WSIRT: A MODIFIED ITERATIVE RECONSTRUCTION TECHNIQUE FOR PHOTON STARVATION ARTIFACTS

by

LINRUN MAO

(Under the Direction of Mark Haidekker)

ABSTRACT

The clinical applications of CT are integral parts of the modern-day biomedical field. One of the many flaw tomography faces is the photon-starved image artifacts. This report presents a modified reconstruction technique to mathematically attenuate the losses of absorbance data. This method utilizes the nature of Algebraic Reconstruction Technique (ART) convergence as well as its construct to detect and complement the information loss. The Weight Shrinking Iterative Reconstruction Technique (WSIRT) method showed promising results in decreasing the streaks in a reconstructed image. A mask effect was found through dissecting the reconstruction that can accurately pinpoint the location of the material that caused the photon starvation, while also creating a mask for the resulting streaks. This provides great potential as a middle step for a further image treatment to suppress this type of artifact in CT.

INDEX WORDS: Computed Tomography, Reconstruction Technique, Algebraic Reconstruction Technique, Photon starvation, Iterative Reconstruction, Image artifact

WSART: A MODIFIED ITERATIVE RECONSTRUCTION TECHNIQUE FOR PHOTON STARVATION ARTIFACTS

by

LINRUN MAO

B.S., The University of Georgia, 2019

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2021

WSART: A MODIFIED ITERATIVE RECONSTRUCTION TECHNIQUE FOR PHOTON STARVATION ARTIFACTS

by

LINRUN MAO

Major Professor: Mark Haidekker

Committee: Caner Kazanci
Kun Yao

Electronic Version Approved:

Ron Walcott

Dean of the Graduate School

The University of Georgia

May 2021

ACKNOWLEDGEMENTS

The author would like to acknowledge Dr. Haidekker and all the University of Georgia faculties who supported the research. The author would also additionally acknowledge all who supported the researcher mentally.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

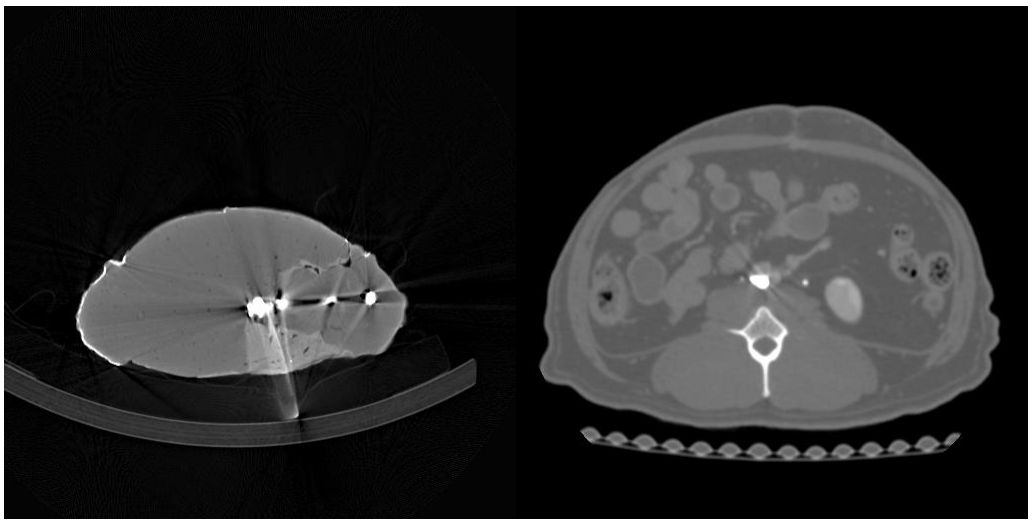
Problem and Purpose of the Study

Computed Tomography relies on the nature of x-ray attenuation as it passes through materials to reconstruct the cross-section of an object. The sensor detects the incoming x-ray intensity to provide data for a meaningful reconstruction, based on the theory of Radon transform [1]. One of the major causes of the image artifacts, photon starvation, is created when the sensor cannot receive sufficient x-ray intensity. This occurs commonly when the x-ray beam path is too long, thus accumulated high attenuation, or when it travels through a dense material that has high absorption, such as metals. The phenomenon usually results in streak-like artifacts in the reconstructed image, decreasing image quality and potentially overwriting minor details around high absorption materials.

The general approaches to combat this issue can be crudely divided into hardware and software approaches. An easy way to alleviate photon starvation is to increase the energy of the x-ray, by simply turning up the tube voltage and current, achieving a stronger penetration power. This method will trade off the image quality with radiation dosage. However, due to modern clinical CT trending towards faster acquisition and less radiation dosage for the sake of the patient, the main focus of looking for a solution weighs on the software side. The purpose of this thesis is to develop and experiment with a form of reconstruction technique to reduce the photon starvation image artifacts.

Background

One of the main applications for CT is on the human body for its clinical advantages. Development in the implant and artificial replacements of the medical field showed promising results in solving various health issues. From hip replacement to dental implants, or even spinal fixation devices, such medical technologies bring improvements, but also problems. Many of the described operations require reliable detection and monitoring techniques in order to track and discover results and complications. The issue arises as most implants are dense and have high absorption values compared to normal human tissues and bones. These high absorption materials cause photon starvation as described previously, thus creating image artifacts in the resulting cross-section reconstruction. See Figure 1 as an example, the image artifacts will blur out the details around these implants, giving radiologists a hard time determining potential implant failures.



(A) (B)

Figure 1[1]. Examples of streak artifacts caused by photon starvation (and generally referred to as beam hardening artifacts). (A) shows a cross-section through a statue made of plaster, which has been reinforced

[1] Images provided by Dr. Haidekker.

with iron rods. While the plaster has a widely homogeneous absorption, apart from a few air pockets, the iron shows excessively high absorption, which causes shadow-like streaks in the reconstruction. (B) shows a cross-sectional image from a primate. The high absorbing spinal bone is prominently visible. In addition, contrast agent was injected into the vena cava, which was poorly distributed at the time of imaging. Its high concentration, combined with the contrast agent's high absorption, causes some minor diagonal streak shadows.

The main focus of the thesis is on software approaches, and so will the discussion on its background. There are three main areas where methods were developed to reduce these artifacts, reconstruction relying on dual-energy x-ray beam, artifact reduction reconstruction algorithms, and image post-processing [2]. Each of these methods targets a different stage of CT in an attempt to suppress or even eliminate this issue.

The first concept is the dual-energy x-ray spectrum, which stands for the preprocessing of the CT stage. Along with hardware changes that deal with different image artifacts, it sets up an environment before acquisition in order to have better raw data to begin with. Changing the x-ray energy level arguably is also a part of the hardware process, but the reconstruction and extrapolation process that goes with it fits in the software method in reducing image artifacts [3].

The remaining two methods are the main focus of the study, representing the reconstruction phase and the post-processing phase. Common reconstruction algorithms used in CT such as Filtered Back Projection (FBP) and Algebraic reconstruction technique (ART) are main targets for these researches. They are often combined and referred to as artifact reduction algorithms, and since most of which is caused by metal, they are often named Metal Artifact Reduction (MAR) algorithms. These algorithms, unlike conventional image processing techniques, target the raw projection data, or the sinogram. Many of the algorithms find a way to remove the projections that did not reach sensors (photon starvation) and interpolate from nearby projections [4],. These methods include frequency filtering/splitting, adaptive filtering, sinogram

interpolation, and iterative algorithms. The WSIRT fits well into the iterative MAR algorithms, finding a way to use convergence as a path to interpolate missing projection data. The iterative methods in MAR algorithms have been proposed as early as the 1990s, as described by [5], to produce superior image quality compared to FBP after linearly interpolate projection gaps. These iterative methods in general are not widely used compared to simpler methods such as FBP, mainly due to their high construction costs.

The postprocessing methods focus on treating an already reconstructed image, often through different transformations into the frequency domain or through filters, examples seen in [6] [7] [8]. Some even attempted to use convolution network-type AI to solve the issue [9]. It is hard to isolate this type of method from the previous phase since the direct issue lies within the missing projection data, and a reconstruction may blur the effect or even hide it. It is much harder to have a solution from this isolated phase, but does not stop the researchers from attempting. For example, in [8], Hao et al. used a wavelet transform-based suppression algorithm to accommodate directional characteristics of the streaks when separating different frequency components. Other researchers created algorithms with similar concepts around the frequency domain, to identify artifact streaks and adaptively interpolate for its values, while not adding in new artifacts or noise [6] [7] [10].

The goal of this thesis is to develop a functioning variant of the iterative algorithm that can effectively suppress photon-starvation artifacts. CT has the feature of rotational acquisition, meaning one point of the material is scanned multiple times in a rotation. This creates the possibility to use unimpaired projection data to replace failed ones in an iterative process and

potentially recreating an image without the artifacts as shown in Figure 2. Results of the algorithm should be identified with either data processing or visually.

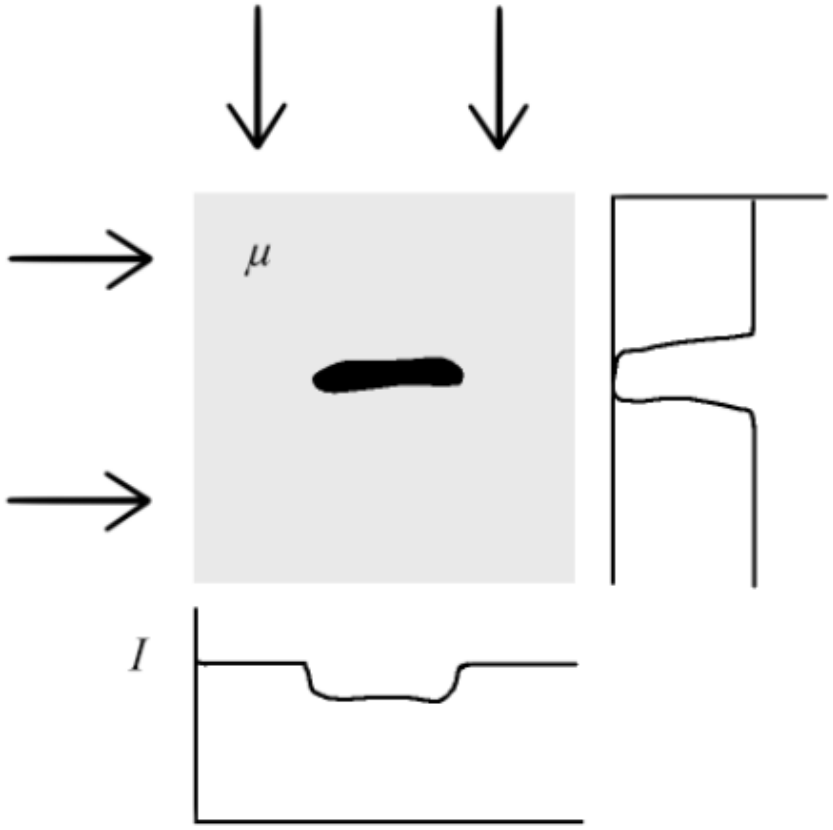


Figure 2. An illustration of the concept, wherefrom one acquisition angle, the absorption of the material brings the x-ray intensity down to an undetectable sensor range, yet from another angle, the information remains.

CHAPTER 2

THEORY

Tomographic Reconstruction

The foundation of this algorithm is the mathematical underpinning of CT reconstructions, the Radon transform. It is the mathematical representation of the CT setup for data acquisition and reconstruction. A pair of illustrations (Figure 3, 4) are provided below to demonstrate the effect as well as defining some nomenclatures.

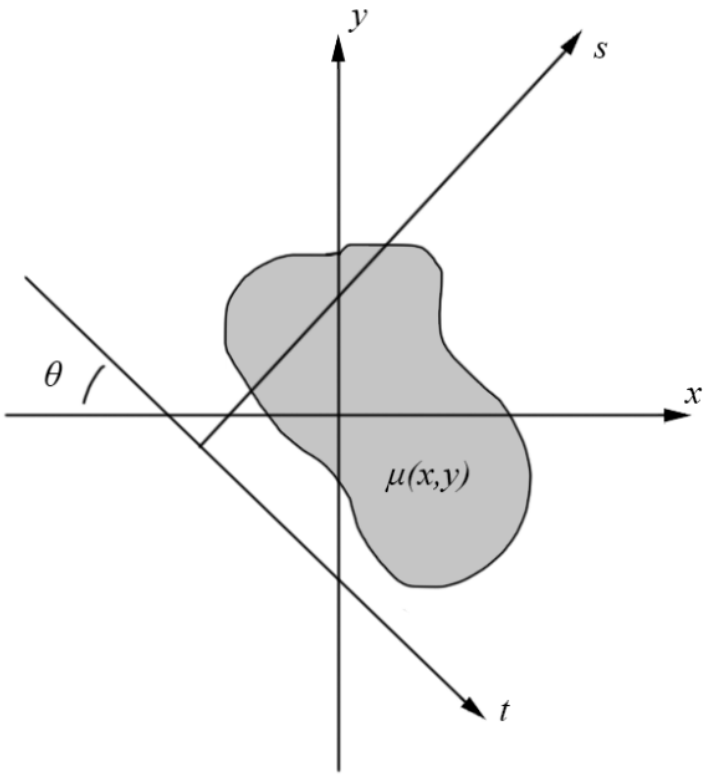


Figure 3. Illustration of an object being scanned. The object is placed in the x-y plane, and the scanning is done through iterations over multiple angles θ in the s-t plane.

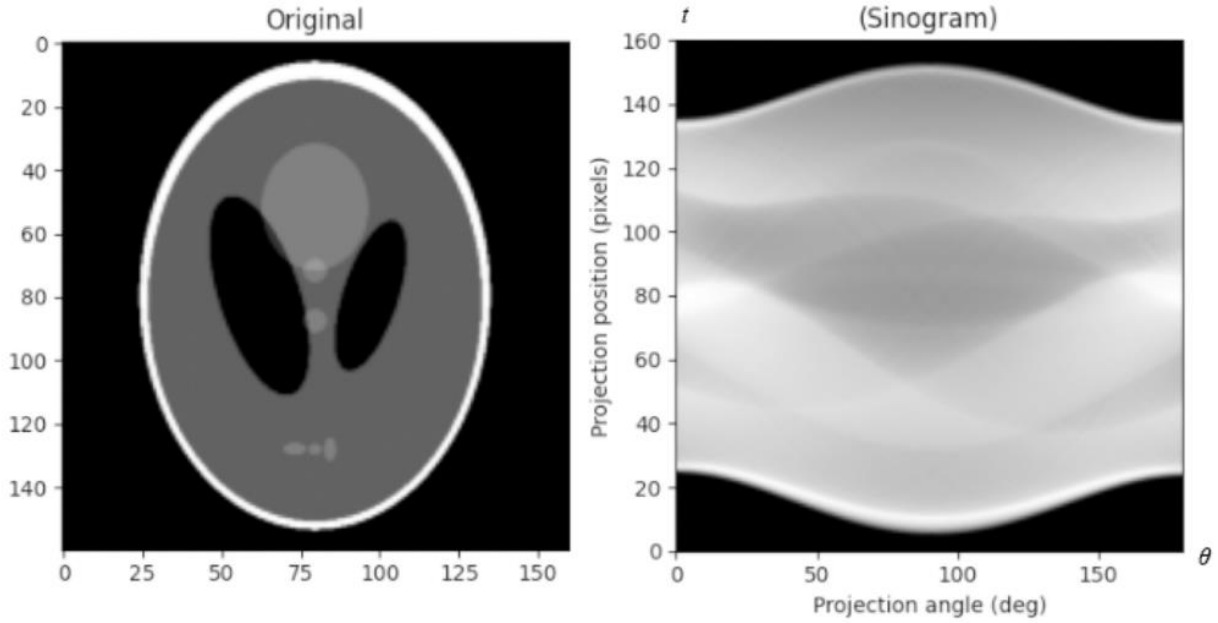


Figure 4. An example of an object (Shepp-Logan phantom) and its corresponding sinogram with labels aligned to the described system.

An equation that can describe the projection path of an x-ray beam at angle θ $p(t,\theta)$,

$$p(t,\theta) = \exp\left(-\int_{s(t)} \mu(s)ds\right) \tag{1}$$

where $s(t)$ is the path of the x-ray beam initiated at position t with respect to the s-t plane, $\mu(s)$ is the absorption coefficient along path s that is a translated version of $\mu(x,y)$ in the θ angle.

Imagine a row of rotating sensor array placed along the t direction, and the x-ray beam shines in the s direction through an object and reaching sensors on the other side. The result is an intensity value that can be described with the following discrete variation of Lambert Beer's Law.

$$I = I_0 \prod \exp\left(-\mu_i d_i\right) \tag{2}$$

After a complete scan, the resulting projection data is stored in an image-like grid known as the sinogram. These sinograms will serve as the input and be treated with a reconstruction algorithm to recreate the cross-section of the scanned object. The common practice of the reconstruction uses FBP or Fourier slice theorem, both have good efficiency. Iterative reconstructions, such as the ART, IRT, CGLS (Conjugate Gradient Least Squares), etc. are more time-consuming but often perform better in categories like improved reconstruction quality.

Concept of WSIRT

What is important about this concept is that one sweep through the object at a certain angle only constitutes one single column in the sinogram, yet every bit of the material is already accounted for by the beam traveling through, albeit collapsed from a two-dimension data down to one. That is to say, the sinogram accounts for a single spot in the material as many times as there are acquisitions, demonstrated more in detail in the Chapter 3: Apparatus section.

Photon starvation results from an accumulated high absorption from the material, but not being able to penetrate at one angle does not necessarily mean it is not present at a different angle. Perhaps the beam travels through less material in some angles, resulting in meaningful data at these but not other angles. Here is where this research differs from most other artifact reduction algorithms, instead of interpolating missing values, WSIRT attempts to fill them in with actual data from other acquisitions. The difference lies in where to sample for the missing data, general interpolation methods take values from neighboring pixels from the same acquisition, whereas WSIRT will use the projection data from a different acquisition that corresponds to the same general location to account for the vacancy.

WSIRT is a modified version of Simultaneous Iterative Reconstruction Technique (SIRT), and the variables will be defined similarly to that of SIRT. Let v (voxel) be the volume in which an object of interest resides. In the imaging setup, this is effective the image itself, stored as an array of $m^2 \times 1$ elements, m being the size of the image canvas. Let $p(t, \theta)$ be the projection data, corresponding to the data from a detector, and forming a sinogram described earlier. Let W_{ij} be the weight accounting for each voxel $u(t, \theta)$ a beam path traveled through. W_{ij} is stored as an $m^2 \times m \times n$ matrix, where m is the size of the image, and n is the number of acquisitions. The process of tomographic reconstruction is essentially solving the equation

$$W_{ij} v = p \tag{1}$$

Ideally, to find v, you want to simply solve $v^* = W_{ij}^{-1} p$, but W_{ij} is non-invertible. Consider a 512 by 512 image with a sensor resolution of 512 with 180 acquisitions. Trying to solve for 512x512 unknowns with 512x180 equation, that is extremely underdetermined. Instead, the focus of the method is to iterate trying to optimize the equation

$$v^* = argmin_v \left\| p - W_{ij} v \right\| \tag{2}$$

such that the measured projection data and the simulated projection data difference reach a minimum. Let I stand for the image acquired for reconstruction, then the general equation for the entire system of SIRT can be defined as the following:

$$I^{k+1} = I^k + C W^T R (p - W I^k) \tag{3}$$

where:

$$(a).\ R_{ii} = \frac{1}{\sum_{j=0}^{n-1} W_{ij}} \qquad (b).\ C_{jj} = \frac{1}{\sum_{i=0}^{m-1} W_{ij}} \tag{4}$$

R, C is the inverse of the row sum and column sum respectively, used to account for the beam path weight and normalize the final result.

The modified WSIRT introduces an extra weight shrinking variable into the equation, targets the impaired projection data while not inhibiting the algorithm's performance.

$$I^{k+1} = I^k + CSW^T R(p - WI^k) \tag{5}$$

S is the weight shrinking matrix used to account for the photon-starved pixels and shrinking their relative weights.

The primary difference between SIRT and SART (Simultaneous ART) is the order of operations. SART iterates through each projection for the set number of iterations, before moving on to the next, while SIRT completes all the projection angles, before moving on to the next iteration. An in-process visualization for SART would look like a clock rotating through and creating an image, where SIRT will look like a blurred image slowly becoming clearer.

CHAPTER 3

APPARATUS

Setup of the reconstruction experiment is done in the MATLAB R2019b. The formation of the reconstruction system utilized modular designs to separate the long equation into parts for the convenience of debugging. To display the experimentation set up as a software package, a documentation-like approach will be used here to demonstrate.

The apparatus accepts projection data, in a form of a sinogram, and outputs a fully reconstructed image. First, a weight matrix is calculated depending on the dimensions of the projection data. A specific weight matrix must be created for each different image resolution, and different rotational values for the projection acquisition. Next, the image is back-projected with the weight matrix to calculate the back projected image, saving it as the base for the output image. The result is then forward projected into a sinogram, and subtracted from the original projection to get the residual. That residual is back-projected and layered onto the resulting image, then forward projected to get the simulated projection. This process is repeated until the system converges or when it reaches the desired number of iterations.

Weighted Forward Projection → WI^k **WFP(W_{ij}, v, m, n)**

Returns the projection data of the image (forward projection), using the weighted matrix W. Inputs to the function are the weight matrix (W_{ij}), image voxel (I in the Equation 5, v here to represent the concept of voxels), image size (m), number of acquisitions (n). The output is represented in the figure as q to differentiate from the original sinogram p (projection data) since it is going to be changing with each iteration. See Figure 5 for visualization.

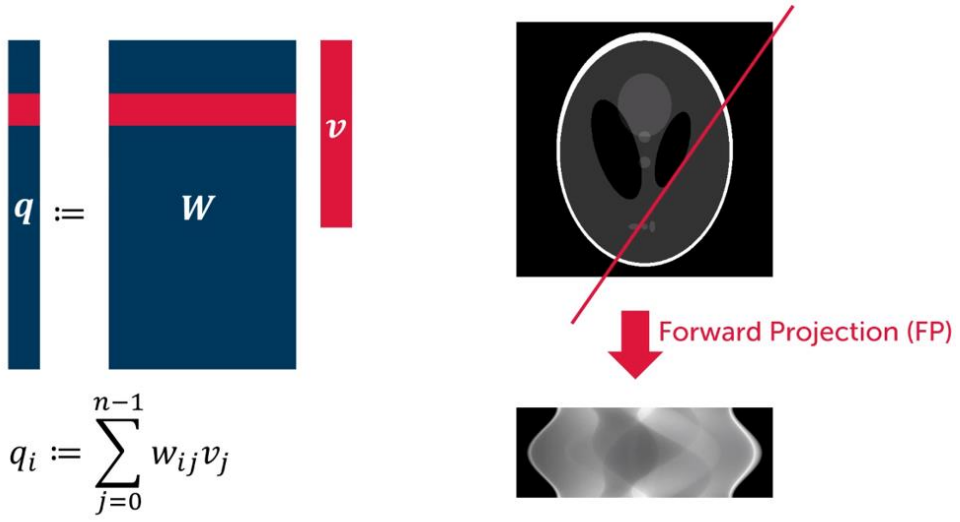


Figure 5. An illustration of the process of weighted forward projection. The system represents the data acquisition of x-rays passing through the material and received by the sensor to create a value within the projection data. The output is then subtracted from the original sinogram to find the difference.

Weighted Back Projection → $W^T q$ **WBP(W_{ij}, q, m, n)**

Returns the image voxels v corresponding to the input projection data, essentially a back projection process. Inputs to the function are the weight matrix (W_{ij}), projection data (q or the original sinogram p, as long as they are in the correct format for the projection data), image size (m), number of acquisitions (n). See Figure 6 for visualization.

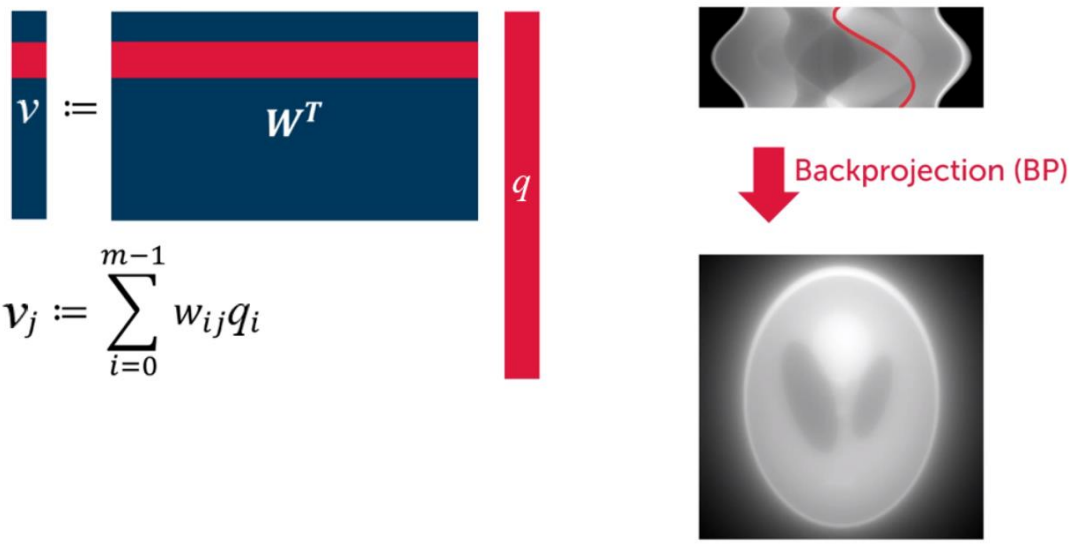


Figure 6[2]. An illustration of the process of weighted back projection. The system represents the smearing of projection data back onto the image canvas to reconstruct the original image. This process is repeated many times in the iterative method.

[2] Figure 6 and Figure 7 structures inspired by ASTRA Toolbox. http://www.astra-toolbox.com/docs/index.html

The weight shrinking component → $SW^T q$

The addition of weight shrinking matrix S is visualized in Figure 7.

Figure 7. The WBP with added shrinking coefficient S. It is added into each layer of acquisitions, checking individual projection data and automatically mapped into a matrix S, to account for potential photon starvation effect.

Figure 8, 9, and 10 explains how the weight shrinking coefficient is determined.

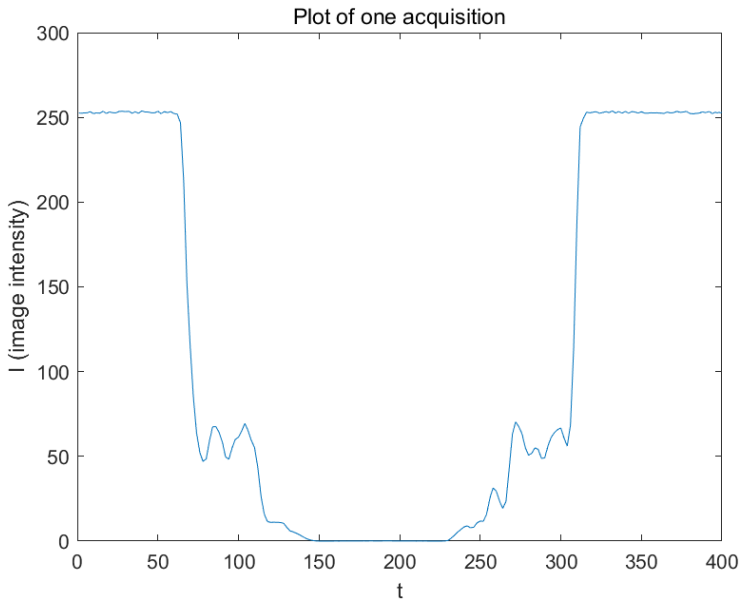


Figure 8. A row of sinogram data visualized with its pixel values with respect to the transformed s-t axis position t.

In this example, some of the projections clearly did not reach the sensor, creating photon-starved vacuums as identified by 0 image intensities, and even near-zero values are detrimental to reconstruction. Corresponding weight values for these projections may look something like this:

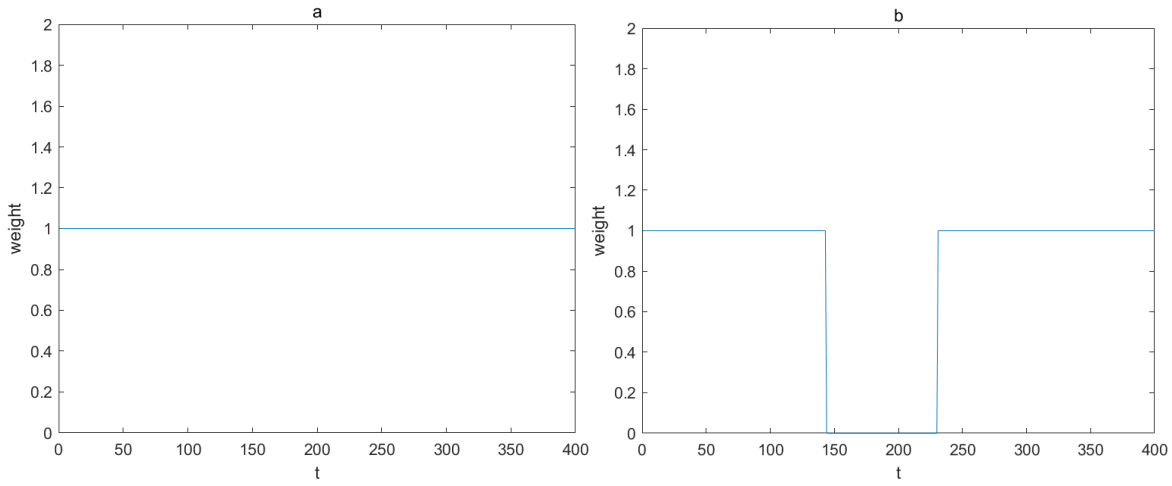


Figure 9. a is the initial weight distribution, assuming each x-ray projection traveled through equal amounts of materials. b is hard thresholding of the weight in this projection angle.

wherein the case of brutal hard thresholding, all of the weight of these projections not reaching the sensor is set to zero. In the conceptual sense, instead of reconstructing the image and pretending all the data is valuable, this weight coefficient discards beams that are not fully representative of the material absorption, and relies on other angles to account for the missing information.

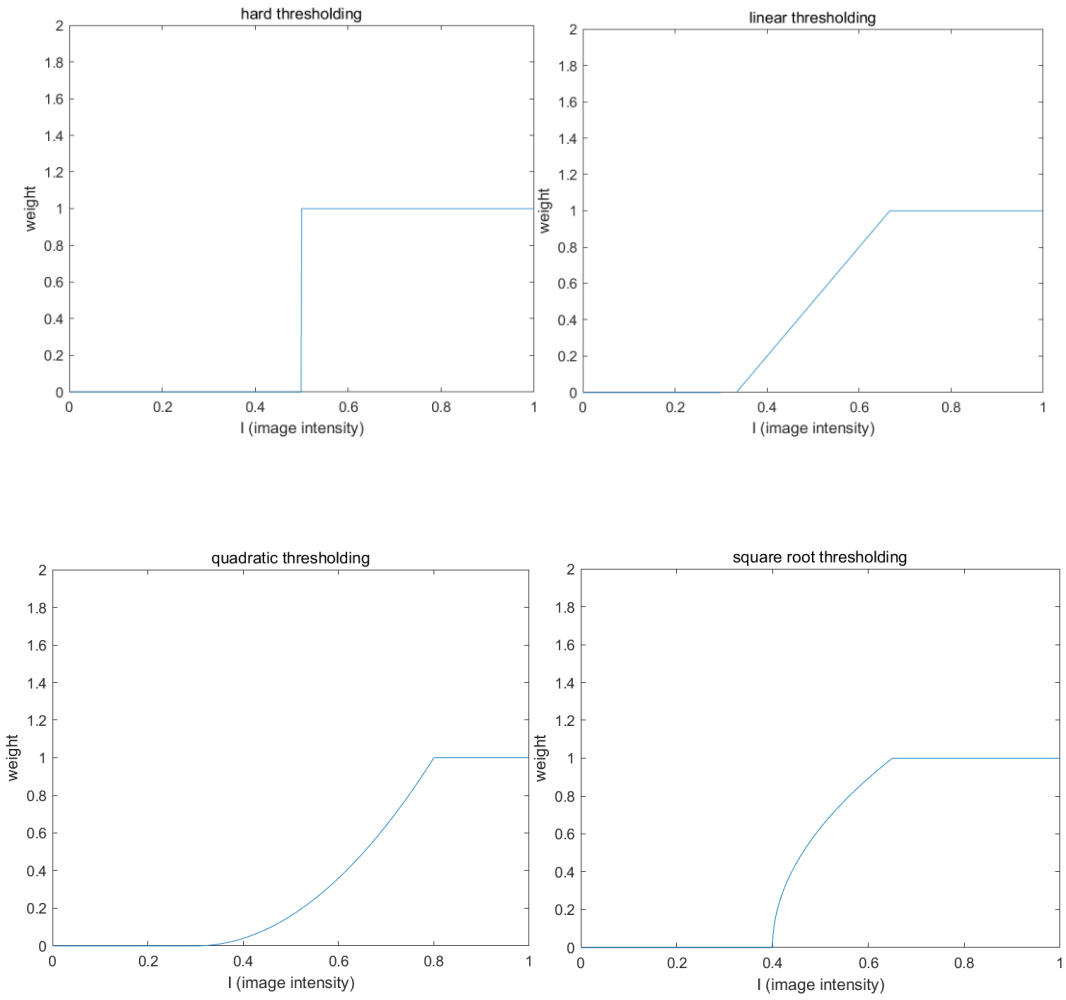


Figure 10. Graph representations of different types of methods in determining the weight shrinking factor, and its relative effect on the weight values. Image intensity here displayed as a relative value, mapped to a certain range in the actual data. Sinogram data received in intensity, mapped to general intensity scale (0-255) while keeping data integrity. An example of the range used to determine the weight shrinking factor would be around 0-5.

All of the methods displayed in Figure 10. are used to test the effect of the algorithm against different samples, their results are compared to find a superior way of determining the weight shrinking factor.

Weight Shrinking Iterative Reconstruction Technique **WSIRT(W_{ij},p, m, n, $iter$)**

Returns the reconstructed image I based on the input projection data p (sinogram). Inputs to the function are weight matrix W_{ij}, original projection data p, image size m, acquisition count n, number of iterations $iter$. A diagram of the WSIRT workflow logic is provided in Figure 11.

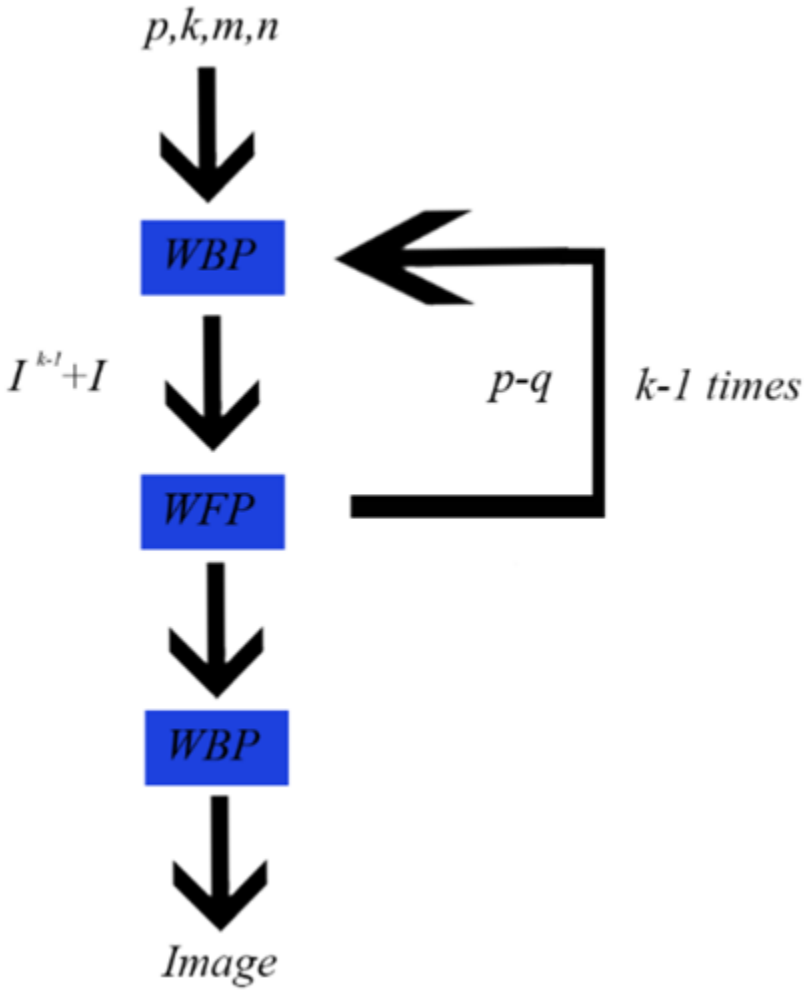


Figure 11. A diagram demonstrating the inside workflow logic of the WSIRT function, with inputs and outputs labeled.

Several background functions not mentioned in the loop above are[3]:

[3] Weights created parallel beams, with beam numbers equal to the number of pixels. Acquisition number and rotation accounted for in weight matrix creation, detailed codes included in the appendix.

LinearWij(m,θ)

Returns a slice of W_{ij} at the angle θ with size m. This method uses linear kernel to map the beam geometry onto the weight matrix.

getCompressedWij(m, n, $period$)

Returns a compressed sequence using a modified run-length encoded W_{ij}, to alleviate the repeated weight calculation in iterative algorithms. Returns a compact array that encompass all of the weight information for a specific period.

recW($tempW$, m)

Input a slice of compressed W_{ij} that represents the current acquisition's beam weights to reconstruct a full weight matrix. Returns essentially W_{ij} corresponding to the dimension m.

CHAPTER 4

RESULTS AND DISCUSSIONS

To determine the effectiveness of the algorithm, several samples and the corresponding cases for different shrink coefficients are tested. The used data samples are a crafted bone phantom, a 280x280 scan of a camera lens, and a 400x400 scan of a different lens. The bone phantom inside a water-filled centrifuge tube is acquired at 140kVp, with parts of the data artificially manipulated to simulate an extreme absorption area serving as an input. The two lens data are from camera lenses with plastic casing that have strong artifact streaks in reconstruction. Strong photon starvation effect samples are selected in hopes to help have a more visualized presentation. All of the scans are parallel beam scans, and the reconstruction system uses parallel beam weights accordingly.

Computational Cost

To have an estimate for the computational cost, each module has to be analyzed individually. For general iterative reconstruction techniques, the forward and backward projection steps are always the most time-consuming. Other steps generally only take several images worth of memory and simple data addition and subtraction. The computational cost comes from two major areas in the projection stage, the large parallel matrix operations, and the weight matrix construction.

The parallel operations generally involve a weight matrix size of $m^2 \times m$, an image voxel array of $m^2 \times 1$. The weight shrinking step additionally adds a weight shrinking matrix of $m \times m$. All of the above matrix operations are repeated n times representing each of the

acquisition angles, for each iteration. This alone is a considerable computational cost, and is the main reason the iterative algorithm isn't as widely used as simpler methods such as filtered back projections. However, nowadays as the GPU system is more and more developed, large parallel operations can be done efficiently on them compared to a general CPU, the time cost of this portion is significantly decreased if the right workload is applied.

The other portion of the cost comes from the weight matrix construction. Due to the nature of these data, a weight matrix is generally empty and has incredibly sparse data. The larger the image canvas, the less one beam of x-ray path is worth. Since each individual row of the weight matrix can be viewed as a piece of coordinate information for each beam path, it is necessary to conserve both the data and the location of the data. The result of these attributes is that the weight matrix can easily exceed 200 GB for a 512 by 512 image reconstruction. This unrealistic both for storage and run-time access, so different methods are created to circumvent the issue. With the development in computational power, thus the speed, a general approach is to calculate them live for each iteration. The problem is that the calculation is repeated since it cannot save the entire stack all at once. Not to mention the coordinate mapping calculation requires a heavy trigonometry presence. These sine and cosine values can be saved to a table to reduce calculation to just simple arithmetic, but the repetitive time cost remains.

The implementation of weight matrix construction used in this thesis is a run-length encoding type compression to save the weight matrix into a compressed form. At the cost of a half-iteration-worth period before running the algorithm, a weight matrix can be created and saved as a file. A single type file for a 400 by 400 canvas size weight file only takes around 330 Mb. The size of it depends on the canvas size, the number of acquisitions, and the angle of the acquisitions. Once the compressed weight matrix is created, it can be used for any reconstruction

of the same type defined at creation. This nature of the compressed weight matrix is prospected to be most valuable in 3D tomographic construction, where each slice of the reconstruction has the same acquisition angle, size, and counts. Since the experiments used in this thesis is purely 2D and have a different focus, a detailed analysis of the method's efficiency is not investigated.

Reconstructions

The following reconstructions used the same input for both SIRT and WSIRT algorithms, with an initial guess of an empty canvas, iteration count varies between different samples to ensure a reasonable reconstruction time. It is confirmed that a higher number of iterations in each sample does not provide any noticeable visual differences before selecting a value.

The first reconstruction is a bone phantom projection data manipulated to have the behavior of a high absorption region, reconstructed with 25 iterations shown in Figure 12. Figure 13 shows the reconstructions with the WSIRT method.

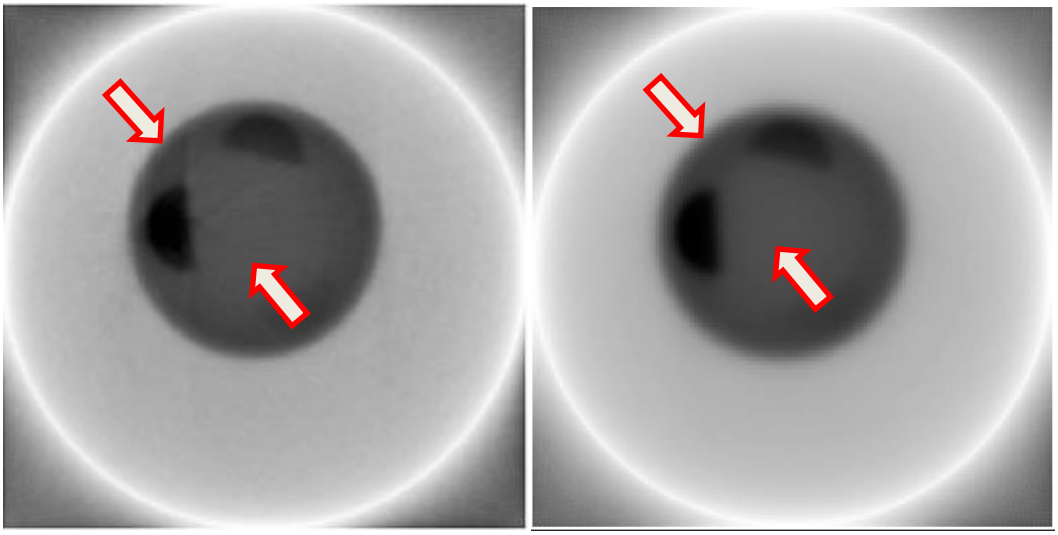
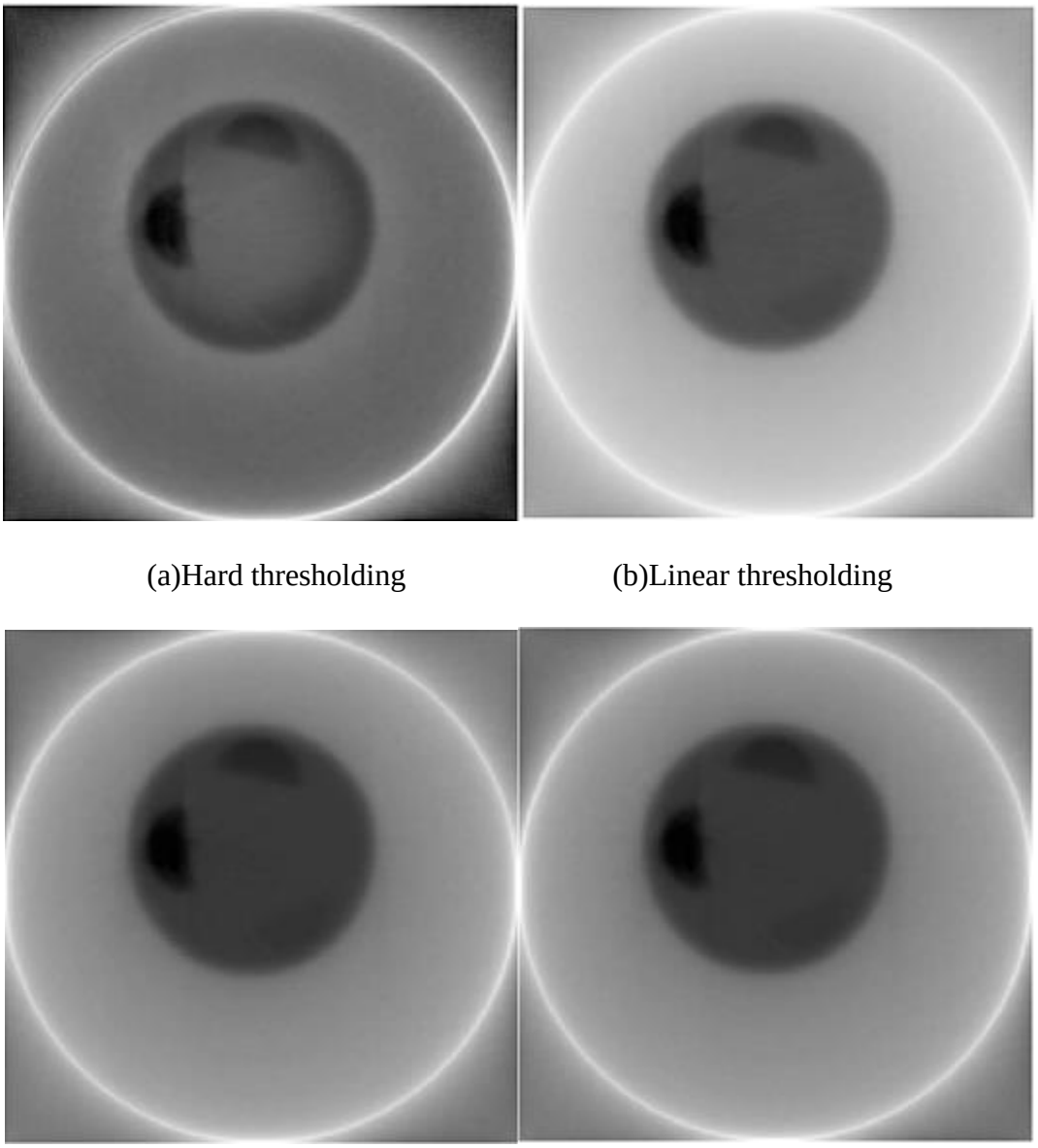


Figure 12[4]. The bone phantom reconstructed using SIRT, left is the modified version with streak artifacts observed around the densest piece, radiating in a ray-like fashion, right is the original projection reconstructed.

[4] Some figures are post-processed to improve contrast, in order to show the artifact streaks more clearly. Projection data acquired from Dr. Haidekker.



(a)Hard thresholding (b)Linear thresholding

(c)Square root thresholding (d)Square thresholding

Figure 13. WSIRT reconstruction with different weight shrinking factor determination methods on the bone phantom.

The next two reconstructions are both from non-altered real sinogram data. First of which is the cross-section of a plastic SRL lens (Figure 14, 15). The center region x-ray intensity drops below even one percent of the air value. This is the direct result of the high center accumulated absorption, making it have strong artifacts around its borders.

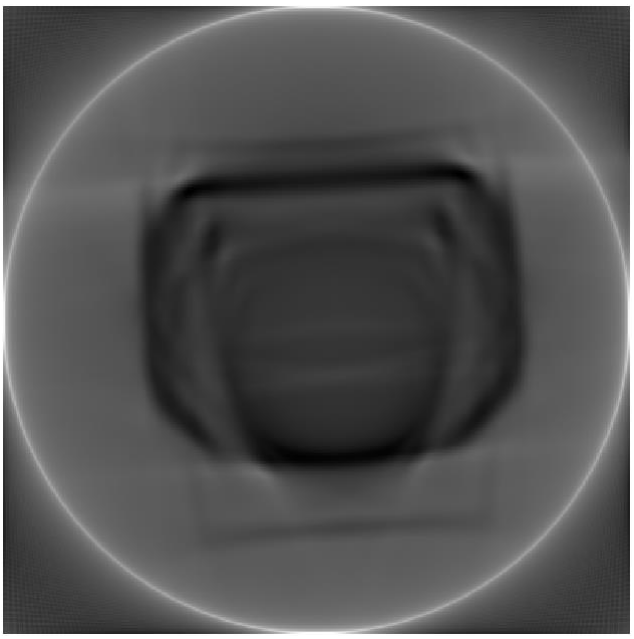
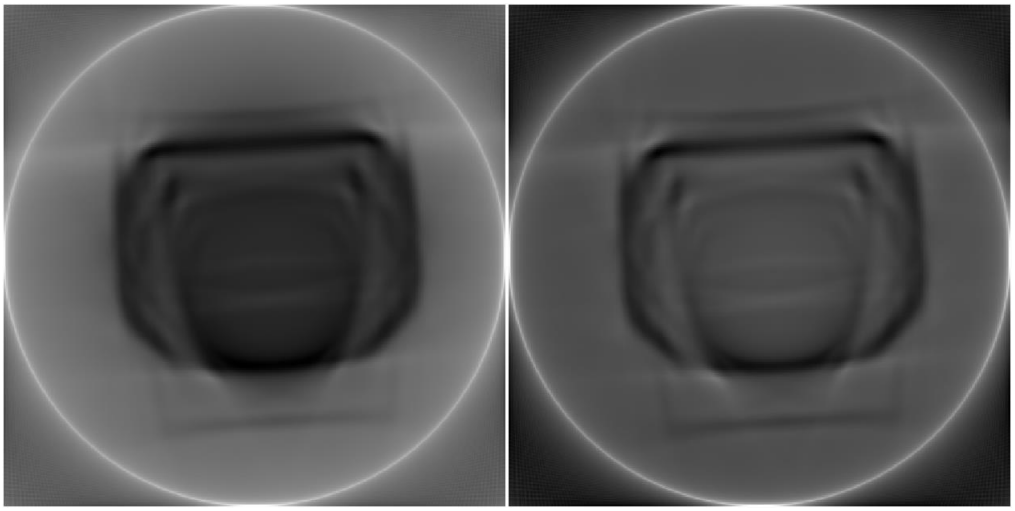
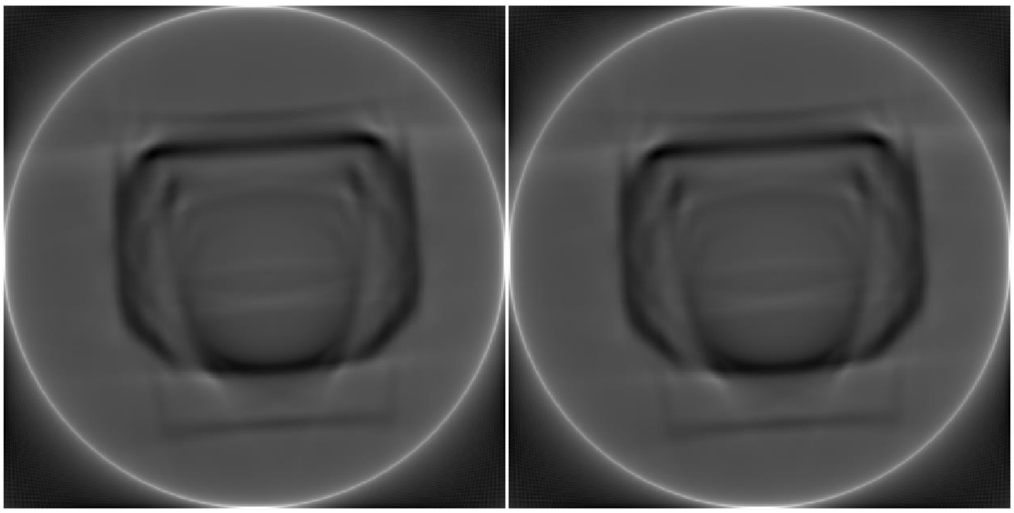


Figure 14. SIRT reconstruction of the SLR lens.



(a)Hard thresholding (b)Linear thresholding



(c)Square root thresholding (d)Square thresholding

Figure 15. WSIRT reconstructions with different weight shrinking factor determination methods on the SLR lens

The last reconstruction is of another lens (figure 16,17) that has metal-like behavior results. The streaks are prominent and inspired many interesting variations to the algorithm and theories to their phenomenon.

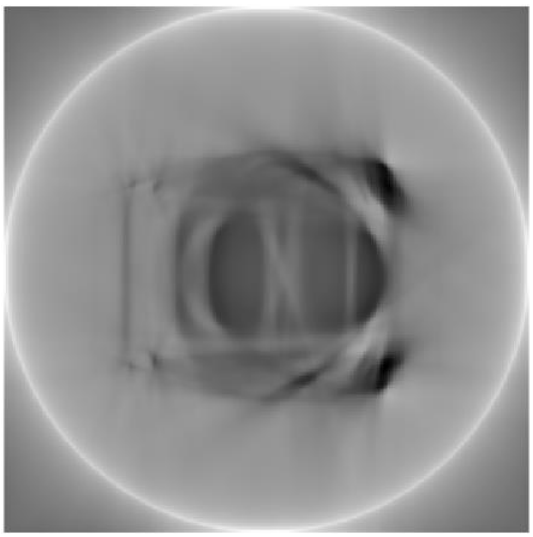
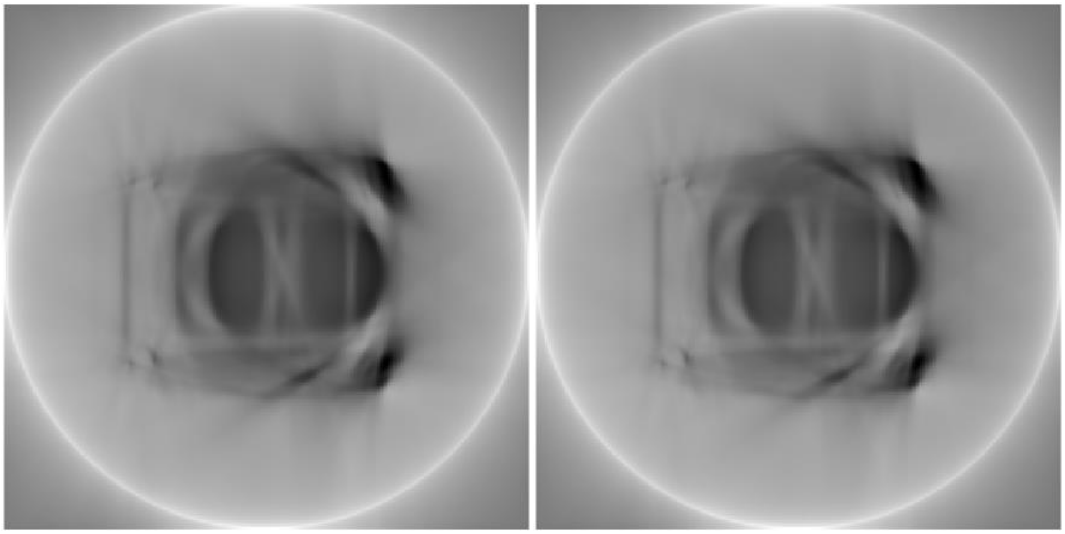


Figure 16. SIRT reconstruction of the lens.



(a)Hard thresholding (b)Linear thresholding

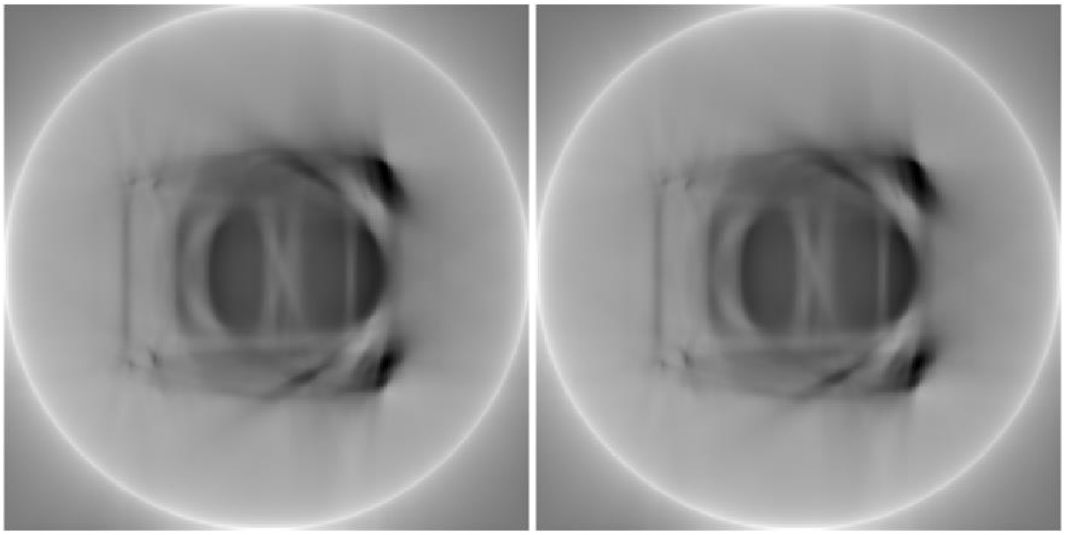


Figure 17[5]. WSIRT reconstructions with different weight shrinking factor determination methods for the lens.

[5] This set of data have suffered minor data type conversion losses, but the effect on data integrity is estimated to be less than one percent, thus should not affect statistical analysis in any major fashion.

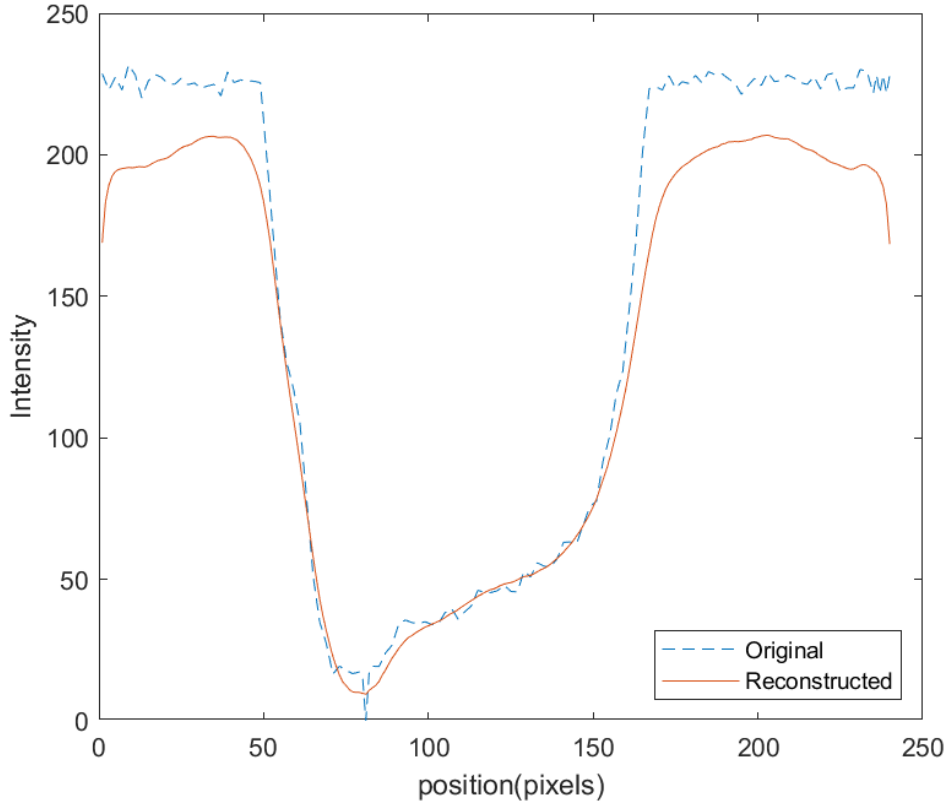


Figure 18. The horizontal intensity profile through the reconstructed image compared to the original projection data.

The graph in Fig. 16 presented the intensity profile through the reconstructed image, forward projected through a simulated projection algorithm. Image intensity is effectively the inverse of the absorption, representing the materials scanned. The figure used projections from hard thresholding WSIRT phantom reconstruction and provided effective information fidelity with smoothing/blurring effects. All of the reconstructions displayed similar features, despite the type of method or sample.

To present the effect in a meaningful way, a data analysis is performed with a specific region of interest as shown in Figure 19.

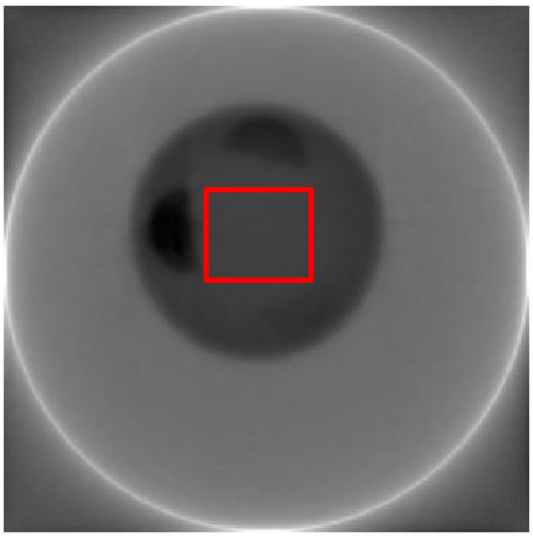


Figure 19. An example of the selected region of interest, where the materials are expected to be homogeneous.

These regions are selected to be able to reflect the effectiveness of the algorithms. Several different methods are tested to have a relationship or represent the amounts of artifacts present. In homogeneous regions of the image, any variations should be the cause of the photon starvation artifacts. Two major methods are selected to present the data, the signal-to-noise ratio (SNR) and pixel counting. SNR is estimated by the mean value of the region divided by the standard deviation (STD).

$$SNR = \bar{I}/SD(I) \tag{6}$$

In the pure homogenous region, the lower the SNR, the higher presence of the streak artifact remains. Pixel counting is the method to estimate the area of streak artifacts. Since the area should be homogeneous, any pixel intensity that varies out too far from the mean can be the

streak presence. Thresholds for pixel counting are picked out by probing image values of streaks and surrounding areas. Value is selected to be the same within each group of images, but different for each group. This method may be a bit inaccurate due to the nature of these streaks having a full spectrum of pixel intensities, but demonstrates a clear comparison to those who are not interested in digging into the mathematical representations of each term.

Table 1. The processed data of the selected region of interest in each sample. Each sample is reconstructed five times with different methods. SNR and pixel counting are the two methods selected to have a parallel comparison of the effect in different reconstruction methods along with different samples. The region selected for each sample is not the same size, thus a large variation on the pixel counting analytics, but within each sample the region selected is identical.

Object Name	Method	SNR	Artifact area
Bone phantom	SIRT	28.42	372
	WSIRT Hard	29.58	348
	WSIRT Linear	29.88	350
	WSIRT Square root	28.97	351
	WSIRT Square	29.04	356
Lens 1	SIRT	7.65	1394
	WSIRT Hard	8.99	1211
	WSIRT Linear	9.50	1334
	WSIRT Square root	9.58	1348
	WSIRT Square	9.50	1352
Lens 2	SIRT	14.43	1069
	WSIRT Hard	16.73	1018
	WSIRT Linear	15.31	1012
	WSIRT Square root	14.38	1068
	WSIRT Square	15.31	1027

It can be observed that within each sample in table 1, the WSIRT reconstruction showed a general trend of improvement. A higher SNR value is the effective none-noise signal in the homogeneous region and a higher pixel count in the area of streak coverage with the unit in pixels. With the experiments presented in this work, very little difference can be determined between each different weight shrinking factor determination method. Several variations exist between different samples for which one provides the best quality, but the result may have been influenced by the specific sample data and its canvas size. The square root approximation, in theory, should be the closest to the original, and hard thresholding being the furthest. This is due to the normalization from the weight matrix creation, where each value is normalized to below one. The maximum voxel weight is one, to represent the entire voxel is covered by a specific beam path. These sub-one values provided a base so that the square root should result in the largest value compared to squaring, remaining the same, and flat out zero. With that, it is easy to understand how hard thresholding should reflect the most effect when the iteration is low. However, these harsh weight shrinking effects are usually nullified after a few reconstructions, and any observable difference by the human eye is quickly hidden by the nature of supplementary information from other unweighted projections.

A theory of principal beam weight is developed to account for the mismatch in some of the method results. In the weight matrix, a beam path is discretized and represented by voxel weights. These values are generally unevenly distributed since a beam path does not usually perfectly cover a voxel in the image. Each voxel of the beam path is remapped onto the weight matrix to represent how much of the material a beam is passing through. The middle voxel, named principal beam weight, is a much larger proportion compared to the side voxels, call it secondary beam weights. Since all of the weights are below one, a near one decimal value, for

example, 0.9, will react very differently to a smaller decimal value, for example, 0.1, when dealing with quadratic operations. It is projected that this effect will not only shrink the particular beam's weight but also "sharpening" it, making the middle portion of the weight much more compared to its outer edges. The effect of this is not yet fully determined and would behave differently with different weight matrix creation methods.

Another interesting find is the masking effect displayed in Figure 20. In early iterations where the weight shrinking effect is harsh and unfit for a result image, its characteristics can be manipulated for certain post-processing procedures. For example, the hard thresholding where the weights of unwanted pixels are cut off completely, and it has not been filled in with other projection data. These harsh lines can be used as a mask to alleviate the effect and have other interesting features such as separating materials and background.

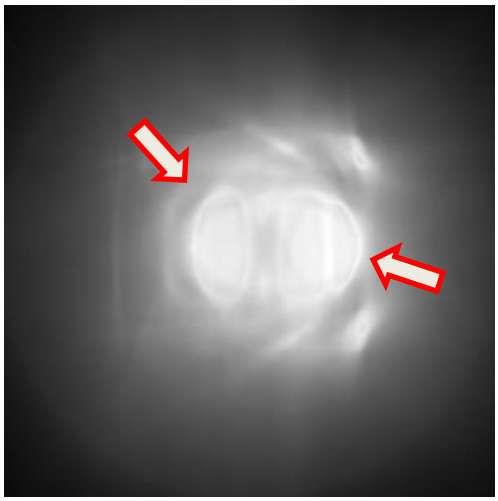


Figure 20. An example of the masking effect that has an edge detection-like effect. The inside structures are brightened and outlined by the weight shrinking mask.

This effect seems promising; however, the usage may be niche. The masking is mathematically equivalent to weight each image pixels, and does not bring in any additional information. Thus, it is a form of interpolation, it may make the result look better, but does not provide information

about hidden structures that may be covered by the artifacts. That being said, it can be an interesting mask for postprocessing images for other uses, and will not be elaborated in this thesis.

CHAPTER 5

CONCLUSIONS

This thesis aimed to present a solution to suppress photon starvation artifacts that stands as a barrier to more effective medical advancements. The method developed and tested in this thesis provided promising results and a direction where artifact reduction algorithms can advance.

Many artifact reduction algorithms have attempted to reduce the projection data lost effect through different directions. Despite the abundant proposed solutions in the postprocessing category, it is evident that the solution may very well come from the reconstruction algorithms. The theory behind tomographic reconstruction is an estimator and an optimizer. Collapsed and missing information cannot be correctly determined but only approached. Augmented iterative algorithms are a promising lead towards nullifying the artifacts in reconstructions.

The downside of this method is a common one shared by any iterative method: computational costs. The constructed system in this work takes several hundred more times to produce an image compared to conventional FBP algorithms. Although the weight shrinking component only adds roughly five to ten percent on top of that, it is still much too expensive for general clinical practices. The good news is that with developing graphics processing units (GPU), large parallel structured computation is more efficient than ever. Several groups have implemented GPU accelerated reconstructions [11], and can reduce the reconstruction time down to seconds. When matrix operation is no longer an issue, the only problem left is the cost to transfer data between the CPU and the GPU. With a code optimized for this structure, the

iterative algorithms can be the future of tomography, even in 3D constructions where large numbers of slices are required for reconstructions.

The present study only confirmed a minor increase in the effect of reducing photon starvation artifacts, and that points out the directions of potential future work. Currently, the sample tested is not very well constructed for the problem, and larger sample size can always help confirm trends and theories. Specially crafted samples may help to identify the effect of retrieving hidden material structures behind artifacts, and even expanding its capabilities. The other main issue is the machine limitations. Resolution of the reconstruction is limited by the resolution of the scanner, and more intricate sensor setup, more sophisticated machine can result in much better quality (resolution) projection data. With better resolution, resulting in more pixels to perform analytical research on.

The biggest strength in artifact reduction algorithms is their ability to supplement each other. Every proposed algorithm targets a different part of the problem, takes a different approach to tackle the issue. For example, previously introduced algorithms that interpolate different projection angle data to fill missing slots, and frequency domain filters in defining structures. All of these creations are aimed to solve a particular problem, but in the clinical perspective, every individual sample creates a different problem. With the combination of existing and future methods, the case-by-case solution may very well be presented. Every method proposed in the field is a step towards a comprehensive solution, and this method aims to be just another step towards that achievement.

REFERENCE

[1] J. Radon, "On the determination of functions from their integral values along certain manifolds," *IEEE Transactions on Medical Imaging,* vol. 5, no. 4, pp. 170-176, Dec 1987.

[2] I. Khodarahmi, A. Isaac, E. K. Fishman, D. Dalili and J. Fritz, "Metal About the Hip and Artifact Reduction Techniques: From Basic Concepts to Advanced Imaging," *Semin Musculoskelet Radiol,* vol. 23, no. 3, pp. 69-81, 2019.

[3] L. Gjesteby et al., "Metal Artifact Reduction in CT: Where Are We After Four Decades?," *in IEEE Access,* vol. 4, pp. 5826-5849, 2016.

[4] Y. M. Levakhina, J. Müller, R.L. Duschka, F. Vogt, J. Barkhausen, and T.M. Buzug, "Weighted simultaneous algebraic reconstruction technique for tomosynthesis imaging of objects with high-attenuation features," *Med. Phys*, vol. 19, no. 3, 2013.

[5] Ge Wang, D. L. Snyder, J. A. O'Sullivan and M. W. Vannier, "Iterative deblurring for CT metal artifact reduction," *in IEEE Transactions on Medical Imaging*, vol. 15, no. 5, pp. 657-664, Oct. 1996.

[6] S. Zhao, D. D. Robeltson, G. Wang, B. Whiting and K. T. Bae, "X-ray CT metal artifact reduction using wavelets: an application for imaging total hip prostheses," *in IEEE Transactions on Medical Imaging*, vol. 19, no. 12, pp. 1238-1247, Dec. 2000.

[7] A. Mehranian, M. R. Ay, A. Rahmim and H. Zaidi, "X-ray CT Metal Artifact Reduction Using Wavelet Domain L_{0} Sparse Regularization," *in IEEE Transactions on Medical Imaging,* vol. 32, no. 9, pp. 1707-1722, Sept. 2013.

[8] S. Hao, J. Liu, Chen, Y. Chen, B. Liu, C. Wei, J. Zhu and B. Li, "A wavelet transform-based photon starvation artifacts suppression algorithm in {CT} imaging," *Physics in Medicine & Biology,* vol. 65, Dec. 2020.

[9] A. R. Podgorsak, M. M. Shiraz Bhurwani and C. N. Ionita, "CT artifact correction for sparse and truncated projection data using generative adversarial networks," *Medical Physics,* vol. 48, no. 2, pp. 615-626, 2021.

[10] S. G. Moon et al., "Metal artifact reduction by the alteration of technical factors in multidetector computed tomography: a 3-dimensional quantitative assessment," *J Comput Assist Tomogr,* vol. 32, no. 4, pp. 630-3, Jul.-Aug. 2008.

[11] Y. Du, G. Yu, X. Xiang, et al. "GPU accelerated voxel-driven forward projection for iterative reconstruction of cone-beam CT". *BioMed Eng OnLine,* vol. 16, no. 2, 2017.

[12] B. De Man, J. Nuyts, P. Dupont, G. Marchal and P. Suetens, "Metal streak artifacts in X-ray computed tomography: a simulation study," *in IEEE Transactions on Nuclear Science,* vol. 46, no. 3, pp. 691-696, June 1999.

[13] P. Sukovic and N. H. Clinthorne, "Penalized weighted least-squares image reconstruction for dual energy X-ray transmission tomography," *in IEEE Transactions on Medical Imaging,* vol. 19, no. 11, pp. 1075-1081, Nov. 2000.

[14] R. M. Rangayyan and R. Gordon, "Streak Preventive Image Reconstruction with ART and Adaptive Filtering," *in IEEE Transactions on Medical Imaging,* vol. 1, no. 3, pp. 173-178, Nov. 1982.

[15] L. Landweber, "An Iteration Formula for Fredholm Integral Equations of the First Kind.," *American Journal of Mathematics,* vol. 73, no. 3, pp. 615-24, 1951.

[16] P. Lim, J. Barber, and J. Sykes, "Evaluation of dual energy CT and iterative metal artefact reduction (iMAR) for artefact reduction in radiation therapy," *Australas Phys Eng Sci Med* vol. 42, pp. 1025–1032, 2019.

[17] G. K. Yadava, D. Pal and H. Jiang, "Reduction of metal artifacts: beam hardening and photon starvation effects," in *Medical Imaging 2014: Physics of Medical Imaging*, vol. 9033, B. R. W. a. C. Hoeschen, Ed., SPIE, 2014, pp. 816-823.

[18] M. A. Haidekker, "Optical transillumination tomography with tolerance against refraction mismatch," *Comput Methods Programs Biomed,* vol. 80, no. 3, pp. 225-35, Dec. 2005.

[19] N. Haramati et al., "CT scans through metal scanning technique versus hardware composition," *Comput Med Imaging Graph,* vol. 18, no. 6, pp. 429-34, Nov.-Dec. 1994.

[20] B. Hamelin et al., "Iterative CT reconstruction of real data with metal artifact reduction," in *2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, Paris, France, 2008.

[21] T. M. Svahn and N. Houssami, "Evaluation of time-efficient reconstruction methods in digital breast tomosynthesis," *Radiation Protection Dosimetry,* vol. 165, July331-6 2015.

[22] A. C. Kak and M. Slaney, Principles of Computerized Tomographic Imaging, *IEEE Press*, 1988.

APPENDIX

Codes

```
%Obtain one slice of the weight at one angle, output in sizexsize^2 matrix
function [W] = linearWij(m,theta) %Obtain one slice of the weight at one
angle, output in sizexsize^2 matrix
tempp = 0;          %p is the axis that the ray (projection) traveled
through.
tempt = 0;          %t lines up with the sinogram horizontal axis
tempx = 0;
tempy = 0;
halfm = m/2;
W = zeros(m,m^2);

    for j = -halfm+1:halfm
        for k = -m+1:m
            tempx = k*cos(theta) + j*sin(theta);
            tempy = -k*sin(theta) +  j*cos(theta);
            temploc = (floor(tempy)+halfm-1)*m+floor(tempx)+halfm;

            if tempx<-halfm+1||tempx>halfm||tempy<-halfm+1||tempy>halfm
            else
                if temploc<1||temploc+m+1>m^2
                else
                    W(j+halfm,temploc) = W(j+halfm,temploc) + tempx-
floor(tempx);
                    W(j+halfm,temploc+1) = W(j+halfm,temploc+1) + 1-(tempx-
floor(tempx));
                    W(j+halfm,temploc+m) = W(j+halfm,temploc+m) + tempy-
floor(tempy);
                    W(j+halfm,temploc+m+1) = W(j+halfm,temploc+m+1) + 1-
(tempy-floor(tempy));
                end
            end
        end
    end
end

%get compressed Weight. Input m = image size (one side), n = number of acq
%period: pi = 180, 2*pi = 360 etc.
function [W] = getcompressW(m,n,period)
count = 0;
m2 = m^2;
W = single(zeros(n,m2));
for i = 1:n
    theta = (i-1)/n*period;             %change haven't confirmed for n
    tempW = linearWij(m,theta);
    tempW = tempW./2;
```

```
    step = 1;
    for j = 1:m
        for k = 1:m2
            if tempW(j,k) == 0
                count = count +1;
            else
                if count ~= 0
                    W(i,step) = count;
                    step = step + 1;
                    W(i,step) = tempW(j,k);
                    step = step +1;
                    count = 0;
                else
                    W(i,step) = tempW(j,k);
                    step = step + 1;
                end
            end
        end
        if count~=0
        W(i,step) = count;
        step = step + 1;
        count = 0;
        end
    end
end


end


%WSART (Weight Shrinking ART)
%W = compressed weight matrix, they will be decompressed in WBP and WFP
%when needed
%sino = sinogram
%iter = how many iterations: 1 iter = 1 WBP + 1 WFP (no last WFP since we
want an image output not sinogram)
%m = image size (only accepts square image)
%n = number of acquisitions (half or full circle accounted for by the weight
matrix)
function I = WSIRT(W,sino,m,n,iter)
    I = zeros(m);
    sino = Normalize(sino);
    sino = abs(sino-255);
    sino_diff = sino;
    for i = 1:iter-1
        tic
        I_update = WBP(W,sino_diff,m,n);
        toc
        I = I + I_update;
        sino_remain = WFP(W,I,m,n);
        sino_diff = sino - sino_remain;
    end
    I_update = WBP(W,sino_remain,m,n);
    I = I + I_update;
end
```

```
function N = Normalize(IMG)
    N = (IMG-min(min(IMG))).*(255/(max(max(IMG))-min(min(IMG))));
end


%weighted forward projection, input weight, image, size m, number of acq n.
function sino = WFP(W,I,m,n)
tempI = reshape(I',m^2,1);
sino = zeros(m,n);
for i = 1:n
    tempsino = zeros(m,1);
    tempW = recW(W(i,:),m);
    rowsum = sum(tempW');
    rowsum = rowsum';
    rowsum(rowsum == 0) = 1000;
    tempsino = tempW*tempI./rowsum;
    sino(:,i) = tempsino;
    %imshow(sino,[]);   %for real time visualization
end
sino = sino';
end




%weighted backprojection input weight, sino, number of acq n, and size m to
get one backprojection
function I = WBP(W,sino,m,n)
I = zeros(m^2,1);

for i = 1:n
    tempI = zeros(m^2,1);
    tempW = recW(W(i,:),m);
    colsum = sum(tempW);
    colsum = colsum';
    tempW = tempW';
    tempS = sino(i,:)';
    %==============================
    Wshrink = zeros(m,1);
    wMax = max(tempS);
    wMin = min(tempS);
    wRange = wMax-wMin;
    for j = 1:m
        if tempS(j,1) <= wMin+wRange*.05
%           Wshrink(j,1) = (tempS(j,1)-wMin)/wRange;
          Wshrink(j,1) = ((tempS(j,1)-wMin)/wRange)^2;
%           Wshrink(j,1) = sqrt((tempS(j,1)-wMin)/wRange);
%           Wshrink(j,1) = 0;
        else
            Wshrink(j,1) = 1;
        end
    end
    Wshrink = Wshrink';
    tempW = tempW.*Wshrink;
    %==============================
```

```
    colsum(colsum==0) = 1000;
    tempI = tempW*tempS./colsum;
    I = I + tempI;
%===================================
%     %uncomment for visualization
%     Ishow = reshape(I,m,m);
%     imshow(Ishow,[]);
%===================================
end
I = reshape(I,m,m);
I = I/n;
end


%input a slice of the compressed Weight matrix to reconstruct a slice
%of the actual Wij

function W = recW(tempW,m)
W = zeros(m,m^2);
i = 1;              % stepper stepping through tempW, the input compressed
weight slice goes from 1 to size(tempW) or whenever it hits 0
count = 1;          % counter for the x in W, going from 1 to m^2
y = 1;              % counter for the y in W, going from 1 to m
m2 = m^2;
sizeW = size(tempW);
while tempW(i) ~= 0
    if tempW(i) < 1
        W(y,count) = tempW(i);
        count = count + 1;
    else
        for j = 1:tempW(i)
            W(y,count) = 0;
            count = count +1;
        end
    end
    i = i + 1;
    if count > m2
        count = 1;
        y = y+1;
    end
    if i >= sizeW(2)
        break
    end
end
end
```