allows users to directly select regions for removal. This is an inferior option which was added last minute due to the necessity of producing cleaner data for studies that were being conducted. The study in question, was a feasibility study for quantifying gold for tumours in mice.

The problem with user guided masking is that the majority of the scan exposures must be personally reviewed. As scans can have 100000+ exposures, this is a time consuming endeavour. Eventually, alternative algorithms will be needed to eliminate the necessity of user guided masking.

The user guided mask can be saved and loaded for multiple scans. However, there are multiple scanners, and the cameras in each scanner can change. Also a new mask is needed each time the calibration process of the scanner is redone. This means that managing user guided masks at the image processing side is complicated.

The ideal solution is for the scanner to generate its own mask to pass on to mPPC. The scanner has direct access to the camera and can perform more tests than simply data mining the exposures from a single scan. In fact, the scanner can maintain a history of detector element behaviour for the complete life of the detector. Therefore, the quality of a scanner generated mask would be superior to any that can be created in an independent image processing application. More importantly, there would no longer be a need for user guided masking.

Concerning data management, the scanner knows when calibration procedures are performed. This means that a scanner generated mask could be kept up to date with negligible effort. Therefore, a scanner generated mask is superior to the current implementation in every way. A solution on the scanner is intended in the near future, but for now, the user guided masking in mPPC suffices.

A different role for inpainting

Originally, the intention was to remove the inpainting algorithm. However, the presences of the ring filter changes matters. The ring filter is the only algorithm in mPPC that actively relies on neighbouring information to correct a detector element. However, if a target region has a lot of dead detector elements on one side, the sample region for the ring filter is biased.
Therefore, the inpainting algorithm still has its uses. Instead of inpainting all dead regions, it acts only on small dead regions where the inpainting algorithm is still accurate. This removes the small dead regions which, in turn, improves the condition of the data for ring filtration as well as for reconstruction in mART.

**Summary of new pre-reconstruction processing algorithms**

The new algorithms that I implemented in mPPC include thresholding, a Poisson distribution test, dilation, user guided masking, re-purposed inpainting, and ring filtration which is described in the following section. Together, these algorithms allow the majority of bad detector elements to be identified and removed in preparation for reconstruction.

The process of the final pre-reconstruction processing chain is shown in Fig. 6.9. The final result still has many dead regions but it is left to the robustness of mART to reconstruct despite this.

The only major issue remaining are subtle, static errors. These are fixed offsets that do not readily change between exposures. The consequences of such errors are ring or windmill artefacts in the reconstruction. The following section covers the ring filtration algorithm implemented in mPPC.

### 6.4 Filtration of ring artefacts

There have been a few approaches to ring filtration. A common approach uses Fourier methods to target frequencies corresponding to rings. Jagadeesh et al. present a comprehensive study on these together with a novel wavelet approach in their paper [95]. However, the initial approach I followed was based on the paper by Sijbers et al. [96]. Yousuf et al. use a similar approach in their paper on ring filtration [97].

In Sijbers’ paper, two directional averaging filters are used to remove rings post reconstruction. The idea being that when transforming the reconstructed volume slices into polar coordinates, the rings turn into lines. The first directional average finds the error across the image by subtracting the average from the image. The second directional average finds the static error by calculating the average of the errors down a line. This error is then subtracted from the image to complete the correction.
Figure 6.9: The effects of mPPC’s pre-reconstruction processing chain. The steps include darkfield masking (1), flatfield normalization (2), the poisson distribution test (3), dilation (4), user defined masking (5) and the revised inpainting (6). Red areas are detector elements marked as bad.
Unlike the paper, I did not filter rings post reconstruction. The main reason for this is that ring artefacts are the result of errors from each exposure converging over a ring. Therefore, the effects on the volume are stronger than is visible in the projection data.

Luckily, when viewing sinogram data, the same phenomenon occurs where rings manifest as lines. Therefore, the same algorithm can be used on sinogram data. However, the mean filter was replaced with a median filter as it better deals with the outliers that commonly appear in current MARS data.

To summarize the algorithm follows a basic two step process.

1. Firstly, find the difference to the median between a target pixel and a line across the sinogram spanning \( n \) pixels (commonly 15 to 25 worked well in practice). This difference is an estimate of the error of this pixel.

2. Secondly, take the median of the calculated differences over a line down the sinogram spanning \( m \) pixels (commonly 100+ worked well in practice). This is now an estimate of the static error of the pixel i.e. an error which is common between exposures. Subtracting the resulting median from the original target pixel gives the corrected value.

The algorithm has two parameters which control the filtration. The first is \( n \), the size of the line across the sinogram for the first step. The second is \( m \), the size of the line down the sinogram for the second step.

Parameter \( n \) is by nature a contradictory existence. On the one hand, we want \( n \) to be large to provide a suitably large set of samples. However, the goal is to create an estimate of the static error present in the pixel. We accomplish this by assuming that the valid signal in the neighbouring pixels is the same as the target pixel. Therefore, the subtraction should cancel out the signal and the median should eliminate random noise. In theory, what’s left is the static error. Of course, the larger \( n \) is the less the assumption holds.

On the other hand, with the addition of the charge summing mode, errors are now coupled together in clusters. This means that if we wish to identify the true static error, \( n \) needs to be significantly larger than the cluster sizes. In general, charge summing mode forms clusters of size 2 and at most size 4. Therefore, the
Choose default values of 15 to 25 should be able to provide a median without skew from the clustered errors.

On the other hand, because the ring filter algorithm is a two step process, some error in the first step can be tolerated. By taking the median of the differences across a line down the sinogram we are estimating the most likely value of the static error. Naturally, this assumes that the static error is fixed for the span of the line (size $m$).

With MARS data, this can be problematic. When dealing with unstable pixels, the errors do jump to a fixed offset regularly, and therefore form a chequered line down the sinogram as shown in Fig. 6.10. However, as the line is not a constant static error, the ring filter often fails. The best that can be done is to choose $m$ so that a balance is brought between the accuracy of the estimate and the ability to detect and filter errors which are static for only short periods of time. By experimentation, I found that a value of at least 100 was needed to guarantee that the estimate would not mistakenly remove valid data.

Let us consider the problem in a little more depth. In the sinogram every structure in the projection data forms a sine wave down the image. If only a single revolution was acquired, this means that there are two edges of the sinogram where the tangent goes down the image as with ring artefact lines. This means that for every structure, there is the potential at two points for valid data to be mistakenly identified as ring artefacts. If we were to set $m$ to be the full length of the sinogram, this would not be an issue (unless the volume were a concentric cylinder). By attempting to minimize $m$ we open up the possibility for blurring such edges.
A potential solution arose while considering what a sinogram is. To produce a sinogram we take a set of 2D projection images \((x, y)\) over \(\theta\) and transform the data so that the new images represent \((x, \theta)\).

However, there is nothing stopping us from taking the alternative transformation to \((y, \theta)\). In these conditions, the structures no longer form sine waves. In fact, only the top and bottom edges of the subject could potentially appear as lines to the ring filter. So now we have shifted the issue to the outer top and bottom surfaces of the subject. For the majority of scans, these surfaces are not the interesting parts of the volume.

However, while the new conditions are better we still haven’t lessened the problem. The natural next step is to consider a volumetric approach \((x, y, \theta)\). Ring artefacts still appear as lines down the \(\theta\) coordinate in the volume of exposures. Therefore, if we adjust the first step to consider a 2D region over \((x, y)\), we should be able to further reduce the risk of blurring.

Unfortunately, this is not as trivial as it sounds. Remember that the goal of
the first step is to separate the static error from the random error and underlying signal. The problem here is that simply taking the median of a 2D region is more likely to break our assumption that the neighbours have the same underlying signal. Fig. 6.11 illustrates the problem with a corner section. The median of the region would suggest that the target pixel belongs to the black region while a median of a 1D line correctly chooses the grey region.

The ideal approach would be to properly segment out the target area so that only neighbouring pixels belonging to the same object are chosen for consideration. However, this would undoubtedly result in a heavy performance hit. Even then, as projection images are filled with non-homogeneous structures and dead regions, clean segmentation may not be possible. In this regard, the original paper had the advantage by filtering post reconstruction.

The immediate solution which I adopted was a simple hybrid of the previous two approaches. The first step looks at a cross formed by a line across the $x$ axis and a line across the $y$ axis.

By using a set of 1D lines, the algorithm maintains the level of accuracy from the original sinogram based algorithm. The line across the $y$ axis helps prevent the
risk of classifying sine wave edges as rings. The line across the x axis helps protect the top and bottom surfaces of the volume. Fig. 6.12 shows results from the final algorithm on a small section of a scan. The improvements to the data are clear as all the lines are removed from the image.

To summarize, a ring filter was developed which uses two directional median filters to estimate the static error in a pixel and then correct for it. The original algorithm operated on the 2D sinograms. Later, this was shifted to the third perspective of the projection data \((y, \theta)\). Finally, a hybrid using the full volumetric data available was developed.

The resulting algorithm is still a two step process where the first step is a median filter over a cross shaped section, and the second median filter acts over a line across exposures. While slow, the algorithm can remove the majority of ring artefacts without blurring the data. The only remaining rings are due to unstable pixels which change state too fast for the ring filter to identify and correct.

### 6.5 Summary

The transition from MCTBuilder to mPPC came with a few major improvements. First, was the characterization of MARS exposure data. This led to thresholding starved and saturated detector elements as well as normalized data which veers too far from the expected bounds of \(0 < n \lesssim 1\). Also, a Poisson distribution test removes detector elements that are clearly not behaving according to the ideal Poisson distribution.

A dilation algorithm removes detector elements which are isolated within large dead regions as they can not be properly validated. The chance of these detector elements producing artefacts is high.

The inpainting algorithm was re-purposed so that it merely removes small dead regions. This improves the condition of the data while ensuring that the inpainting process is not injecting significant errors into the data.

The addition of a ring filter significantly improved the quality of the reconstructions. The algorithm uses two median filters to identify repeated static errors and then correct for them. The end result was a volumetric algorithm that considers all exposures to minimize the risk of mistakenly classifying valid structures as rings.

Lastly, with the adoption of DICOM the user interface of mPPC was simplified.
with the removal of the old directory hierarchy. Also, the presence of mART meant that the stitching and sinogram conversion algorithms could be removed. Together, this means that mPPC is better equipped for MARS datasets while being faster, and easier to use.
Chapter VII

Study of truncated datasets in mART

The work described in this chapter started in June 2013 and it presents the results of a study into salvaging truncated datasets without the need for interior reconstruction. In this case, truncation refers to an acquisition which does not cover the full span of the subject in all angles. The basic idea is to reconstruct the full volume despite the truncated measurements. Then, comparisons between the interior regions will reveal if the structures have still been faithfully recovered.

The motivation for this study is that the region of interest (ROI) selection tools in the scanning software were not comprehensive enough at the time to guarantee that the full volume was acquired. The true solution for this problem was to improve the ROI selection tools. However, in the meantime, this study resulted in a quick temporary solution which also made the reconstruction software more flexible.

7.1 Summary of the condition of the MARS system

This section provides an overview of relevant changes to the MARS system as of June 2013. In this chapter, the relevant components include the state of the scanner software, and the state of the reconstruction software.

7.1.1 MARS scanning software

By June 2013, CT10 had been completed. This scanner had a new design designated v5 or commercial release version one (CR1). The most important change was the adoption of slip-ring technology which allows free rotation of the gantry. This meant that continuous rotation could now operate at faster speeds to further reduce scanning times.

123
Other changes in the new design include; built in filtration, and adjustable collimators. The filtration offered various thicknesses of copper and aluminium to reduce the flux of low energy photons. The collimators could move during the scan allowing the beam to target the camera without irradiating the rest of the subject.

7.1.2 MARS reconstruction software

Although continuous rotation scans reconstructed successfully, the definition of a projection caused problems in mART. A projection was defined as a set of exposures with the same gantry rotation index. The index would guarantee that the angle for each exposure was identical. This definition ensured that the projection covered the full span of the subject at every step of the reconstruction. In other words, every voxel gets an update in every reconstruction step.

However, with the lack of synchronization in continuous rotation mode, there were cases where the same gantry rotation indices would correspond to completely different angles. This meant that the reconstructions often had holes in the projection which produced discontinuous updates to the volume.

Originally, this problem was expected to be resolved as synchronization techniques improved. However, a quicker solution was offered by redefining the definition of a projection. The new definition represents a projection as a binned set of exposures within an angular range. For example, given a bin size of 3 degrees, the second projection would be all exposures between 3 and 6 degrees. For convenience, 1 degree was set as the bin size for a total of 360 projections.

The binning guaranteed that each projection would completely cover the span of the subject. The scanning procedure aims to acquire 720 exposures per revolution. With continuous rotation this could vary down to below 600 for a camera position in the worst case scenario. A bin of 1 degree is large enough that no camera position will miss any angle. Therefore, the SART algorithm could proceed reconstructing full projections without discontinuities.

7.2 Motivation and methodology

For a significant number of scans, mistakes have been made which result in truncated acquisitions. In this case, truncation refers to the scan not completely covering the span of the subject. In most cases, this truncation only affects a small
Figure 7.1: An illustration showing the typical region of interest which is exposed by 360 degrees of coverage. The truncated region is inside the scanner, the further away from the center of rotation, the less angular coverage there is. Lastly, outside the scanner is no angular coverage at all.

Reconstructing truncated acquisitions the normal way does not work. Instead, it is an interior reconstruction problem which is much more complex than the standard exterior tomography. Standard reconstruction makes the assumption that the space between the x-ray source, detector, and the volume region of interest is zero valued. Obviously, this assumption is broken when reconstructing an interior region.

However, since the truncation is minor, it would be good if the data could still be salvaged, even if there is only a small portion inside the subject which is reconstructed correctly. The problem is that a scan cannot always be easily repeated, especially if the subject deteriorates quickly. Depending on the target of the study, being able to reconstruct even a small portion of the volume correctly will save the pre-clinical research teams a lot of time and trouble.

Let us consider the definition of a truncated acquisition. In a cone beam CT
scan rotating the full 360 degrees, every part of the subject will be exposed at some point. What we normally refer to as the region of interest, is the region which is exposed for the full 360 degrees of coverage. The moment a region has less than 360 degrees coverage, we refer to the dataset as truncated. Fig. 7.1 illustrates this concept.

The point here is that for the MARS datasets in question, the truncation is minor i.e. the regions still have a large amount of data available. Therefore, my idea was to simply reconstruct a larger region of interest as if the data was not truncated at all. The hope is that the inner region of the volume is reconstructed correctly, even if the outer structures are not.

So the experimental setup involved a single dataset with 16 camera positions which fully covers the span of the subject. Then, reconstructions would be performed with all 16 positions, then the 14 inner positions down to the 8 inner positions. In each case, the same volumetric region would be reconstructed. Then the difference image would be compared between the reconstruction of with 16 positions to each of the other cases. This should reveal what effects occur to the interior portions of the volume.

7.3 Results

The results from the study are shown in Fig. 7.2. The first observation that can be made about the interior portion of the volume is that structurally, they are identical. The outer regions of the volume are distorted as expected, but the interior regions appear perfect.

From the difference images we observe that there are some differences. This is especially prominent in the bottom row where the window/level is maximized to the interior region. However, appearances are deceiving. The maximum difference in the interior regions are less than 0.01 cm$^{-1}$ while the average difference is less than 0.001 cm$^{-1}$. This falls within the bounds for rounding errors given that the data is stored as integers and is rescaled.

The critical evidence of this is the difference between the ground truth with 16 camera positions, and the most severe test with 8 camera positions. We note here, that the difference image shows no structures in the interior regions at all. If the interior regions were distorted in a controlled fashion we would expect any
Figure 7.2: The results from the truncation test. From top left, the results show the reconstructions with 16, 14, 12, 10, and 8 camera positions. The second row shows the difference between the ground truth (16 camera positions) and the test result, 16-14, 16-12, 16-10, and 16-8. The third row shows the same difference images as the second but with a tight window/level to highlight the differences in the interior region.

distortion to be largest in this test. The fact that it is actually the closest to the ground truth suggests that the other results are due to rounding errors, rather than a measure of distortion.

7.4 Summary of truncation study

This study concludes that reconstructing the full subject under the conditions of slightly truncated data sufficiently salvages the interior component of the volume. The internal structures are well defined and the differences between the truncated reconstructions and the ground truth are small enough to be considered negligible.

As a study consisting of a single dataset, this result is not expected to be absolute. In fact, I expect that the usefulness of this result only extends to minor truncations of the data such as the mistakes made during some MARS scans.
Figure 7.3: The new GUI for mART which includes the option to explicitly set the volume radius, length, position and the voxel size. A button then uses the old procedure to provide default settings if desired.

definitely not recommended as a solution to interior tomography.

Another consequence of this study are a few changes to mART. The new GUI is shown in Fig. 7.3. The primary change is the way that the volume is defined for reconstruction. Before, the volume size and voxel size were calculated automatically, by searching through all the exposures. This procedure is still available as an option to provide default settings. However, mART now allows the user to explicitly set the volume radius, length, position, and the voxel size.

For truncated data, it means that the full subject can be reconstructed as performed in this study. However, it is also useful in the opposite case. When a small subject is scanned with a large region of interest there is no point in reconstructing the full region. By allowing the position and sizes to be changed explicitly, the bounds around the volume can be tightened. This makes the reconstruction more efficient as less voxels need to be reconstructed.

The other change to the mART GUI includes a preview for all three primary views; coronal, saggital, and axial. This provides users with a more complete view of the reconstruction progress.
Chapter VIII

Study of random sampling techniques for projection

The work described in this chapter started in February 2014 and it reports the results of a study into random sampling as an alternative projection scheme for reconstruction. In previous chapters, various methods for projecting between the volume and the exposures were discussed including raytracing, raycasting, and splatting. Random sampling offers a very simple and natural way of observing data that would otherwise require complex models.

During this time, two papers were published which included uploading MARS datasets for the wider scientific community. In the first case, Dr. Raja Aamir performed a study of imaging lamb meat to discriminate between bone, muscle, and fat [24]. I assisted the study by reviewing the image processing results and preparing the data for publication.

In the second case, Kishore Rajendran performed a study of imaging metal objects while observing the presence of beam hardening artefacts in energy bins of varying width [25]. I assisted with the image processing of the data ensuring that the best possible results were obtained. Lastly, I helped prepare the data for publication.

Also, during this time Christopher Bateman published his work on his material decomposition routine [98]. My contribution to his work was a validation of his use of my software, mPPC and mART. Both of us completed reconstructions of the data, which resulted in the same image quality. This was a quick test of how well other users operate my software. Lastly, I also helped review the final publication.

8.1 Summary of the condition of the MARS system

This section provides an overview of relevant changes to the MARS system as of February 2014. In this chapter, the relevant components include the state of the
scanner hardware, the state of the material decomposition software, and the state of the visualization software.

8.1.1 MARS scanner hardware

By October 2013, CT11 had been completed which had an identical design to CT10. In addition, new detectors with CdZnTe sensor crystals had arrived and had completed preliminary testing. Unfortunately, these cameras suffered from severe distortions which rendered them unusable. Unlike CdTe, these distortions clearly alter the geometry of the acquired data as shown in Fig. 8.1.

An issue arose where some acquired exposures would flash on occasion. This flash would either result in the acquired image being noticeably brighter or darker. The result of such cases is that the reconstruction would produce harsh streaks across the image. As the cause has yet to be identified, the only solution to this problem was to remove the bad exposures completely.

Fortunately, removing the offending exposures was a simple task. Using Image
Figure 8.2: A sinogram section of a set of flatfield exposures. The bright vertical line is a single exposure which flashed. The exposure should be removed before image processing begins.

J (a free scientific imaging tool) to view the sinogram quickly reveals the bad exposures as lines across the image. Fig. 8.2 shows a flashing exposure in a section of a sinogram from a flatfield scan.

8.1.2 MARS material decomposition software

A team had been formally created to investigate and develop a material reconstruction algorithm. This algorithm would combine reconstruction and material decomposition into a single process using the polychromatic form of the Beer-Lambert Law.

It is expected that a material reconstruction algorithm would perform much better than material decomposition. This is because the polychromatic form of the Beer-Lambert Law is a model which incorporates beam hardening as valid signal. Therefore, decomposition operates on data without distortion.

My role in this team is as a consultant. As my PhD was now approaching its end, I did not play an active role in the development itself. That was left to the team of mathematicians and physicists.

8.1.3 MARS visualization software

Alexander Chernoglazov and I added support for DICOM to MARS Vision. The benefits of this change is direct access to the DICOM tags. These tags provide measurements, patient information, and other useful data for display. The MARS DICOM data that MARS Vision received is the spectral CT DICOM IOD described in chapter 5.
Despite the experimental state of the reconstructed spectral CT DICOM IOD, there were no issues with the importing the data. The DICOM chain is now complete from the scanner through to visualization.

8.2 Motivation

In the near future, one of the options for the reconstruction algorithm is to incorporate octree structures. This would represent the volume with the largest voxels given the homogeneity of each region of the subject. As an octree would minimize the number of voxels, it means that the reconstruction will be much more efficient. Also, variable sized voxels is in essence a sparsity constraint which means that the quality should also be improved.

Ray-tracing through an octree is a slow process \cite{99}. The speed currently achieved with ray-tracing is due to exploiting the regular nature of the volume grid. In an octree structure, significant delays arise as the ray-tracing algorithm needs to traverse up and down the octree hierarchy as well as between voxels. Therefore, an alternative sampling scheme is needed for efficient traversal through the octree.

There were two motives for investigating random sampling. The potential for a performance boost is the first one, and the ability to naturally model complex ray geometries is another. While octrees are slow for traversing through a ray, direct sampling has almost no penalty. Therefore, random sampling should be suitable for any octree implementation.

Consider the problem of a cone beam, continuous CT scan, with finite source and detector element sizes. The source could be modelled as a disc and the detector element as a square or cuboid (if the volume is to be considered). The volume may be a regular or irregular grid (such as with an octree). Lastly, due to continuous rotation, the whole ray is rotated during the exposure.

Fig. 8.3 illustrates this scenario. Assuming no scatter, notice the complex geometry of the intersection between the ray and the volume.

In the given scenario, raytracing and raycasting could approximate this geometry by casting multiple lines over the ray. The more lines that are cast, the better the approximation. However, \( n \) rays would slow the process in the order of \( n \), not counting the fact that the irregular grid would further introduce delays.
Splatting could also represent the complex ray geometry. In this case, an analytical solution could be found with difficulty. However, it would most likely be faster to approximate the projected ray by splatting multiple sub-regions with simpler geometries e.g. a set of small cubes. Notice the common approach, that to approximate a complex system requires repeating a simple process many times over. Naturally, each additional step would slow down the algorithm.

In contrast, random sampling deals with any geometry in a very consistent and natural way. The complex ray geometry is broken down into its basic components. For the problem described above, the random sampling approach is as follows.

1. Find a random point in the source disc
2. Find a random point in the detector element
3. Rotate the points by a random angle within the exposure limits
4. Find a random point along the resulting ray inside the volume
5. Read the voxel value at the final point and perform the projection.
6. Repeat the previous steps.
As with the other sampling methods, a simple process is repeated many times over. However, as this done at the sampling level the delays introduced should be minor.

Another advantage of random sampling is the potential for minimizing the number of required samples. In the context of forward projection in CT reconstruction; raytracing, raycasting, and splatting all ask the question, “How much does voxel \( v \) contribute to the detector element \( d \)?”. This is achieved by sampling each and every voxel along a ray.

In contrast, random sampling asks the question, “How much of material \( m \) is present in the ray?”. This is achieved because the volume of a material in the ray determines the probability of being hit by a random sample.

Traditional sampling methods require each voxel to be sampled along a ray. In contrast, random sampling should only require sufficient samples to approximate all the materials in the ray. As volumes tend to have large homogeneous regions, this should require significantly fewer samples than with traditional sampling methods.

All in all, random sampling appears to be an alternative sampling scheme with high potential. It naturally deals with complex ray geometries, and complex volume structures, while potentially having better performance than the alternatives.

### 8.3 Methodology

This section covers the methods used to perform random sampling in mART. The algorithms included here are very simple but must be repeated many times to achieve an accurate result.

#### 8.3.1 Random sampling in forward-projection

The process for random sampling in the forward problem is slightly different from general sampling schemes. We want to estimate \( Ax \) from \( Ax = b \) without using \( A \) at all. In section 8.2, the basic sampling procedure was described to randomly locate a voxel inside the ray geometry. Afterwards, the value of the voxel must be accumulated to an ongoing estimate.

The first algorithm is the estimate of the transmission \( \tau \) along a line inside the ray. This is given in Eqn. 8.1. Here, \( \tau \) is the resulting estimate, \( M \) is the number
of samples along the line, $\mu_j$ is the voxel value of the $j$th random sample, and $d$ is the length of the ray.

$$\tau = \frac{d}{M} \sum_{j=0}^{M} \mu_j$$ \hspace{1cm} (8.1)

Interpreting the equation another way, we select a random line between the source and detector paths during the exposure. Following, we select a random sample on the line and assume that the value is homogeneous over the whole line. If we then take the average of a large number of samples, the average determines the size of the material over the ray.

For example, given 100 samples, if 15 of these hit calcium then we can say that 15% of the volume within the ray contains calcium. More samples simply means a more precise estimate.

In the scenario described before, we also consider the finite size of the detector, the x-ray source, as well as the continuous rotation. To model this would require a set of lines inside the ray. In this case, we must consider the accumulated counts instead of the transmission. The result is a modified Beer-Lambert law as described in Eqn. 8.2. Here, $C$ is the accumulated counts in the detector element. $S_i$ is the counts produced by the source along the $i^{th}$ line inside the ray. $N$ is the number of lines to simulate within the ray. Lastly, $d_i$ is now a measure of the length of each line inside the ray.

$$C = \sum_{i=0}^{N} S_i e^{-\frac{d_i}{M} \sum_{j=0}^{M} \mu_j}$$ \hspace{1cm} (8.2)

If we now use our flatfield estimate for the counts produced by the source we get the estimate from Eqn. 8.3 (assuming $\mu_{air} = 0$). For simplicity, let’s also assume that $S_i$ is constant over the size of the ray. This allows us to substitute the flatfield counts into the algorithm. Therefore, the final estimate is shown in Eqn. 8.4.

$$C_f = \sum_{i=0}^{N} S_i$$ \hspace{1cm} (8.3)

$$\frac{C}{C_f} = \frac{1}{N} \sum_{i=0}^{N} e^{-\frac{d_i}{M} \sum_{j=0}^{M} \mu_j}$$ \hspace{1cm} (8.4)
8.3.2 Random sampling in back-projection

Back-projection is a very different problem. In forward projection we want an estimate of the photons that accumulate at a detector element from the source. It should be a good model of the physical scan. With back-projection we simply wish to adjust each voxel such that the volume is closer to the ground truth. Note that in this case, all voxels need to be updated or we end up introducing noise into the reconstruction.

This means that the best approach is to adopt a voxel driven method. For each voxel, randomly sampled points are projected onto the exposures. This guarantees that all voxels will be sampled at least once. However, it is expected that the back-projection will be significantly slower then forward projection. The update to the voxel is described in Eqn. 8.5.

\[ x_{k+1} = x_k + \frac{1}{N} \sum_{i=0}^{N} \delta \tau_i d_i \] (8.5)

Here \( x_k \) and \( x_{k+1} \) are the current and next voxel values respectively, \( N \) is the number of samples, \( \delta \tau_i \) is the correction factor, and \( d_i \) is the length of the ray. Note that this is a simplified form of the SART equation, which uses a weighted average.

The main point is that the contribution of a detector element is determined by the probability of being hit by a randomly selected line in the ray. While only one sample is required to guarantee an update per voxel, more samples mean a more precise estimate of the contribution of each detector element. Given that each detector element is corrupted by noise, a good aim would be to sample at least one point per detector element in the ray.

8.4 Results

The results presented here cover a range of implementations to give a broad overview of their performance. These implementations include, pure raytracing, pure random sampling, pure random sampling using a lookup table, and a hybrid approach with random sampling for forward projection and raytracing for back projection.

The implementation of the random sampling assumed a point x-ray source and
Figure 8.4: Slice 29 from 1000 steps of a reconstruction with raytraced rays for forward and backward projection. The image on the right is the 1000th update. Notice the subtle striped pattern orthogonal to the direction of the projection in the update image.

...a square detector element. The motion of the gantry was not considered in this test. Therefore, the ray geometry is a pyramid in this case. Sampling occurs randomly within this pyramid.

All reconstructions were performed on the same dataset. This was a human excised atheromatous plaque with sample tubes containing reference solutions for calcium, fat, and water. Each reconstruction proceeded for 1000 steps so that the reconstruction time could be accurately discerned.

Also, the update to a slice of the volume at the 1000th step was extracted. This provides a perspective on the effect of the sampling process to the reconstruction. When a reconstruction has reached convergence, the update images should represent the noise in the measurements. This is because all geometric structures should already have been added to the volume.

The first result is the pure raytraced reconstruction in Fig. 8.4. We can see a reasonable result, but interestingly, the update shows a striped pattern orthogonal to the direction of the projection. We expect stripes along the projection but not across. This is the result of aliasing artefacts from approximating the ray as a thin line. It is expected that any foreign structures such as these will contribute some
Figure 8.5: Slice 29 from 1000 steps of a reconstruction with randomly sampled rays for forward and backward projection together with its 1000th update. Notice that no unexpected patterns are visible in the update image on the right.

From Tab. 8.1 the average time taken per step is 2.076 s seconds. Note that for each step there is an average of $128 \times 128 \times 70$ rays being cast over a volume with $160 \times 160 \times 59 \times 4$ voxels. 2 seconds is a reasonable reconstruction time for a non-optimized, CPU based implementation.

The second result is the pure random sampling shown in Fig. 8.5. This reconstruction provided the highest quality with the lowest noise and fewest rings. We can immediately note that the update shows no unexpected patterns over the projection. This demonstrates that the sampling process is not contributing errors to the reconstruction.

From Tab. 8.1 the average time taken per step is almost 3.76 s which is almost 2 times slower than with ray tracing. However, note that the requirements for a pure random sampled reconstruction are not ideal. Back-projection requires significantly more samples as all voxels need an update. The number of samples for the forward projection was determined by estimating the number of voxels along the ray and dividing by two. This was an arbitrary choice used for preliminary investigation. The idea being that if the results are sufficiently good, a more optimized system
Figure 8.6: Slice 29 from 1000 steps of a reconstruction with sampling from three tables of 65535 random numbers for forward and backward projection together with its 1000th update. Notice the strong curved patterns which have a clear impact on the quality of the reconstruction.

could be determined later.

Another issue with the pure random sampling is the choice of the psuedo random number generator. The Mersenne Twister [100] was the generator of choice but it is not the fastest. The use of the Xorshift [101] generator improved the timing to 3.6 s, which is a minor improvement.

It seemed that the number of calculations in the inner most loop of the reconstruction has a significant effect on performance. Therefore, an alternative was to adopt a lookup table of random numbers.

Three lookup tables were created containing 65535 random numbers each. There are two lookup tables to randomly sample the point inside the 2D detector element. A third lookup table then samples the ray.

The choice of 65535 numbers favours speed. It exploits the overflow of an unsigned short construct. When an unsigned short exceeds 65535, it automatically returns to 0. Therefore, the numbers loop automatically without any need for checking conditions in the inner loop of the reconstruction code.

The results in Fig. 8.6 show the outcome of using the lookup table. Notice the strong curved patterns in the update image and the corresponding bad quality of
Figure 8.7: Slice 29 from 1000 steps of a reconstruction with hybrid sampling together with its 1000th update. The quality is equivalent to the fully raytraced example.

the reconstructed image.

What this means is that more random numbers are needed. A large table should give quality equal to the pure random sampling case. However, the average time taken for each step is 2.581 s, which is still significantly lower than pure raycasting. Increasing the size of the lookup tables will remove the unsigned short exploit and make the process even slower. Therefore, further tests were not performed using lookup tables.

The final test was to take the best of both worlds, random sampling for the forward projection and raytracing for the back projection. This would allow us to exploit the fewer samples required for forward projection while avoiding the slow voxel driven method for the back projection.

Remember that voxel driven methods are crippled with the current state of the MARS data. Since the exposures are not stitched together into single projection images, a voxel driven method has to determine which exposure is relevant per voxel. The delay introduced by this is substantial.

Fig. 8.7 shows that the quality of this test is equivalent to the pure raytracing. The striped pattern across the update image is visible here as in the pure raytracing.
<table>
<thead>
<tr>
<th>Reconstruction type</th>
<th>Average time per step (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raytraced rays</td>
<td>2.076</td>
</tr>
<tr>
<td>Randomly sampled</td>
<td>3.76</td>
</tr>
<tr>
<td>Randomly tabled</td>
<td>2.58</td>
</tr>
<tr>
<td>Hybrid</td>
<td>1.98</td>
</tr>
</tbody>
</table>

Table 8.1: Average time taken per step for each reconstruction.

case. However, from Tab. 8.1, the average step time of 1.979 s is the fastest result obtain, although only slightly. This tells us is that the observed delays in the previous cases were a result of requiring a voxel driven method.

8.5 Conclusion

The investigation presented in this chapter was a feasibility study for using random sampling techniques to improve performance and quality. The main goal was to get a feel for the behaviour of random sampling techniques to determine if further investigation is warranted.

My conclusion from the results shown, is that random sampling techniques may be useful in the future if octree based reconstruction is explored. However, there is little advantage for random sampling in the current implementation of mART.

The true random sampling case did produce the best image. However, the changes were subtle and do not justify the almost twofold increase in reconstruction time. The use of a lookup table does not improve the performance to an acceptable level and also offers the poorest quality.

The hybrid solution presents an interesting case with a slight advantage in speed. However, the 0.01 s advantage translates to 1 minute every 6000 steps. The solution will have converged by then, so 1 minute is negligible. In fact, differences in speed as minor as this could be attributed to the state of the computer at the time e.g. background application activity.

The quality in the hybrid case appears identical to the raytracing case. The orthogonal striped patterns in the update images are the same as well. This means

---

Note that the rate of convergence is not affected by the choice of the projection routine. The differences between algorithms manifest as noise around the same average value. Therefore, an increase in the average time per step is an increase in the total reconstruction time.
that there is no advantage in using the hybrid technique against pure raytracing.

To summarize, random sampling will be put aside for now. It is expected that an octree solution will be investigated for reconstruction in the near future. At that time, random sampling techniques can be revisited as the penalty to raytracing should balance out the results obtained.

A final note is in the other lessons learned from this study. The update images shown demonstrate the fact that the reconstruction algorithm can easily introduce artefacts. The quality of each image appeared to be directly correlated to observed structures and defects in the update images. Therefore, the update images could be a useful guide in verifying the achievable quality of a reconstruction algorithm.
Chapter IX

Steps towards automation

The work described in this chapter started in May 2014 and it details the steps completed towards automating the data processing chain. As the only manual processes in the current data processing chain are in mPPC, this work essentially aims to render mPPC obsolete.

Another motive to remove mPPC is to simplify the work flow for the users. Two applications means that users need to participate twice in producing reconstructed images.

A third motive is reducing the volume of data produced. Reconstructing directly from raw data, means that there is no requirement to save intermediary steps. As datasets can be in excess of 20GB, the savings are significant enough to warrant action.

While I was not completely successful in removing mPPC, the improvements shown in this chapter are a good step towards that goal.

The primary contribution of this chapter is the investigation into adapting the relaxation factor and weighting average components of SART to target the characteristics of MARS datasets. In this way, the quality of the reconstructed datasets can improve while reducing the burden of filtration on the pre-processing chain.

During this time, Kishore Rajendran was working on a denoising technique which exploits the structural similarities between energy bins. The algorithm uses a frequency-split to separate the structures from the energy signal. The structures are then cleaned in the higher energy bins, recombined and converted back to a better representation of the original image.

I involved myself with his work as a consultant, ensuring that the denoising process preserves the edges and energy signal in the datasets. In the near future we expect to publish this work in Medical Physics [102].
9.1 Summary of the condition of the MARS system

This section provides an overview of relevant changes to the MARS system as of May 2014. In this chapter, the relevant components include the state of the scanner hardware, the state of the scanning software, the state of the pre-reconstruction processing software, and the state of the visualization software.

9.1.1 MARS scanner hardware

Dr. Michael Walsh managed to solve the detector flashing issue. The problem was related to loose wiring and not wear and tear of the detector as was originally thought. This meant that the problem was easily fixed, and no more corrupted exposures have since been observed.

9.1.2 MARS scanning software

The charge summing mode produces four adjustable charge-summed energy bins. However, the remaining counters can still be used. This gives an additional three energy bins without charge summing mode, and an arbitration counter. The arbitration counter counts every detectable photon above the electronic noise level, and can be considered a charged summed energy bin set at this minimum energy.

In previous versions, only the four adjustable charge summing mode energy bins were stored. The current version of the scanning software stores all the energy bins.

9.1.3 MARS pre-reconstruction processing software

The consequences of the changes in the scanning software led to some issues in mPPC. The acquired energy bins from the MARS scanner can be referred to as low threshold bins. This means that a low threshold is set, and all detectable photons above the threshold are counted. This means that the energy range overlaps between bins.

mPPC has a minor tool which subtracts energy bins to get non-overlapping, narrow energy bins. This tool relied on the energy bins being of the same type, and ordered according to the low threshold energy.

Now, datasets were saved with three different types of energy bins which are scrambled in order. This means that the subtraction of energy bins is no longer
correct. Currently, I’m waiting for the scanning software to add DICOM tags which provide the details of the energy bin type and order. This will allow mPPC to produce the correct data once more.

9.1.4 MARS visualization software

Alexander Chernoglazov implemented some measurement tools into MARS Vision as per request by a research team from Lincoln university. This team acquires data from a GE Pro Speed scanner on their campus. As their data is not from a MARS system, this was a useful test between importing commercial CT data, and the spectral CT IODs now used in the MARS system.

The outcome of the testing was that MARS Vision can load both cases with the same code. In other words, no MARS specific code is required to import and view MARS data compared with commercial CT datasets. This is good validation of the design of the spectral CT IOD in chapter 5.

9.2 Manual tasks and automated tasks

In the current pre-reconstruction processing chain there are two crucial tasks which prevent reconstruction from raw data in counts. The first is the task to eliminate dead pixels. This includes testing to see if the distribution is not Poisson, eliminating starved and saturated detector elements, and most importantly the custom user masking.

The second is the task to filter rings. In its current form, the reconstruction algorithm produces highly corrupted reconstructions if the ring filter is not applied.

In an ideal reconstruction algorithm, the majority of the tasks can be a part of the equation. In a fall-back scheme, some tasks could become part of an automated pre-reconstruction processing step. However, purely manual steps such as the custom user masking need to be eliminated. To this end we can summarize each step of the pre-reconstruction processing chain.

9.2.1 Custom user masking

This task needs to be eliminated. As detector element behaviour is not scan specific, identifying bad and dead detector elements should be a part of the scanner
calibration process. Still, smart algorithms to detect the likelihood of misbehaving
detector elements could suppress their effects in the reconstruction.

9.2.2 Darkfield masking

The only manual part of this task is the selection of the darkfield. Some new
DICOM tags could be implemented which reference the unique identifiers of the
darkfield. This is all that is needed to locate the darkfield on a PACS. So in the
near future, this can become fully automatic. For now, selecting a darkfield before
starting reconstruction is sufficient.

An alternative is for the darkfield masking to be performed on the scanner
itself. If the scanner develops methods to detect bad and dead detector elements,
a mask can be created. Such a mask would render both the custom user masking
and the darkfield masking obsolete. The scanner could then apply the mask to
the dataset directly or pass on the mask to the reconstruction in the same way
proposed earlier with new DICOM identification tags.

9.2.3 Energy bin subtraction

This task was always a temporary measure. In the future, the material recon-
struction algorithm will use the polychromatic form of the Beer-Lambert Law. In
that case, energy bin subtraction becomes obsolete. The only advantage of nar-
row energy bins is improving the conditions of the data for the monochromatic
approximation of the Beer-Lambert Law.

In the meantime, energy bin subtraction has no parameters and can be easily
automated.

9.2.4 Flatfield normalization

As with darkfield masking, some new tags referencing the unique identifiers of
the flatfield exposures is all that is needed to make flatfield normalization fully
automatic. As an intermediate step, mART can simply request that the user
locates a directory containing flatfield exposures.
9.2.5 Poisson distribution test

In mPPC, this is a manual process where the user defines a percentage threshold. If the distance from the Poisson distribution exceeds the threshold, the detector element is removed.

In the reconstruction process, this could be part of a confidence test. In the following sections, a confidence test is introduced which gives detector elements different levels of influence over the reconstruction. If the Poisson distribution test becomes part of the confidence calculations, then there is no longer any need to include this in mPPC.

9.2.6 Dilation

This task was included to remove isolated detector elements within large dead regions. The motivation being that without neighbouring data the quality of the detector element is hard to ascertain. Also, operations such as ring filtration cannot operate on isolated detector elements.

In terms of its use, the dilation task often uses the same parameters for the majority of scans. Therefore, this process could be automated if the common values are hard coded as constants. However, in the ideal case, dilation should not be required.

9.2.7 Inpainting

This task aids the ring filtration and reconstruction process by eliminating small dead regions. This makes the data more continuous which results in better conditions for filtration and reconstruction.

As with the dilation task, the inpainting also uses the same parameters for the majority of scans. Therefore, if the common values are hard coded as defaults, the inpainting also becomes automatic.

9.2.8 Ring Filtration

This task is the most time consuming of all the pre-reconstruction processing tasks. Ideally, this task would become obsolete. A fall back plan would be to include this task as an automated step before or after reconstruction. The best scenario would
be a reconstruction algorithm with very low sensitivity to noise, static errors, and outliers.

9.2.9 Summary of tasks

As mentioned in the previous task summaries, almost all the tasks can be automated. In the ideal case, most of the tasks would become obsolete. The only task which cannot be automated within the image processing chain is the custom user masking. However, improved detection of bad detector elements on the scanner could render masking obsolete.

9.3 Relaxation factor and weighting factor

The ideal reconstruction algorithm would be able to reconstruct high quality images from raw MARS data. The first investigation to this end looks at the relaxation factor and the weighted average. To this end, refer to the common form of the SART algorithm in Eqn. 9.1.

\[
x_{i,k+1} = x_{i,k} + \lambda \frac{\sum_j a_{i,j} b_j - a^+_{i,j} x_k}{\sum_j a_{i,j}}
\]  

(9.1)

In this expression \( \lambda \) is the relaxation factor, \( \vec{x} \) is the volume, \( \vec{b} \) is the projection data, \( A \) is matrix relating \( \vec{x} \) and \( \vec{b} \), \( k \) is the iteration, and \( i \) and \( j \) are the indices for the volume voxel and projection pixel respectively. Also, by inspection we can see that SART is simply a weighted average of a set of correction terms. In fact, we can rewrite the expression as in Eqn. 9.2. Here, \( w_{i,j} \) is the weight introduced to the correction term \( c_{i,k} \).

\[
x_{i,k+1} = x_{i,k} + \lambda \frac{\sum_j w_{i,j} c_{i,k}}{\sum_j w_{i,j}}
\]  

(9.2)

This form of the SART expression brings up an interesting point. Commonly, the weight is simply a value from the matrix \( A \). If rays are modelled as lines, and the values \( a_{i,j} \) are the length of the line segments in a voxel, then the weighted average prioritizes rays that pass through bigger portions of the voxel. However, this is not a requirement.
As a test, let us consider the effect of treating $w_{i,j} = 1$. In other words, a simple mean of all the correction terms that contribute to the voxel. Fig. 9.1 compares the effects on the first step of a reconstruction between $w_{i,j} = a_{i,j}$ and $w_{i,j} = 1$. As expected, the difference image reveals that the only effect is minor aliasing at boundaries.

The question then becomes, what if the weighting factor were chosen based on more principled constraints, such as a confidence level in the validity of the measured data? In that way, we should be able to minimize the influence of noisy detector elements and strengthen the influence of detector elements with ideal behaviour. A similar idea was published by Levakhina et al. for digital tomosynthesis [103].

The alternative to the weighted average is the relaxation factor. In literature, this is often treated as a global constant. In other cases, it has been used to apply windowing functions to the reconstruction. Anderson and Kak [56] applied a Hamming window so that the center of the volume took priority over the outer edges. Other cases would base the relaxation factor on the number of steps completed so that the influence of the measured data decreases over time.

The effects of the relaxation factor is both similar and opposite to the effects of the weighted average. The relaxation factor affects the magnitude of the correction term. This has consequences for both the image quality and the convergence rate.
A low relaxation factor tightens the noise levels as each step has less effect on the volume. This produces smooth looking images. However, this comes at the cost of slower convergence as more steps are required to reach the final value.

In contrast, the weighted average does not affect the magnitude of the correction term. If only a single correction factor contributes to a voxel, and the confidence is not zero, the weighted average does not change.

Therefore, the effectiveness of the weighted average is reliant on the number of correction terms that influence the voxel. The more terms, the more likely that trustworthy correction terms can dominate the result.

Either way, the magnitude of the correction terms are preserved. This means that the weighted average is appropriate for ranking detector elements against each other.

To summarize, the SART algorithm can be rewritten to use a generic weighted average based on more principled constraints. Such constraints could a confidence level in the measured data. As a weighted average, this will not affect the rate of convergence as the magnitude of correction terms are preserved.

The relaxation factor controls the overall magnitude of the correction terms. In this way, the influence of one projection may be adjusted compared with another. Reducing the relaxation factor improves quality at the cost of the rate of convergence.

Together, the relaxation factor and the weighted average should allow reconstructions to draw out the best data from the acquired exposures.

### 9.4 Poisson distribution test

The first goal was to integrate the Poisson distribution test into the reconstruction process. This test determines how likely a given set of samples is not a Poisson distribution. As it grades detector elements, the test can naturally form a part of the weighted average. In this way, the detector elements which are not ideal will have less influence over the end result.

\[
\text{trust} = \begin{cases} 
\frac{-10}{e} \left( 1 - \frac{\text{variance}}{\text{average}} \right) : \text{variance} \leq \text{average} \\
\frac{-10}{e} \left( 1 - \frac{\text{average}}{\text{variance}} \right) : \text{otherwise}
\end{cases}
\]  

(9.3)
Figure 9.2: Slice 71 of a phantom dataset which has a high number of outliers corrupting the image. The left image has no additional weighting while the right image adopts the Poisson distribution test to weight each detector element. Notice that a good portion of the outliers have been removed with the Poisson distribution test. Lastly, note that the dark streaks are due to dead regions in the detector which will be filled in over time.

The final equation used is given in Eqn. 9.3. Note that this is the same as the Poisson distribution test, except that the test result is skewed as an exponential decay. This allows the influence to quickly weaken in a controlled manner as the detector elements stray from the ideal. Before the measure is applied to the weighted average, a weak threshold of 0.0001 eliminates outliers.

Using a phantom dataset, I compared the results of reconstructing a single step directly from raw data. The first case had the default weighting factor based on the path length, while the second added the Poisson distribution test. Fig. 9.2 shows the results obtained in a particularly bad slice of the phantom.

Clearly, a good number of outliers were removed, although not all were eliminated unfortunately. Given that the Poisson distribution test was not sufficient to clean data in mPPC, the results were as expected. More tests to determine the detector element confidence are required.
9.5 Targeting outliers

The next step was to target the remaining outliers in the acquired exposures. One of the features of an outlier is that it is typically a local minimum or a local maximum. This means that reducing the influence of local minima and maxima should help eliminate the rest of the outliers.

\[
\text{trust} = \begin{cases} 
  e^{-10 \left(1 - \frac{\text{median}}{\text{average}}\right)} & : \text{median} \leq \text{average} \\
  e^{-10 \left(1 - \frac{\text{average}}{\text{median}}\right)} & : \text{otherwise}
\end{cases} \tag{9.4}
\]

The algorithm adopted here is given in Eqn. 9.4. The algorithm uses the difference to the median of a neighbourhood to estimate the error in the same way as the first step of the ring filter. In fact, as with the ring filter, the neighbourhood is a cross shaped section with a radius of 5 (window size of 11). In this case, the radius is less than in the ring filter as the outlier test does not compare exposures to refine the estimate of the error.

Again, a weak threshold of 0.0001 eliminates the worst offending outliers before the test is applied to the weighted average. The process is the same as with the Poisson distribution test.

The biggest question for this step is when the test should be performed. We could perform this test in a pre-reconstruction step. The test could also be performed as each exposure is loaded during reconstruction. Lastly, the test could be performed on the error image \((\vec{b} - \vec{A}\vec{x})\) just before back-projection.

The advantage of the pre-reconstruction testing is that we will get an indication of the history of each detector element. A detector element which constantly produces outliers will quickly lose confidence.

However, this could also be problematic. Even if a detector element does not produce an outlier, a weighting factor will be given. Therefore, a detector element which is mostly good but has a few isolated incidents will unfairly lose confidence for all the exposures in the scan.

This problem would be solved if we test the exposures as they are loaded during reconstruction. Each exposure is treated independently so times when a detector element misbehaves will not negatively impact other exposures.
Figure 9.3: Slice 71 of a phantom dataset which has a high number of outliers corrupting the image. The left image only has the Poisson distribution test applied while the right image also adopts the outlier test. Notice that all the outliers have now been removed with the outlier test. The dark streaks are due to dead regions in the detector which will be filled in over time.

However, testing exposure values directly has same problem as the ring filter. The underlying signal needs to be distinguished from the errors. Comparing a detector element to a local median is a rough approximation with a high risk of losing valuable signal. Multiple median values from a range of kernel sizes could refine the approximation. However, this would also induce a significant penalty to the performance of the reconstruction.

This leaves the final case which tests the error image. At the start of the reconstruction, the volume is initialized to zero. Therefore, testing the error image is the same as the original exposure image.

However, as the reconstruction progresses, the error images are reduced to pure noise as the signal is deposited into the reconstructed volume. This means that at a later stage in the reconstruction, comparing a detector element to a local median has almost no risk of mistaking signal for noise. Clearly, these are the best conditions for identifying outliers.

The only remaining question is what units the data should be for the compar-
ison. I chose the intensity ratio $\frac{I}{I_0}$ because this guarantees that the values are positive and within a well defined bound. This behaves nicely with the adopted algorithm as the division component would have difficulty with negative numbers.

In this case, the same experiment was performed as with the Poisson distribution test. The same phantom was reconstructed for a single step to determine its effects. As can be seen in Fig. 9.3, the remaining outliers were removed. The only discontinuities in the image are the result of dead regions in the detector.

### 9.6 Blending factor

In past reconstructions, even with the ring filter some harsh rings were still visible. I ran an ongoing investigation to find the cause, but had been unsuccessful until now. A unique property of these rings was that they were not isolated. The rings were more like hollow cylindrical sections as illustrated in Fig. 9.4.

There are no bad rows or columns in the detectors so I concluded that these rings were due to mismatching levels between overlapping exposures. Consider a detector which has an out-of-plane tilt. The geometric model ignores such cases due to the difficulty of measuring the magnitude and direction at the same time. It was left to careful installation to eliminate any tilts in the system.

However, if a tilt is present and the detector is translated, it means that the region where the exposures overlap is unbalanced. The edge of one detector is closer than the opposite edge of the other detector which means that there will
be sudden jump in the acquired values. If the exposures were stitched together, this imbalance would be reduced by the ring filter. Since the exposures are not stitched, the imbalance remains in the acquired data.

The correct solution would be to determine all tilts in the geometry and account for them in the geometric model. Even better, a more rigid design could eliminate the possibility of tilts altogether. However, at this point and time both are impractical to achieve.

Therefore, the next addition to the weighting factor of SART is a blending function. This is a simple function which prioritizes detector elements at the center of the detector. This means the imbalance is smoothed over the overlapping portions of the exposures. The algorithm used is given in Eqn. 9.5. Here, $x$ and $y$ are the position of the detector element relative the detector center in mm while 7.04 is half the width of a Medipix detector.

$$trust = (7.04 - |x|) (7.04 - |y|)$$  \hspace{1cm} (9.5)

Another phantom dataset was used to test the results of this addition. In this
case, the volume is reconstructed for 3500 steps to guarantee convergence of the volume and any rings. Fig. 9.5 shows the reconstructions obtained. The blending function applied to the right image suppressed the rings which were caused by an imbalance between exposures.

9.7 Summary

The main contributions in this chapter are three additional components to the weighted average of the SART equation. Together, these additions assign a level of confidence in the detector element. This means that detector elements with more confidence have greater influence over the reconstruction. Naturally, this only has an effect if the voxel in question has more than one correction term influencing its next value.

The first confidence term is a restructuring of the Poisson distribution test. This term reduces confidence in detector elements which are clearly behaving differently from the ideal.

The second confidence term attempts to remove any remaining outliers. It achieves this by targeting detector elements whose values are too different from its neighbours. The greater the difference, the more likely the detector element is an outlier, and the less confidence is given.

The outlier test operates on the error image. This is useful because the error image is reduced to noise as the reconstruction progresses. This means that the accuracy of the test improves over time.

In both the Poisson distribution test and the outlier test, a weak threshold is applied to remove the most extreme cases. This is to guarantee that detector elements with no valid signal are completely removed from the reconstruction process.

The final confidence term is a blending function. This term prioritizes detector elements in the center of the detector. This is to reduce the imbalance between overlapping exposures.

The results for the three additions demonstrate that they are effective in removing the worst offenders in the reconstruction i.e. streaks and harsh rings. However, as can be seen in Fig. 9.5, these tests do not remove rings completely. Low magnitude errors still persist, and generate artefacts in the reconstruction.
Therefore, these three tests are not sufficient to completely remove the need for mPPC. Further improvements are needed to make the solution fully robust against all sources of error.
Chapter X

Conclusion

Over the scope of my PhD, three applications were developed: mPPC, mART, and MARS Transfer. Together, these applications provided members of the MARS team and external partners, the ability to reconstruct spectral CT datasets from the MARS molecular imaging system.

Over the evolution of the MARS system and the underlying technology, I maintained and adapted the algorithms in mPPC, and mART, to maintain a high standard of quality for the user groups. In this way I supported and enabled both the ongoing development of the MARS system, as well as various pre-clinical studies that have advanced using MARS technology.

A wide variety of studies were conducted to facilitate the creation of mPPC, mART, and MARS Transfer. These studies are multi-disciplinary and range from characterizing specific Medipix detectors in collaboration with the scanner development team, to developing image processing algorithms for the image processing, to directly supporting pre-clinical research conducted by various teams in the MARS group.

The chapters of this thesis chronologically outline the work I performed along with the ongoing changes in the MARS project which motivated and directed the research and development in my PhD. The primary research questions addressed in these chapters are summarized through the following.

• Chapter 3 - How can we deal with thermal sensitivity in Medipix 3.0?

• Chapter 4 - How should I model the MARS prototype geometry for projection stitching and reconstruction?

• Chapter 5 - How should DICOM data structures be implemented for MARS spectral CT datasets?
• Chapter 6 - What kind of signals do the various Medipix detector produce and how should they be dealt with?

• Chapter 6 - What is an appropriate algorithm for ring filtration?

• Chapter 7 - How can unintentionally truncated scans be salvaged for qualitative analysis?

• Chapter 8 - Are random sampling techniques effective for reconstruction?

• Chapter 9 - How can the quality and performance of basic iterative reconstruction techniques be improved?

Alongside these research questions, the academic contributions section lists the publications made in collaboration with other MARS teams. These papers are also mentioned throughout the chapters in this thesis.

10.1 Achievements over the PhD timeline

The work achieved over the course of my PhD can be divided into overlapping tasks. Each task represents a chapter in my thesis, and contributed to one or more applications for the purpose of meeting the needs of the user groups.

10.1.1 Creation of MCTBuilder

The first task in my PhD was the creation of MCTBuilder, a unified pre-reconstruction processing application. At the start of my PhD, the pre-reconstruction processing was performed using sets of scripts written in Matlab, or Python. As different versions were scattered around the MARS team, it was difficult to keep up to date with the latest scripts to produce the best quality reconstructions.

Therefore, I developed MCTBuilder as a port of a set of pre-reconstruction processing scripts written by Dr. Paul Ronaldson. These scripts helped him achieve his impressive results in his thesis [35]. His scripts contained algorithms for darkfield masking, flatfield normalization, stitching, inpainting, and sinogram conversion. MCTBuilder, took these scripts to create a single user friendly application for the MARS team.
During this time period, studies of the behaviour of Medipix 3.0 revealed a temperature dependence which caused acquired values to drift. Therefore, I altered the flatfield normalization algorithm so that multiple sets of flatfield exposures were interpolated over the progress of the scan. This effectively compensated for drift in scans of moderate length.

To support MCTBuilder, I designed a new packaging scheme for the MARS data in conjunction with Dr. Michael Walsh. The new scheme packaged image data in a directory hierarchy so that the images were pre-sorted into categories for easy navigation. The scheme also contained the scanner configuration files along with a scan report.

This unified the import and export between the scanning software and MCTBuilder. Also, the scan report and configuration files allowed the scanner to inform MCTBuilder of the scan procedure rather than the estimations that the scripts made before.

Finally, MCTBuilder provided a simple GUI which presented an explorer view to navigate through data, a wizard to tweak the algorithm parameters, and a preview that shows a live update of the algorithms applied to an image. Once the algorithms were set up, a batch process would apply the pre-reconstruction processing chain to a set of images.

Once completed, MCTBuilder was released to the MARS team and to our external partners including the Mayo Clinic and Virginia Tech. With the addition of the new scanning software created by Dr. Michael Walsh, there were no longer any dependencies on scripts in the MARS system.

10.1.2 Creation of mART

The second task was the creation of mART, a reconstruction application which replaced Octopus CT. The motives for creating mART were two-fold. Firstly, we wanted a flexible in-house reconstruction platform which could be used for developing and testing reconstruction algorithms.

Secondly, due to the increased use of a dual CdTe MXR camera, the assumptions and requirements of the previous image processing chain were broken. In particular, the requirement for inpainting all dead detector elements was problematic given the size of dead regions in the dual CdTe MXR camera.
Therefore, mART was created with three goals in mind. Firstly, it should at least match the quality from Octopus CT. Secondly, it should be flexible enough to reconstruct all MARS datasets irrespective of the quality of the camera. Finally, mART should be created as soon as possible as there is already a waiting list for the dual CdTe MXR camera.

After a brief review of CT reconstruction techniques, I decided to implement SART. This was a simple algorithm which met all the goals for mART.

To support the implementation, I first built a geometric model of the MARS scanner using 4D translation and rotation matrices. Translations corresponded to physical translations, or mounting positions. Rotation corresponded to physical rotations, or mounting skews and tilts.

The next step towards implementing mART, was the projection routine. I chose to use raytracing due to the precision with which it operates. This would minimize sources of error in the first revisions of mART.

Using the geometric model and the projection routine, mART version 1 implemented the Kaczmarz solution instead of SART. This is a simpler, quicker algorithm to implement which was useful for testing the components that made up mART. Reconstructions with a simulated Shepp Logan phantom and a MARS scanned phantom showed that the geometric model and projection routine were operating as expected.

The first public release of mART was version 2 which completed the SART implementation. An extension was added to SART so that the results from the projection routine was shared between energy bins. This was possible as the energy bins are acquired simultaneously and have identical geometry. This resulted in a speed up of $n$ times for $n$ energy bins.

The final tests of mART compared image quality with Octopus CT under various conditions. Also, mART was trialled as part of a pre-clinical study of human excised atheromatous plaques. The results revealed that mART met the initial design goals.

10.1.3 Integration of DICOM

The third task was the adoption of the DICOM standard which offered protocols to package, backup, and transfer MARS datasets in a unified way. Also, as DICOM
is a modern clinical standard, it means that the MARS molecular imaging system will be compatible with clinical infrastructures and data formats.

Spectral CT is a new modality. This means that DICOM has not yet standardized an IOD for packaging spectral CT data. Therefore, I had to develop a MARS DICOM IOD for reconstructed data.

Also, the MARS system operates in a modular fashion where the image processing is performed on a workstation external to the scanner. This meant that we also wanted a MARS DICOM IOD to represent the raw exposure data that the MARS scanner produces. Lastly, the image processing is broken down into two applications; MCTBuilder, and mART. This means that we also wanted to represent the normalized exposure data using DICOM.

This led me to create two MARS DICOM IODs. The first represents exposure data, which could be darkfield, flatfield, scan, or normalized exposures. The energy bins are packed together as multiple frames using the existing NM multi-frame module. The detectors in the camera are pseudo-stitched into a single image.

The functional component of the projection IOD are three private modules. The first private module includes indexing tags which reference the angle, camera position, and subject position indices. The second private module includes the geometric positioning of the x-ray source and camera. Lastly, a repeated private module represents each detector in the image, indexing the detector and providing the geometric location and orientation.

The second IOD is the reconstructed IOD which is based on the CT IOD. Once again, energy bins are packed together using the NM multi-frame module. However, a big change to the CT IOD is that the data represents linear attenuation values instead of Hounsfield units. This is because Hounsfield units offers little benefits, but a lot of extra complexity. Otherwise, no private modules are defined for the MARS reconstructed IOD at this time.

Together, the two DICOM IODs were carefully designed so that all MARS datasets are transferable and viewable in standard DICOM software. Where possible, standard tags were used, and private modules were restricted to functionality which is not related to viewing the data. This means that the MARS DICOM IODs are compatible with the DICOM standard. Perhaps in the near future, the designs given in this thesis will influence the standardized spectral CT IOD in DICOM.
10.1.4 Evolution of MCTBuilder into mPPC

The fourth task was the evolution of MCTBuilder into mPPC. With the recent changes including DICOM, mART, Medipix 3.1, the pre-reconstruction processing chain warranted a full review to improve the quality.

The first step was to integrate the MARS DICOM IODs developed before. This meant that the old directory hierarchy along with the scanner configuration and scan report files were obsolete. Also, the flexibility of mART meant that stitching and sinogram conversion were no longer required.

The result was the mPPC present a much simpler interface. The data could be presented in a single list, and the algorithms could also be presented in a small list. These simplifications meant that mPPC was easier and quicker to use.

The second step was to investigate the characteristic of bad detector elements as they presented a variety of behavioural differences. To this end I first defined the ideal detector element to follow a Poisson distribution. Afterwards, I categorized bad detector elements in the following way.

- Dead detector elements do not respond to stimulus from x-ray photons.
- Erratic detector elements produce too much noise to observe any signal.
- Unstable detector elements randomly change their behaviour at times.
- Truncated detector elements are the result of photon starvation or saturation.
- Drifting detector elements change their state over time.

To deal with these bad detector elements, a few additions were made to mPPC. Firstly, a threshold is applied to both raw and normalized data. Secondly, a Poisson distribution test removes detector elements whose response is clearly not ideal. Next, a dilation task removes detector elements which are isolated within large dead regions. A user guided mask then allows the removal of any other detector elements in the image as a backup option.

Lastly, a ring filter was developed to suppress static errors in MARS datasets. The filter calculates the difference between a detector element and a local median to gain an estimate of the static error. A second median is then calculated down
a line between exposures to refine the estimate of the static error. This error is subtracted from the original exposure to correct the image.

Altogether, the new algorithms meant that mPPC was better equipped to deal with MARS datasets. Despite ongoing issues with CdTe and GaAs based cameras, all bad detector elements could be filtered or removed in preparation for reconstruction.

10.1.5 Study of truncation

The fifth task was a study of truncation in MARS datasets. A significant number of scans were emerging where the scanning region of interest was poorly selected resulting in truncated datasets. Repeating such scans correctly is inconvenient and may not even be possible. Therefore, if the truncation is minor, it would be useful to be able to salvage at least some of the data.

The idea was to simply reconstruct the full subject as if truncation had never occurred. The hope is that despite the minor truncation, sufficient data remains to correctly reconstruct the internal regions, even if distortion occurs at the outer portions of the volume.

The testing procedure consisted of a single dataset which was reconstructed multiple times. In each reconstruction, the region of interest was identical, but truncation was progressively added to the scan data. After reconstruction, the internal regions of the volume were compared.

The conclusion of the study was that sufficient information is preserved in the interior regions of the volume. The difference images revealed that any errors in the internal region were sufficiently small to be negligible. This is by no means conclusive evidence that truncated data can be easily reconstructed. However, it is sufficient to allow pre-clinical studies to salvage some of their data from unfortunate mistakes in the scanning preparation.

The outcome of the study is the mART now allows the users to set the voxel size, along with the volume radius, length, and position. This allows datasets with minor truncation to be salvaged, but it also allows the user to tighten the bounds when a small subject is scanned in a large region. This makes the reconstruction process more efficient.
10.1.6 Design of a MARS processing server

The sixth task was a first look at automating the data processing chain. This led to the concept and design of a MARS processing server.

The investigation into how the MARS system might be automated drew me to the unified procedure step protocol within the DICOM standard. This protocol allows DICOM applications to submit requests for computing jobs to a PACS. The jobs can be any computing task including image processing, data analysis, data mining, simulation, etc. A workstation can then pick up the job from the PACS and perform the necessary steps to complete the task.

This protocol allows all MARS image processing algorithms to become independent jobs. As a set of plugins, the image processing algorithms would be independent modules within the MARS processing server. This would allow plugins to be developed individually and replaced if necessary.

The unified procedure step operates on a standard PACS. This means that the MARS processing server is compatible with modern clinical imaging standards and infrastructure. Therefore, integrating such a system into a hospital should be straightforward.

The central component of the MARS processing system is a job manager. This is the engine which controls the plugins and the interface to the DICOM network. It would profile the computing resources available and use the information to schedule and allocate its resources towards processing the incoming jobs.

Together, the unified procedure step protocol, the job plugins, and the job manager, form the MARS processing server. I have verified the details of the unified procedure step within the DICOM standard to guarantee that the design is feasible. Now, the only remaining task is implementation.

An brief set of development tests towards implementing the MARS processing server, resulted in the application called MARS Transfer. This application is a download and upload tool for DICOM datasets. It schedules data transfers as jobs and performs the jobs in sequence as quickly as possible. Due to its speed and simplicity, MARS Transfer quickly replaced K-PACS as the DICOM transfer tool for the MARS team.
10.1.7 Study of random sampling techniques

The seventh task in my PhD was a study of the feasibility of using random sampling techniques for projecting data during reconstruction. The motive for this study was the potential of investigating the use of octrees in CT reconstruction. Octrees could offer reconstruction improvements in efficiency, performance, and quality. However, octrees will not operate efficiently with the raytracing projection routine adopted in mART. Random sampling might offer a better option.

Even without octrees, random sampling could offer benefits to CT reconstruction in general. It deals with custom geometries, of the ray and the scanner, in a very natural way. The finite size of the x-ray source and detector, along with the motion of the gantry can be directly included in the reconstruction process.

My first achievement in this study was to develop algorithms for random sampling in both forward and back projection. Afterwards, four reconstructions were performed on a human excised atheromatous plaque dataset. These reconstructions included cases for pure raytracing, pure random sampling, random sampling with lookup tables, and a hybrid technique using random sampling for the forward projection and raytracing for the back projection.

The outcomes of the study was that the best quality could be achieved with pure random sampling at a heavy cost of performance. The attempt to use a lookup table to boost the performance for random sampling was unable to match the speed of raytracing. Also, the lookup table reduced the quality of the reconstruction. The hybrid approach was the fastest reconstruction but the gains were negligible.

Altogether, random sampling does not offer anything to the current form of mART. However, if octree based reconstruction is explored in the future, it would be good to revisit random sampling techniques as it does provide the best quality. Under the conditions of an octree, it could easily prove to be the best option.

10.1.8 Steps towards automation

The final task in my PhD was a series of steps towards producing an image processing chain that can be automated. The issue with mPPC and mART is that a series of steps require user guidance. In particular, the user guided masking could not be automated.

Other issues were present such as the presence of the ring filter. The ring filter
was necessary to produce high quality images. However, the performance left much to be desired as it was the biggest bottleneck in mPPC. Therefore, the goal of this task was to improve the SART algorithm to better deal with noise, and outliers.

Three additions were made to the SART algorithm. The SART algorithm applies a weighted average of a set of correction terms to each voxel in the volume. The changes I made add some conditions to the weighted average based on trustworthiness of the detector element.

The first condition added, was the Poisson distribution test from mPPC. This reduced the influence of any detector element which clearly does not behave according to the Poisson distribution.

The second condition was an outlier test. This compares a correction term to a local median to reduce the influence as the difference increases.

The final condition was a blending function. This rates detector elements in the center of the detector higher then the outer edges. In this way, any imbalance between overlapping exposures is blended together seamlessly.

Together, the three additions were able to remove the worst offenders in MARS datasets. These are the outliers which produce harsh rings and streaks. However, the static errors which produce minor rings are still present in the data. This means that the three additions are not yet sufficient to allow direct reconstruction from raw data. Further additions are necessary to remove the remaining ring artefacts.

10.2 Summary of the academic contributions

Throughout my PhD, I’ve worked with numerous pre-clinical teams and development teams in their research studies. Together with these teams I have published 6 journal articles and at 2 peer-reviewed conference proceedings, with another article ready for submission in the near future. I have also assisted with various MARS presentations to meetings in CERN and radiology groups which have furthered the cause of the MARS molecular imaging project.

The software developed during my PhD is available to the whole MARS team and to our external partners. This includes the Mayo Clinic, Rochester MN, Virginia Tech, Blacksburg VA, and the Joint Institute of Nuclear Research (JINR), Dubna Russia. While I have not directly collaborated with all the external partners and MARS pre-clinical teams, I have provided ongoing support to enable their
research using MARS technology.

The work presented here in my thesis is part of an ongoing development to improve the image processing capabilities of the MARS system. I leave this work here as a guide for future members to continue advancing MARS molecular imaging technology.

10.3 Future endeavours

There are many possibilities for spectral CT, the MARS project, and for extending the work presented in my thesis. As spectral CT matures as a modality, better image processing algorithms, better detector technologies, and better approaches to scanning should emerge. For the MARS project, the end goal is to integrate MARS technology into clinical practice. Lastly, the work in my thesis can be improved in various ways such as automation, material reconstruction, and octree accelerated reconstruction.

10.3.1 Short term outlook

In the short term, I expect that most of the advancements will be software based. The software changes would include the merging of mPPC and mART, the use of octrees to accelerate reconstruction, and material reconstruction. At some point in time Medipix 4 will enter the scene, however, as the feature list has not been decided yet, I don’t expect this to emerge for at least the next three years.

Combining mPPC and mART

The merging of mPPC and mART is an important step towards automation. Even if the MARS processing server were completed, the job plugins need to be algorithms without user guidance.

At the end of my thesis I managed to add conditions to the SART algorithm so that outliers are suppressed or removed. This means that the dead detector element identification algorithms, such as the Poisson distribution test and user guided masking, are no longer needed.

A key remaining improvement would be a faster method of suppressing or filtering rings. The current ring filter does not hinder the merge of mPPC and
mART, but it is a significant bottleneck as it a volumetric algorithm which operates on a large amount of data. Also, with the size of the dead regions in current MARS cameras, the conditions for the ring filter to operate are poor. A method of suppression which does not rely on neighbouring information would be superior in terms of quality.

**Octrees and CT reconstruction**

Earlier, in chapter 8 of this thesis I introduced the possibility of using octrees in CT reconstruction. A more generic description would be that it would be good to represent the volume using the largest appropriate voxel size for any given region. This means that the number of voxels is minimized which would improve both efficiency and performance.

Another advantage of maximizing the voxel size is that it is similar in nature to compressed sensing techniques. In compressed sensing, a sparsity constraint might limit the total variation between small voxels in a regular grid. However, a group of small voxels with no variation is the same as a large voxel. Therefore, I would expect space partitioning structures such as octrees to improve image quality as well.

**Material CT reconstruction**

Throughout my thesis I mentioned the concept of material reconstruction. At the time of writing, this was an ongoing investigation into combining CT reconstruction and material decomposition. The primary motivating factor is that considering the density of materials directly changes the polychromatic form of the Beer-Lambert Law into a form which can be solved. Instead of an infinite number of energy bin volumes, there are a discrete set of material volumes.

The main advantage here is that beam hardening will no longer cause image artefacts. Instead, it is valid signal which will contribute to determining the material quantities. This should greatly improve the ability to discriminate between materials.

Another advantage of material reconstruction is better exploitation of spatial information. As the energy bins narrow, the number of available photons decreases given the same exposure settings. This means that the noise is more statistically
significant and the image quality decreases correspondingly. This is because the current reconstruction algorithm treats energy bins as independent entities.

With material reconstruction, each energy bin contributes to every material volume. Therefore, it should be possible to use all the signal from all energy bins to improve the spatial signal in the reconstruction material volumes. At the very least, the image quality will be consistent between volumes.

### 10.3.2 Mid term outlook

In the mid term, I expect that the biggest change to the MARS system will be automation. When I started my PhD, the MARS system was very much a proof of concept. The papers published over my thesis tend to investigate the technology and what benefits it might provide to clinical diagnosis.

Now, the MARS prototype has become a commercial small specimen scanner for biological and pre-clinical research. I expect that from this point onwards, the MARS project will complete the change into a commercial venture.

To meet the needs of a professional, commercial system, an equivalent infrastructure will be needed. In particular, the software used to operate and process data should at least match the expectations of customers who are already familiar with commercial CT and Micro CT systems. For image processing, this means that automation will become a priority.

To this end, I have already complete a high level design for a MARS processing server. This is a server which can automate tasks and present a professional interface to users while still be modular and open enough for ongoing research into image processing algorithms. Also, as the design uses DICOM, it will be compatible with modern clinical infrastructures.

Therefore, I expect that the MARS processing server, or an equivalent system, will be completed within the next few years. This is a key step to customer satisfaction and will play an important role in commercializing the MARS system.

### 10.3.3 Long term outlook

The long term goal of the MARS molecular imaging project is to commercialize a human MARS system into clinical practice. This will involve developing the human MARS molecular imaging system, guaranteeing compatibility with modern
clinical infrastructure, and gaining approval for medical use.

The groundwork towards this goal is already in place. The MARS project recently succeeded in an $12M MBIE bid to complete the human MARS system. Also, active discussions are taking place to prepare a hosting facility, as a human sized scanner requires a lead shielded room. Lastly, for an initial prototype, there is the potential to gain access to a commercial CT gantry to speed up the process.

The human MARS system is still a five to ten year undertaking. However, with a wide collaboration of the Universities of Otago and Canterbury, GE Healthcare, CERN and the Medipix collaboration, as well as various research groups that form our external partners, the human MARS molecular imaging system is looking to become a reality in the not too distant future.

Alongside the human MARS system, the pre-clinical studies are also progressing. As the technology of the MARS system evolves to maturity, I expect that various pre-clinical studies will also come to fruition. This means that by the time the human MARS system is ready for approval, I expect that the clinical benefits of spectral CT will already be firmly established.

10.4 Afterword

My PhD has been a great opportunity and a learning experience. I hope that the work I performed and presented here continues to benefit both the MARS team and the wider scientific community. I also hope that the lessons learned provide guidance for improvements in the future. Lastly, I look forward to the ongoing work of the MARS project and wait with anticipation for its acceptance into clinical practice.
References


181


Appendix A

MARS processing server

The work described in this chapter started in October 2013 and it introduces the concepts and design of an automated processing server. Unfortunately, the processing server was unable to be completed during my PhD due to time constraints. However, the concepts and design are available for future development. Also, a useful outcome was MARS Transfer, an application which simply downloads and uploads data from a PACS server.

A.1 Motivation for Image Processing Server

When a typical person considers CT, the first thing that comes to mind are reconstructed images. In fact, in commercial CT systems, it can be difficult to even look at projection data. All of the image processing up to the reconstructed images are completed in the background behind a black box.

From a user’s perspective, this is the ideal. No further intervention is required to produce reconstructed images once the scan has been started.

In the case of the MARS system, we have a unique situation. Currently, the state of the MARS project is a research and development undertaking. Over time, this is evolving into a commercial venture with the end goal of producing human spectral CT systems for clinical practice.

To achieve the end goal, the MARS project needs to be fully automatic or at the very least, the ease of use should be comparable to current commercial systems. Also, the MARS system should be compatible with modern clinical standards so that the future human spectral CT system may be integrated into hospitals worldwide. Adopting DICOM data formats was a good first step towards this goal.

However, in its current state, every step of the image processing chain is an
active area of research. In addition, our external partners are also interested in conducting research in the intermediate steps of the image processing chain. For example, Virginia Tech University is actively researching reconstruction algorithms for interior tomography using a hybrid CT/spectral CT system [104].

The conclusion is that the MARS system needs an image processing chain which can be both automatic, compatible with modern clinical systems, but also be modular so that image processing components are plug and play.

A.2 Unified Procedure Step

The DICOM networking protocols offer some advanced functionality. A good example of this is the modality worklist. The modality worklist offers a method to request the creation or acquisition of modality data i.e. request a scan. The protocol also defines methods for operators to register themselves and the scanner to the request. From this point on the request status is set to “pending” and later “completed” once the scan has finished uploading all the data.

In another section of the DICOM standard is interoperability with HL7 (Health Level 7)\(^1\); a communication standard used by Radiology Information Systems (RIS). The modality worklist protocols in DICOM are designed to interact with HL7 standards. Together, this links patient history from RIS systems with imaging functionality and data from DICOM.

The advantage of the complete system is that each component acts as an independent entity. So a clinician can request a scan during a consultation. In an independent location, the scan can be completed. In a third independent location, the results can be discussed during a follow up consultation.

Naturally, it is convenient for the patient if all three happen in the same place at the same time. However, such requirements are not imposed on the system. This flexibility allows clinicians to treat any patient from any location as long as access to the data is permitted.

A recent addition to the DICOM standard, is the unified procedure step protocol. At a glance, this protocol is almost identical to the modality worklist, and the methods of operation are similar. However, the differences lie in what is requested,

\(^1\) HL7 is a set of international standards for transfer of clinical and administrative data between software used in healthcare.
and in what performs the tasks.

In a modality worklist a modality job is requested which is typically performed when the patient reaches the location of a relevant scanner. An operator then completes the procedure.

In contrast, the unified procedure step is all about data processing and analysis jobs. These would be completed by an independent server or workstation. As long as the server is informed of the job, it can take responsibility and perform the necessary steps.

Therefore, implementing the unified procedure step could allow all the goals mentioned earlier to be achieved. The jobs can be modular, replaceable plugins. Through the use of a standard PACS, and the unified procedure step protocol, these jobs can be processed automatically. As part of the DICOM standard, such a system would be compatible with modern clinical infrastructures.

The unified procedure step has the same advantages as the modality worklist as each component is independent of each other. In fact, with a fast networking grid such as the Karen Network in New Zealand, powerful servers could perform image processing jobs for MARS scanners all over the country. The locations of each server would only depend on the convenience of the location, and the cost of hosting the server. This would allow smaller hospitals to buy MARS systems without needing to maintain computing facilities.

A.3 Design of a Processing Server

The first part of the design process is to look at the roles that the processing server plays within the MARS project and the DICOM system. Fig. A.1 shows a detailed overview of a system which may be completed in the near future.

The University of Otago intends to set up a PACS to connect between numerous research groups including the MARS system. The primary interface to each group would be a RIS along with visualization software. The RIS would use the modality worklist to request scans from the imaging devices scattered around the research groups. The visualization software would interact directly with the PACS to allow viewing of data.

The MARS team would have the additional processing server. This would interact with the PACS using the unified procedure step to find and process jobs.
Figure A.1: An overview of the processing server in a larger DICOM system. In this example, the MARS system is one of a few research groups that use a centralized PACS. The red paths show the modality worklist, the blue paths show the data transfer, the green paths show the unified procedure step.

These jobs could include image processing, data analysis, simulation, etc. As each job is a plugin, all that is needed is a method to advertise the jobs available so that users, or scanners, can request them. Since the scanner can also request jobs, it means that reconstruction can start automatically.

A.4 Basic Infrastructure

The processing server has three components which need development. Fig. A.2 shows the internals of the proposed processing server.

The first component is the DICOM server. The server supports the protocols to transfer and verify data as well as the unified procedure step. It is important to note that the processing server is not a PACS. It will not store data, it can not be queried via other DICOM applications. Instead, all requests go to the central PACS, the processing server then fetches the jobs and the relevant data in its own time.
The second component is the job manager which is the heart of the processing server. The job manager knows and profiles the system resources. When sufficient resources are free it asks the DICOM server to locate a new job. If available, the job will be received and passed on to the relevant plugin for processing. The plugin will interact with the system resources provided by the job manager.

The final component are the plugins themselves. The plugins will be activated by the job manager and be given access to some system resources. This might include a CUDA or OpenCL context (if the GPU is used), some disk space to temporarily store the dataset, etc. Each plugin will perform a single task and the outputs of the task will be uploaded back to the PACS. Therefore, a reconstruction job would receive projection data, complete the reconstruction, and upload the results.

### A.5 MARS Transfer

An initial attempt at implementing part of the processing server resulted in an unexpected application called MARS Transfer. This application is simply a DICOM server which supports uploading and downloading of data. The upload and download tasks are treated as jobs which allows easy scheduling of data transfers. Fig. A.3 shows a screenshot of MARS Transfer in action.

MARS Transfer quickly replaced K-PACS, which performed the transfer of data.
between a central PACS and the image processing workstations. There were a few important reasons for this change.

Firstly, due to its simplicity, MARS Transfer was significantly faster. K-PACS downloads data, then adds the data to its database, then creates a bitmap thumbnail for each instance. MARS Transfer simply downloads the data to a directory. No database is maintained, also, no instances are converted in any way.

Secondly, the lack of a database is a strong point for MARS Transfer. After the downloads are completed, mPPC and mART will alter the data. Also, users can easily rename directories and files at their leisure. A database would be disrupted by such events and lose synchronization with the data. MARS Transfer has no issues with such cases. Data can be altered, deleted, or moved in any way without problems.

Finally, as a piece of MARS software, MARS Transfer has access to knowledge about private DICOM tags defined in the MARS system. This is important because MARS Transfer can recover the private tag details if it was lost in translation over
the network.

DICOM has syntaxes for how data is packaged. The default syntax is “little endian implicit.” This policy has three parts to it. Firstly, it is uncompressed. Secondly, little endian means that the bytes are stored with the least significant byte first. Lastly, implicit means that the VR of each tag is not included in the dataset. This saves space but means that each DICOM application must look up a dictionary to recover the VR for each tag.

Therefore, the MARS systems prefers the “little endian explicit” syntax. The explicit component means that each tag includes the VR in the dataset. This allows any DICOM application to read and display the values correctly. MARS Transfer can receive the MARS DICOM instances in any syntax and convert to the preferred “little endian explicit” syntax. This is useful when loading MARS datasets in DICOM applications not created by the MARS project.

Therefore, MARS Transfer was a welcome addition to the MARS system. Transfers were faster, and there was no risk of database related issues. Also, data could be maintained in a format which is better suited for loading in generic DICOM applications.

A.6 Conclusion

The processing server design presented here is still just a concept. However, it will meet the needs of the future MARS project during its transition to a commercial venture. It allows all the image processing to be automated while maintaining a modular design so that any algorithm can be upgraded or replaced. Most of all, as a DICOM compliant system, this will fit nicely into any clinical infrastructure. This means that the MARS system will be ready for clinical approval when the time comes.

A side product of this work, was the application MARS Transfer. This is a basic application which can query a PACS, and then download or uploaded data as a scheduled job. This was a fast application with several advantages over K-PACS, which was used before. Therefore, it quickly became the DICOM transfer software for the MARS team.