Supervisor – Dr. Christopher Lester, Blind Grade Number 8252X

Muon Scanning Tomography Image Reconstruction using X-ray Computed Tomography Software

Abstract

Muon Scanning Tomography (MST) uses cosmic ray muons to scan structures. One problem in bringing this technology to industry is the lack of a consistent reconstruction algorithm. Muons undergo multiple coulomb scattering, which has a dependence on the mass thickness of the material. This is analogous to x-rays, whose attenuation also depends on the mass thickness. This paper investigates whether these similarities can be exploited to image MST data. An algorithm was written which converted the muon data to mock x-ray data, by calculating the mass thickness. The mock x-ray data was then passed to open source CT software. The algorithm was fed with idealised, computer-generated muon data, for a range of materials and shapes, and the resulting images were analysed. The algorithm worked, with promising results, producing images which were sensitive to the shape and density of the objects being scanned. However, further work is required to investigate and reduce image artefacts, and to test the algorithm with real muon data before it can be used in commercial imaging.

1. Introduction

Muon scanning tomography (MST) uses high energy cosmic ray muons to scan structures, such as buildings and containers. Its advantage is that the muons are naturally present, so no radiation generation is required for the scanning. MST takes a long time, as the incidence of muons is low. Therefore, its use is in scanning inanimate objects. MST is not yet in industry, and one of the problems to be overcome is the development of an algorithm to turn the raw data into an image. This paper investigates whether the analogy between MST and X-ray computed tomography (CT) scanning can be exploited to image muon data.

Section 2 introduces the theory behind this concept and section 3 discusses the software used. Section 4 discusses the results, and section 5 presents some future development ideas, as time constraints meant this work was a simple proof of concept.

2 - Theory

2.1 - Moliere's multiple scattering theory

Muons traversing a material undergo many small angle scatters, because of the coulomb interaction. The effect of this multiple coulomb scattering is an approximately Gaussian [1] distribution of scattering angles, which, for small angles, is well described by Moliere's multiple scattering theory. The spread depends on the distance travelled through the material, the scattering length and the muon momentum (equation 2.1.1).

$$\theta_0 = \frac{13.6MeV}{p\beta c}\sqrt{\frac{x}{X_0}}\left[1 + 0.038 \cdot \ln\left(\frac{x}{X_0}\right)\right] (equation\ 2.1.1)$$

Cosmic ray muons are highly relativistic ($\beta \sim 1$), with an average energy of 4GeV. The scattering length depends on material properties. For elements, it depends on the density, atomic number and mass number (equation 2.1.2). To avoid complication, only elements are looked at in this proof of concept.

$$X_0 = \left[\frac{A \cdot 716.4gcm^{-2}}{Z(Z+1)\ln\left(\frac{287}{\sqrt{Z}}\right)}\right]\left[\frac{1}{\rho}\right]\ (equation\ 2.1.2)$$

2.2 - X-ray attenuation theory

X-rays travelling through matter are attenuated. The degree of attenuation depends on the density, mass thickness and mass attenuation coefficient of the material traversed [2] (equation 2.2.1).

$$\frac{I}{I_0} = \exp(-\frac{\mu}{\rho}x_m)\ \ (equation\ 2.2.1)$$

In this equation, the mass thickness x_m is the density multiplied by the distance travelled. The mass attenuation coefficient ($\frac{\mu}{\rho}$) is dependent on the x-ray energy, as well as material properties. Coefficients are shown for various elements in figure 2.2.1. [3] The complexity of these functions may cause problems in the data conversion stage.

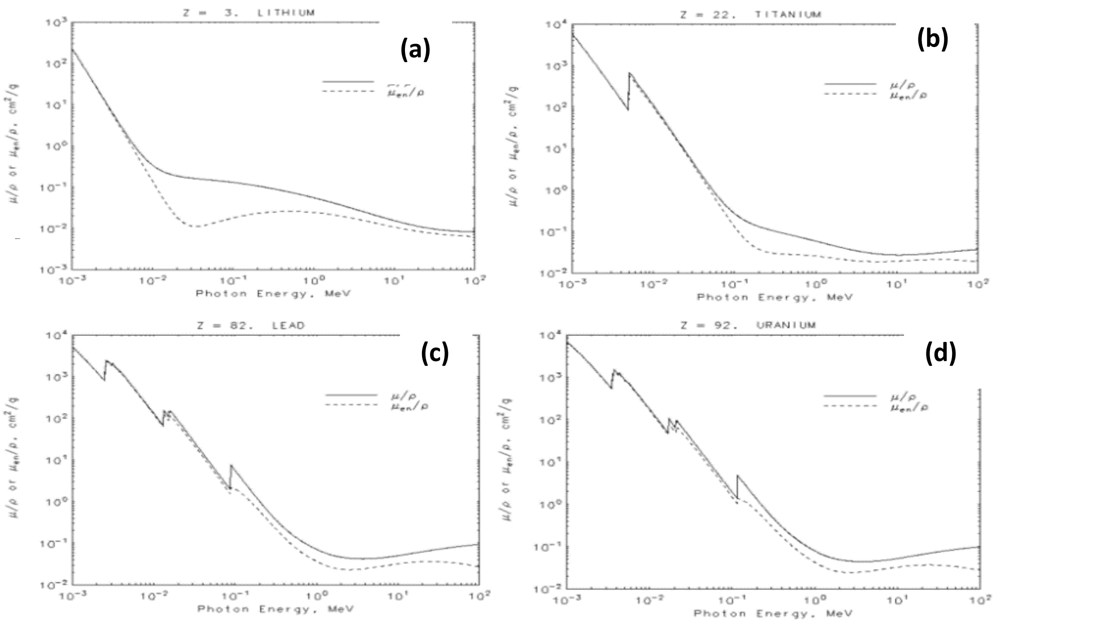

Figure 2.2.1 – X-ray mass attenuation coefficients for Lithium (a), Titanium (b), Lead (c) and Uranium (d). The dependence on energy has a complex functional form, ranging over several orders of magnitude. Whilst Titanium, Lead and Uranium are similar, the coefficient of Lithium is an order of magnitude smaller.

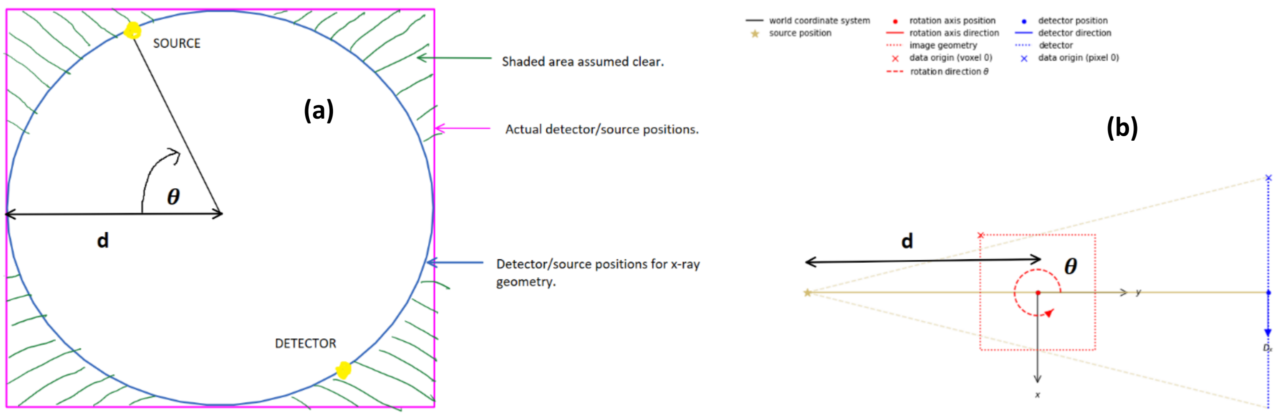

Figure 2.3.2 – Acquisition geometries. (a) shows the muon acquisition geometry. The positions and incidence angles of incoming muons are measured – then the position and angle are measured after passing through the sample. The data is grouped by incidence position and angle, which translates to a “source position”. This source position is labelled by an angle θ, and is assumed to be on the circle shown, which requires the region between the actual square of detectors and the inscribed circle to be empty. (b) shows the fan beam x-ray CT geometry, which the muon data has been mapped onto. An anticlockwise rotation of the material in this diagram corresponds to a clockwise rotation of source and detector in (a).

2.3 – Analogy between MST and x-ray CT

This work uses the parallels between muon scattering and x-ray attenuation to image muon data, using already highly optimised x-ray CT reconstruction algorithms. Muon data was analysed, and then converted into mock x-ray data, which was fed into the CT reconstruction software to produce an image.

Since the radiation length is a function of the density of material traversed (amongst other things), the value of mass thickness can be inferred from measurement of the scattering angle. The radiation length can be written in terms of a constant, multiplied by the inverse of the density (equation 2.3.1).

$$X_0 = \frac{C}{\rho} \; where \; C = \frac{A \cdot 716.4gcm^{-2}}{Z(Z+1)\ln\left(\frac{287}{\sqrt{Z}}\right)} \; (equation\ 2.3.1)$$

This constant is dependent on the material’s atomic number and mass number. The value for a number of elements is shown in Figure 2.3.1.

Element	**C / gcm^{-2}**
Lithium-7	81.78
Titanium - 48	16.52
Lead - 208	6.33
Uranium - 235	5.79

Figure 2.3.1 – The value of the prefactor for equation 2.3.1 for various elements.

Unlike the x-ray mass attenuation coefficients, these numbers are not multiple orders of magnitude apart. A representative value of 17 was chosen for this work, as this is within the range – however the value is arbitrary, since it simply scales the mass thickness, which is later multiplied by an arbitrary number when generating the mock x-ray data. After making this approximation, equation 2.1.1 takes the form (equation 2.3.2):

$$\theta_0 = \frac{13.6MeV}{4000MeV}\sqrt{\frac{x_m}{17}}\left[1 + 0.038 \cdot \ln\left(\frac{x_m}{17}\right)\right] \; (equation\ 2.3.2)$$

From this, the value of mass thickness was obtained.

Once the mass thickness was known, fake CT data could be generated. The muon geometry was converted to 2D Cone Beam x-ray CT geometry. Figure 2.3.2(a) shows the muon acquisition geometry, and 2.3.2(b) [4] shows the x-ray CT acquisition geometry.

After this, mock x-ray transmission data was generated using equation 2.2.1. Since real x-rays were not used, the mass attenuation coefficient is arbitrary, so could be chosen to optimize the images produced. Once this data was generated, it could be fed to the CT Package, which produced an image.

3 – Software

3.1 – X-ray Software

The CT package used was the core imaging library (CIL) [5]. This software was chosen because it is open source, and has a large number of demos. It is easy to use, and allows the user to specify the scanning geometry. The specific algorithm used was filtered back projection (FBP) from the Astra toolbox [6]. Back Projection takes absorption data, and projects it back along the ray path, based on the angle and detector pixel number. After doing this for all ray paths, a set of simultaneous equations is produced. Each voxel has a specific absorption associated with it, and the simultaneous equations add various combinations of voxels together. The FBP algorithm then solves this set of equations, to obtain the amount of absorption, and therefore the density of each individual voxel. Figure 3.1.1 shows a simple example.

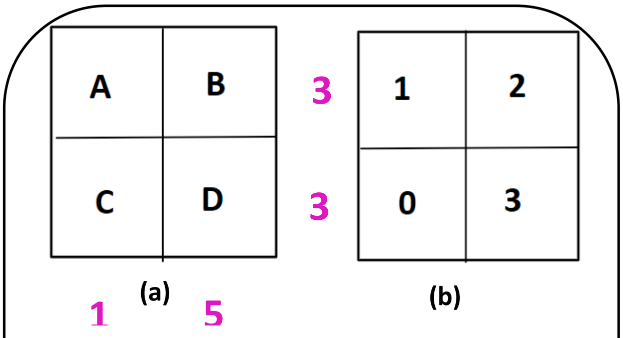

Figure 3.1.1 – A simple demonstration of filtered back projection. (a) represents the data given to the algorithm. There are 4 voxels of unknown value, but the pink numbers show the total value of a given row or column. For example, if you add up the first row, you get a total value of 3. (b) shows the result of the algorithm, which solves the 4 equations to obtain the values of the voxels.

The filtering step serves to sharpen the image – the default filter was used. Recently, iterative reconstruction algorithms have gained popularity in CT imaging. These perform FBP, but then take this image, and simulate what x-ray absorption data the image would give, and then compare this to the original data. This process continues for a specified number of iterations, improving the image. Since the project already generates imperfect CT data, this method may not have improved the results, and simple FBP was sufficient. More information about CIL can be found in 'Core Imaging Library - Part I: a versatile Python framework for tomographic imaging' [5].

3.2 - Data Generation

As this was a proof of concept, mock data was generated using a simple python script. The script determined which materials were passed through, and the distance traversed. The first pass found the average material properties across the entire muon path, and used this information to calculate an average radiation length. This was combined with the total length of the muon path to find an approximation to θ_0. Twenty random numbers were then generated, from a Gaussian distribution of standard deviation θ_0. The data was returned as a numpy [7] array, with a set of muon entry positions and angles, and a corresponding set of twenty scattering angles.

Later, an alternative method of calculating θ_0 for the generated data was tried. This depended on the number of material blocks traversed. If no materials were traversed, θ_0 was simply zero. If one material was traversed, equation 2.1.1 was used. If multiple materials were traversed, an average radiation length across the materials, and total distance travelled through the materials were calculated. This is better than adding the contributions in quadrature – which gives a result which is systematically too small [1]. The final images produced were compared with the original method.

4 - Results

4.1 – The effect of changing the data generation

The effect of changing the data generation method was investigated, to heuristically find the best method. It was investigated for a single object, and then for two objects of different, but similar densities. The results for a single square of Uranium-235 are shown in Figure 4.1.1.

The averaging method of data generation (b) clearly leads to a worse image. The edges of the square are fainter, and the cross shaped artefact is stronger. The density scales of the two images are also quite different. This is expected, since (c) corresponds to the actual expected gaussian width, whereas (b) involves averaging, so you would expect the result to be more "spread out". The "density" of the square in the averaging image is about 5 times weaker than that of the exact calculation image. This is expected – the value of x/X_0 for the averaging method is approximately 10, whilst that for the more realistic method is approximately 50 – giving the observed factor of 5.

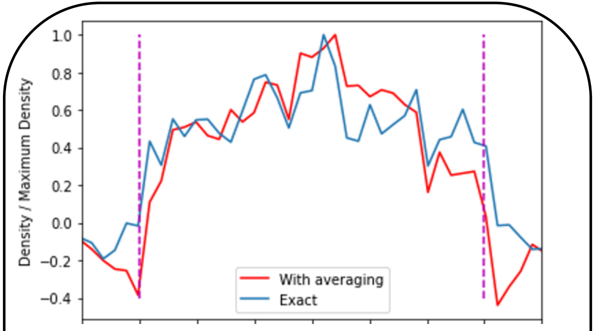

Figure 4.1.2– A 1D slice for both methods, normalised by the maximum density. The averaging method performs worse at the edges, but in the centre the performance is similar.

Figure 4.1.2 shows a 1-Dimensional slice through the centre of the image, normalised by the maximum density. The normalised density for the image with the averaging data generation is clearly below that for the exact calculation for at least the first 15% of the

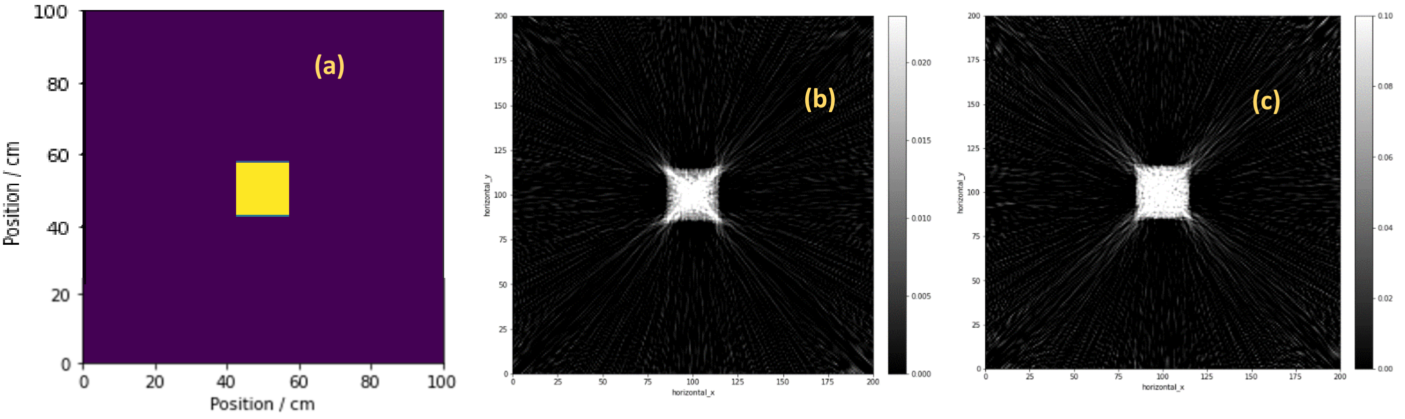

Figure 4.1.1 – The effect of changing the data generation method on the quality of the image. (a) shows a density map of the situation being simulated – a square of Uranium in the centre of the sample. (b) shows the image with data generated using the averaging method, whilst (c) shows the image with data generated using the radiation length and distance travelled through the square.

object. This further emphasises the observation that the edges are less well defined with the averaging method.

Figure 4.1.3 shows the images for the two different methods for two squares, one made of Lead and one of Uranium.

Again, the first method (b) produces a worse image than the second method (c). There is a clear “Leakage” artefact between the two squares in (b), whereas (c) has much better defined corners of the squares, and you can observe the correct situation of two unconnected squares from this image. This time, (c) has about a factor of 3 more density than (b) – we would expect the discrepancy to be less than with a single object, as the actual situation is closer to the averaging.

Figure 4.1.4 shows a 1D slice of the images, from the top left corner to the bottom right. These pass through the Uranium first and then the Lead. The first data generation method clearly has problems here – the Lead cannot be distinguished from the empty space between the two squares. Comparing the two methods, we see a clear region of lower density between

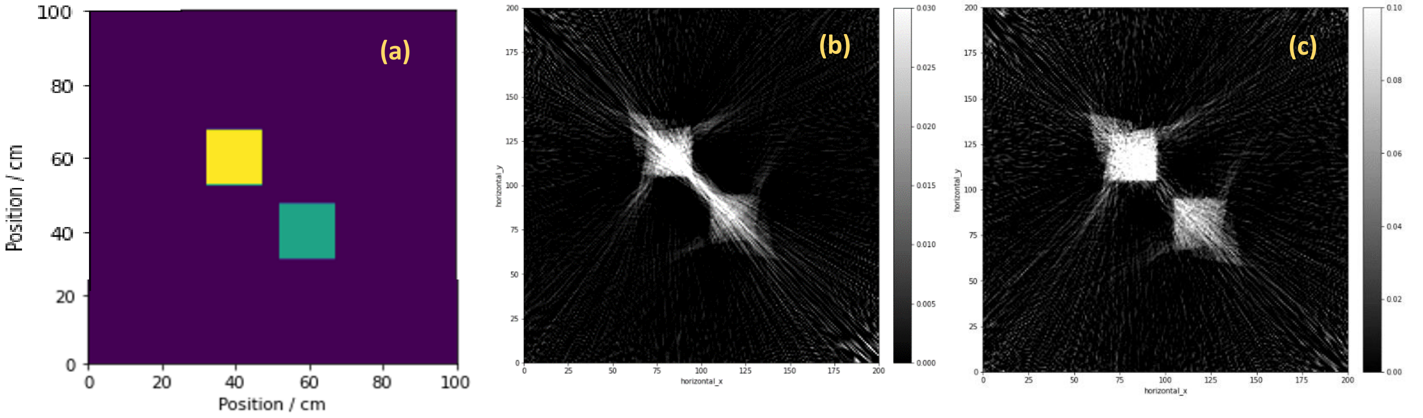

Figure 4.1.3– The effect of changing data generation on the images produced for two objects made of different materials. (a) is a density map of the situation being imaged – two squares, one made of Uranium and one made of Lead. (b) shows the image produced when data is generated by averaging across the entire ray path. (c) shows the image produced when data is generated by averaging only across the objects.

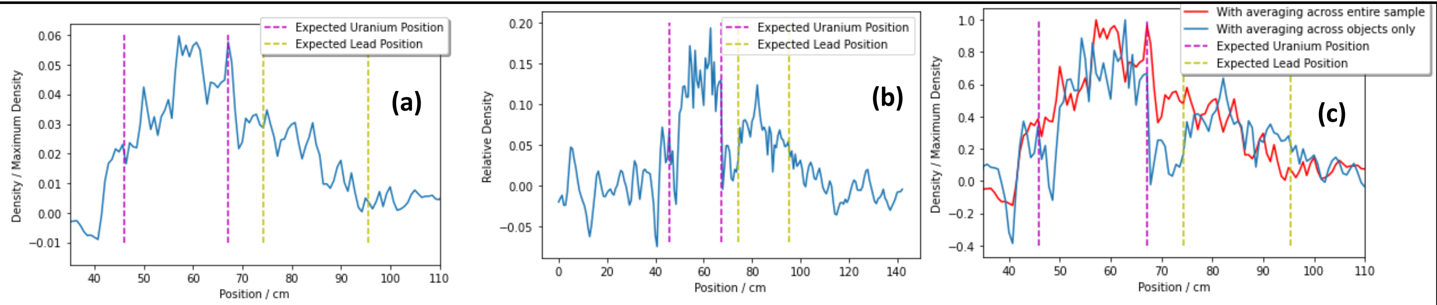

Figure 4.1.4 – A diagonal slice through the images. (a) shows the slice for averaging across the entire sample, and (b) for averaging just across the objects. (c) shows a comparison of the normalised densities. The second method of averaging across only the materials leads to a better result.

the objects with the second data generation method, but no such region for the first.

After viewing this data, the second data generation algorithm is clearly more accurate, so this was used for all future plots.

4.2 - The effect of changing pixel number

Three different numbers of pixels were investigated. Data was taken for a single square of Uranium-235 in the centre of the scanner. A density map for the generated data is shown in figure 4.2.1 (a).

As there was a single object, the data was generated using the distance travelled through the Uranium, and the radiation length of Uranium-235. The data was then analysed for each pixel number. As expected, increasing the number of pixels reduced the pixelation, and sharpened the image. A reduction in the visibility of image artefacts was also seen.

However, in the real world, increasing the number of pixels results from increasing the data collection time. A lower bound for the rate of measurable muons is approximately 4.2m^{-2}s^{-1} [8]. The lowest resolution image has data for 10,000 different muon paths. Since we need approximately 20 measurements to meaningfully find the standard deviation of the scattering angle, this image corresponds to 200,000 muons. The resulting data collection phase, for a 1m square detector, would take approximately 13 hours. The central image would take 9 days and the highest resolution image 33 weeks. If the detector size was of a similar order to the sort of large structure that MST could be used for, for example a square of area 400m^2 – analogous to a motorway bridge, these numbers come down to 2 minutes, 32 hours, and 5 weeks. For the rest of the results, we will proceed with the resolution of the central image.

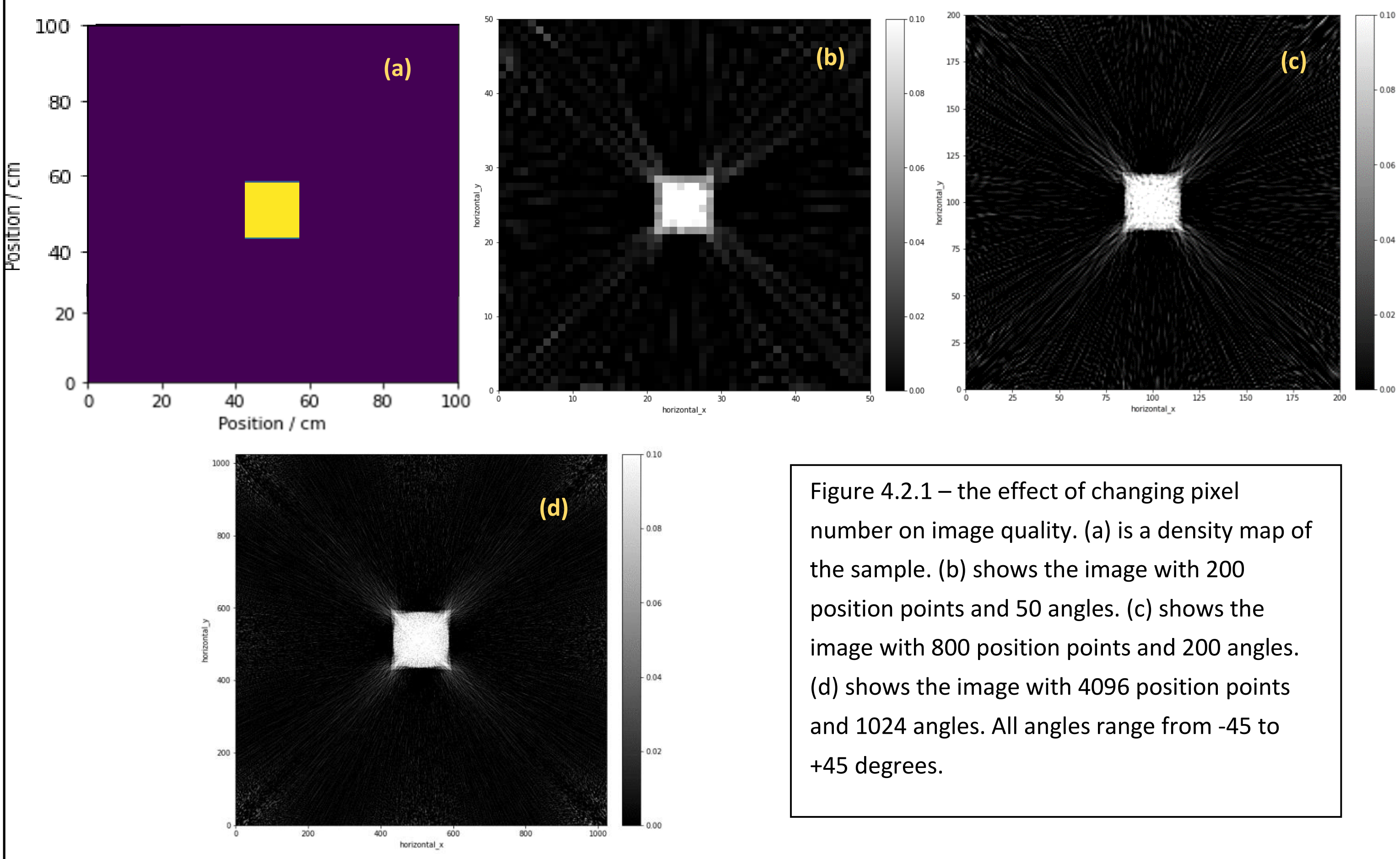

Figure 4.2.1 – the effect of changing pixel number on image quality. (a) is a density map of the sample. (b) shows the image with 200 position points and 50 angles. (c) shows the image with 800 position points and 200 angles. (d) shows the image with 4096 position points and 1024 angles. All angles range from -45 to +45 degrees.

4.3 - The effect of changing attenuation coefficient

The value of the x-ray mass attenuation coefficient is arbitrary, since we are dealing with imaginary x-rays, so can be tuned to highlight specific image qualities. Increasing the coefficient is analogous to increasing the contrast of a scan – the x-rays are attenuated more as the coefficent increases. The changing sensitivity provides a method of identifying the absolute density or specific material of an object – whereas a single scan in isolation can only identify shapes. Figure 4.3.1 shows a series of images of two squares, one made of Uranium and one made of Titanium, with varying coefficients.

With the lowest two values of attenuation coefficient ($\mu = 0.001, \mu = 0.01$), only the Uranium square is visible. The third image ($\mu = 0.1$) is the first where the Titanium square is visible, although it is significantly fainter than the Uranium. In the fourth image ($\mu = 1$) the squares look identical. Only through looking at the combination of images with different values of μ, can it be concluded that there are two squares, one with a higher density than the other.

Figure 4.3.2 shows the image densities of a square of various types of materials, as well as the densities of image artefacts, and how these are affected by the attenuation coefficient. This data was generated by considering a single square of each type of material, in the centre of the detector. The average density across the expected position of the square in the image was then calculated. The background numbers are the average density across the rest of the image (with a small exclusion zone around the square), whilst the density of artefacts was calculated by considering the densest 5% of the background. Both of these numbers were calculated with a square of Uranium, since the

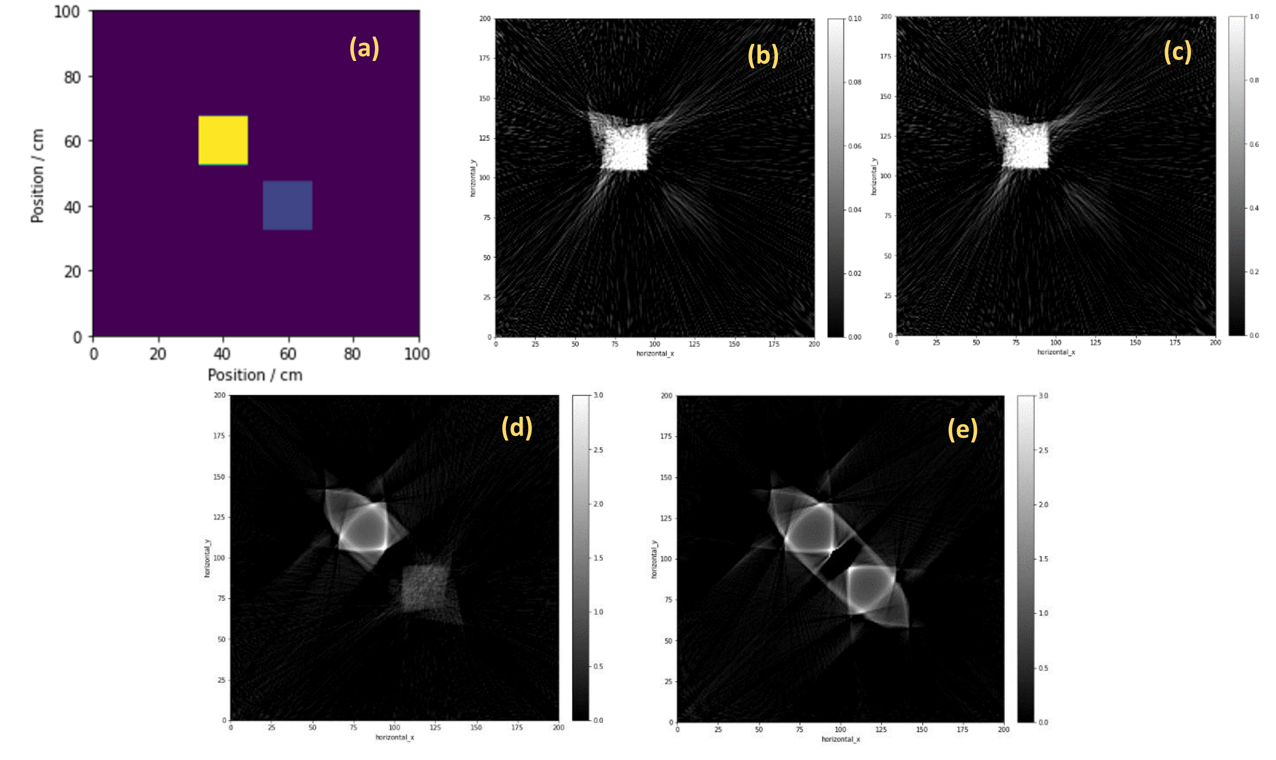

Figure 4.3.1– The effect of changing the attenuation coefficient on materials of different densities. (a) shows a density map of the situation to be imaged – a square of Uranium and a square of Titanium. (b)-(e) show images with increasing values of the attenuation coefficient, μ, (0.001, 0.01, 0.1, 1).

densest object gives rise to the largest artefacts, and an upper bound was sought.

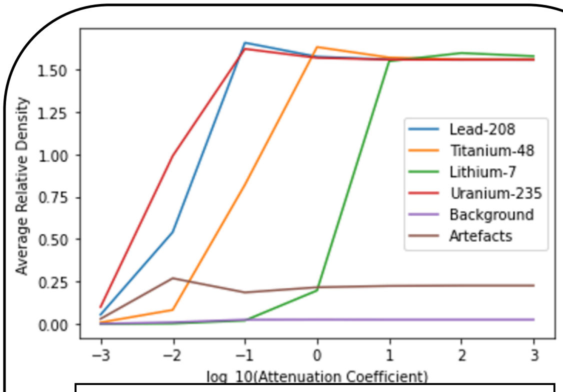

Figure 4.3.2– The average relative densities of various materials, as well as background and artefacts. This shows that different materials become visible at different attenuation coefficients. At high coefficients, all materials appear the same, since mock x-rays are fully attenuated.

At low values of attenuation coefficient, the relative density is larger for denser materials, as expected. As the attenuation coefficient increases, the average density increases – and a factor of 10 increase in the coefficient leads to approximately a factor of 10 increase in the density. This is expected, since increasing the coefficient increases the mock x-rays' sensitivity to density, and is analogous to increasing the contrast. The artefacts and background do not have the same sensitivity to density, and remain approximately constant throughout. This means that as the coefficient is increased, lower and lower density materials overtake the artefacts in visibility. This is why in Figure 4.3.1, the Titanium square starts to be visible between a coefficient of 0.01 and 0.1. We could therefore predict that a Lithium square in a similar arrangement would become visible between a value of 1 and 10. This coefficient dependence is very useful, as it means we can also use this software to distinguish between different materials. For example, test objects of various materials could be imaged using the algorithm, at varying attenuation coefficients and the images from scans compared, to determine the actual material scanned.

As attenuation coefficient is increased further, there is a saturation value of approximately 1.5. Again, this is expected – at high values of attenuation coefficient, any amount of mass will lead to complete attenuation, so no information can be gained other than that there is some minimum amount of mass thickness between source and detector. This explains why in Figure 4.3.1, at an attenuation coefficient valus of 1, the two squares appear to be the same density.

4.4 - Imaging complex shapes

One potential application of MST is to check internal structures of buildings. Therefore, it would be useful if complex shapes could be accurately imaged. Figure 4.4.1 shows an image of "complicated shapes" – here a smiley face and a sad face.

There are significantly more artefacts than for simple shapes, which provide uncertainty as to the actual shape of material. However, the overriding features are still there – the image does resemble a smiley/sad face. If we have some prior information – for example we know that the object is either a smiley face or a sad one – the images produced can tell us which it is.

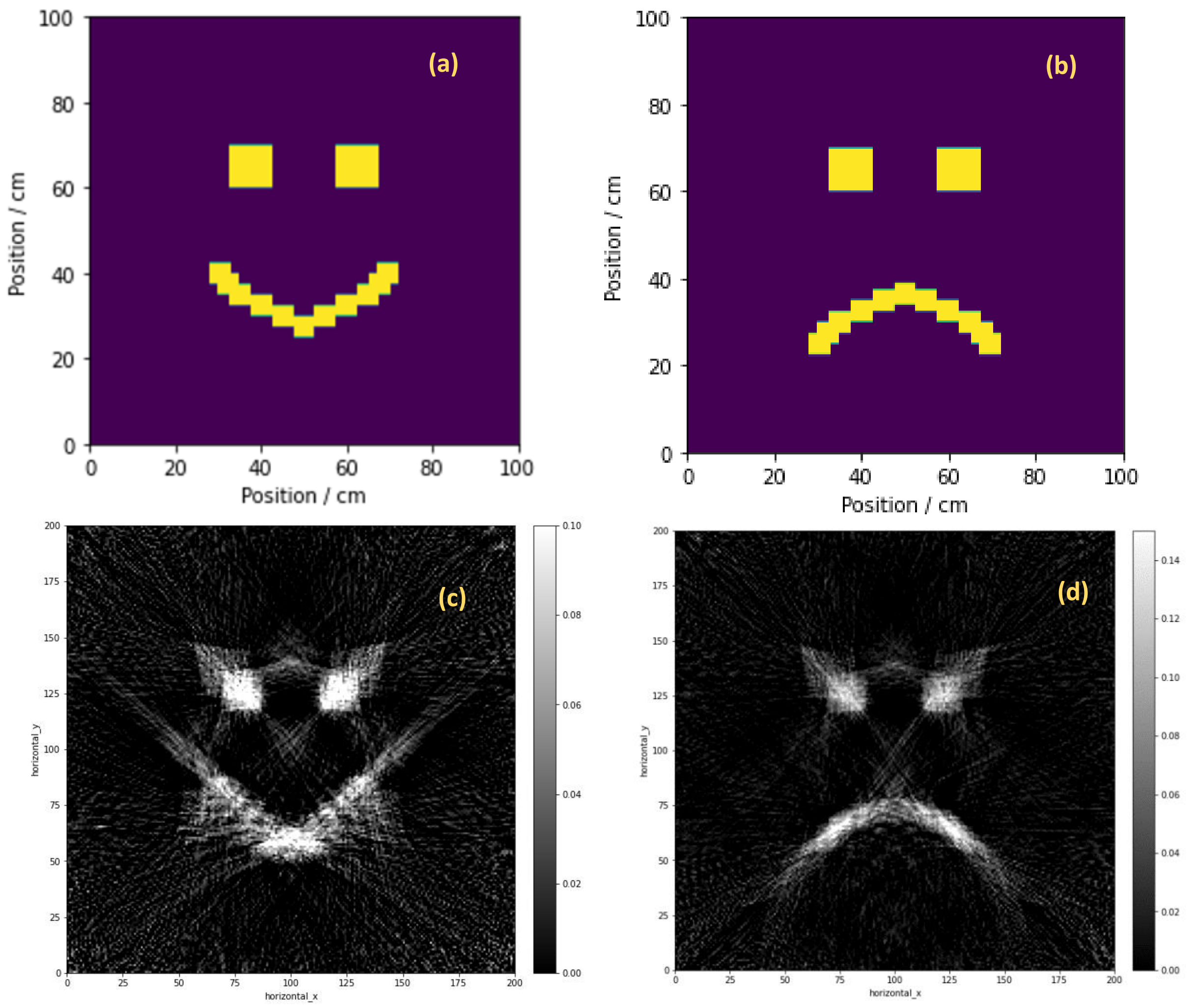

Figure 4.4.1 – Images of complicated shapes. (a) and (b) show density maps of the situation being imaged – a smiley/sad face. (c) and (d) show the resulting images. There are large artefacts, but the images still strongly resemble the real situation.

5 – Future Developments

5.1 – Artefact Reduction

A significant problem throughout these results has been the presence of image artefacts. Unfortunately, time constraints meant there was no time to investigate the causes of these artefacts, and therefore take steps to remove or at least reduce them. However, some suggestions are presented here for future developments of this type of MST reconstruction software. The images of the central square (Figure 4.2.1), being the least complicated can provide good insight into the type of artefacts. The highest resolution image shows predominantly streak artefacts, which are a known problem with x-ray CT images [9]. Therefore there is some hope that the CT package used would be able to eliminate some of these, through existing functions. These functions involve iterative reconstruction algorithms, which have become more popular due to increasing computing power in recent years [9]. This technique could improve the images, however, there is significant danger that this would actually worsen the images, since the x-ray data being used is not from real x-rays.

5.2 – Potential Developments

Another improvement would involve trying to extract more information about the particular material. Both MST and x-ray CT data actually depend on the type of material, not just its density, as is assumed in the work above. A future algorithm could take extensive data for known samples using both muons and x-rays, and compare these heuristically. After learning the data, the machine could more realistically convert muon data to x-ray data.

The algorithm has only been tested on idealised, computer-generated muon data. Future developments would need to test it on more realistic simulated data, such as from Geant-4 [10], and eventually on real muon data.

6 - Conclusions

The work has shown that it is possible to exploit the similarities between x-ray CT and MST data to create an image reconstruction algorithm for MST. The resulting algorithm was sensitive to density differences between materials, and the shape of the objects. The mass attenuation coefficient of simulated x-rays can be tuned to identify specific materials. The algorithm is already sufficient to determine the contents of the area if prior knowledge indicates the contents to be one of a finite number of options. However, further work needs to be carried out, in particular to reduce the artefacts. Additionally, the algorithm needs to be tested on real data. Overall, the results have been promising, and represent a step towards a consistent algorithm for MST being developed.

References

[1] – Particle Data Group Collaboration, S. Eidelman et al. *Passage of particles through matter.*

[2] - https://physics.nist.gov/PhysRefData/XrayMassCoef/chap2.html

[3] - https://physics.nist.gov/PhysRefData/XrayMassCoef/tab3.html

[4] - https://github.com/TomographicImaging/CIL-Demos/blob/main/training/2021_Fully3D/Week1/00_CIL_geometry.ipynb

[5] - https://royalsocietypublishing.org/doi/10.1098/rsta.2020.0192

[6] - https://tomographicimaging.github.io/CIL/nightly/plugins.html#id4

[7] - https://numpy.org/

[8] – F. Keizer et al. *A compact, high resolution tracker for cosmic ray muon scattering tomography using semiconductor sensors.*

[9] - https://pubs.rsna.org/doi/abs/10.1148/radiol.2015132766?journalCode=radiology

[10] - https://geant4.web.cern.ch/