Image Restoration of Medical Images with Streak Artifacts by Eulers Elastica Inpainting

by

Xiaochen Zhang

A research paper
presented to the University of Waterloo
in partial fulfillment of the
requirement for the degree of
Master of Mathematics
in
Computational Mathematics

Supervisor: Prof. Justin W.L. Wan

Waterloo, Ontario, Canada, 2016

I hereby declare that I am the sole author of this report. This is a true copy of the report, including any required final revisions, as accepted by my examiners.

I understand that my report may be made electronically available to the public.

Abstract

Streak artifacts caused by unavailable metal implants degrade the CT image's quality and disturb medical diagnose. The common methods used to reduce these metallic artifacts often consists of interpolation or iteration methods. The former tend to lose image quality by introducing extra artifacts, and the latter costs heavy computational time. This paper proposes a new strategy based on Euler's elastica inpainting method, which can preserve the curvature of the original image, making the restoration results more continuous and complete. Results of quantitative and qualitative experiments on both simulated phantoms and clinical CT images demonstrate that our method can suppress metal artifacts significantly.

Acknowledgements

I would like to greatly appreciate my supervisor, Prof. Justin W. L. Wan. Without his large support and guidance, this work could not be possible. I would also like to thank Prof. Jeff Orchard, who helps reading this paper and gives me valuable advice. Particularly, I want to thank my lovely friends in the CM program, as well as all the people who give me kindly help during this memorable year in the University of Waterloo.

Dedication

This is dedicated to my dear parents and the ones I love.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Computed Tomography (CT) has been widely used as a medical diagnostic tool over the years since it was first invented in 1972 [19]. CT images show clear and reliable information about body structures, bones and soft tissues in the interested regions of the examining body. However, metal artifacts, which commonly appear in CT images, will severely degrade the quality of CT images. The formation of artifacts is related to the metallic parts inside the body. Metal is usually a high-attenuating material, which will disturb the X-ray propagation while passing through the human body. As a result, light and dark artifacts originating from the metal piece will spread out the whole image, causing serious damage to the information provided by CT images. It has been reported that about 10% of the CT images are affected by metal artifacts to different degree [11]. This will cause misdiagnosis for doctors, which largely limits the application of CT imaging. Therefore, the research about how to reduce the artifacts is of significance.

Figure 1.1 illustrates how metal artifacts affect the visual quality of CT images. They are CT images with metallic objects in head, hip and tooth. In the first image of 1.1, the metal objects are inserted at two knees, and there are straight artifacts crossing on the image. The second and third images have one metallic object in the middle, and radioactive artifacts are emitted from the metal object. In the second line of 1.1, we can notice that dramatic white lines are gathering around the region of teeth. The quality of the CT images has been severely damaged by the streaking artifacts.

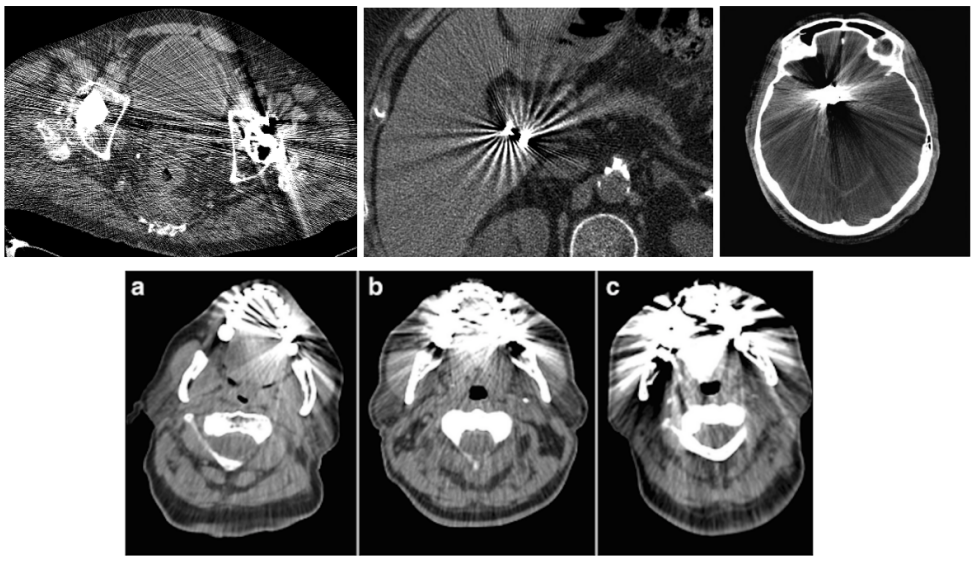

Figure 1.1: Streak artifacts in head, hip and tooth CT images

1.1 Metal Artifacts in CT Radiology

The formation of metal artifacts is closely related to the principle of CT imaging. When a patient is under going a CT scanning, X-ray beams carrying different levels of energy will pass through the body from all directions. For an object containing tissues with different density and thickness, the attenuation to the X-ray beams is not uniform. High-density tissues tend to absorb more high-energy photons thus becoming harder in the final image. This effect is commonly referred as beam hardening effect, or cupping effect. Because of this selective absorption character, the collected projection data no longer follows the linear relation as before. When there exists strong attenuating objects, typically metallic objects, the beam hardening effect becomes so strong that severely breaks the propagation process and results in artifacts.

After projection, the collected data are stored as a matrix with each column corresponding to one direction of the projection. The matrix shown as a two dimensional image is called raw data, or sinogram. Then, a filtered back-projection (FBP) algorithm will be used to reconstruct the raw data shaping into a two-dimensional image, which is the final CT image we are familiar with. FBP algorithm assumes a linear propagation to these collected projection data and re-project them in each column to recover the shape of the

original object. A detailed description of how FBP works is demonstrated in the latter chapter. Basically, it is the missing information in the raw data leads to the artifacts after a reconstruction in the final CT images.

1.2 Metal Artifact Reduction Development

The problem about metal artifacts on the CT images starts to appear from the beginning when CT technique was invented. Over the decades, there have been a large variety of methods to tackle this problem. All these approaches are referred as metal artifact reduction (MAR) methods. Generally speaking, they can be categorized into two types: projection completion methods, and statistical algebraic methods.

The first projection completion based method was invented in [22]. The main idea of this type of approach is to replace the corrupted pixels in the raw data with proper values calculated from the surrounding information. The specific approaches used to predict missing values are various. Linear interpolation [20] is one of the most frequently used methods among those, due to its simplicity and fast speed. Later on, there are cubic interpolation [3], polynomial interpolation [2], and wavelet interpolation [36]. Basically, all the interpolation techniques can be used for prediction. Besides traditional interpolation methods, there are some other ways trying to make use of the surrounding pixels of the objective object, such as pins at the opposite or an optimal direction. Moreover, some image preprocessing techniques have been introduced into this problem and used as one step before implementing interpolation algorithms. Among these, one of the most efficient approaches is to take advantage of a prior image. To calculate the prior image, a segmentation is applied on the original image in order to separate different types of tissues [6]. Making use of the prior image can reduce the possibility of choosing improper values when applying interpolation. One of the most popular and successful methods using a prior image is called normalized MAR (NMAR) [26]. In their experiments, this method outperformed most of the one dimensional interpolation methods and largely fixed the missing information in the corrupted raw data.

While most of the MAR methods belong to the projection completion based methods, they have significant drawbacks. One of the major problem is that most of the interpolation methods use one dimensional interpolation, and limited information can be taken advantage of. The inconsistency and mis-prediction problems result in new artifacts in the reconstructed image after a FBP reconstruction. Subsequently, another type of approach has been proposed to avoid these problems arisen during FBP reconstruction process.

The statistical iterative approach uses the idea of treating the reconstruction as an optimization problem. More precisely, an initial image is guessed, followed by a forward projection. The original sinogram is then divided by the estimated sinogram values accordingly [13]. The goal of the iteration is to obtain the "best" estimation based on the restored result from last time. The best known example of statistical iterative reconstruction technique is the maximum likelihood-expectation maximization (ML-EM) algorithm. From theoretical analysis, the statistical iterative method can obtain a "perfect" recovery of the original un-affected sinogram with infinite iteration times. Accordingly, the main problem to this type of methods is their extremely high computational time and heavy dependence on the CT equipment characteristics [4].

In this thesis, we present a novel approach to reduce metal artifacts both preserving the consistency of the whole image. This method uses inpainting technique, more specifically Euler's elastica and curvature-based inpainting to fill in the corrupted domain such that the sinogram image looks consistent and continuous. If the sinogram is better restored, then after reconstruction, the final corrected image will be better.

The layout of this thesis is as follows. In Chapter 2, we will introduce the background of the current MAR methods and then the mathematical fundamental of Euler's Elastica inpainting. Detailed numerical implementation formula will also be given. In Chapter 3, a methodology of our model and how we implement the method on medical artifact image will be proposed. Chapter 4 shows the experiment and results with different images as well as the comparison between different methods. A conclusion will be provided in Chapter 5.

Chapter 2

Background

In this chapter, we will first introduce some current MAR methods, and discuss their principles, advantages and disadvantages. Then we will introduce and explain Euler elastica inpainting. A brief review of how inpainting works and develops will also be given. Finally, we will describe the formulation and discretization of the equations.

2.1 Projection Completion MAR Methods

Before inpainting, our raw image data are 2-D images corrupted with artifacts. The artifacts on the image are caused by multiple mechanisms, including the beam hardening, scatter, noise, motion, and edge effects. Among those, beam hardening is one of the most prominent sources.

Beam hardening effect is caused by high-density materials in the body. When a bunch of X-rays carrying beams with different level of energy passes through the scanning body, low-energy beams are easier to be absorbed. To be specific, the beams with lower energy will firstly be absorbed by tissues, while those with higher energy can pass through and be received by detectors.

The intensity of the X-ray exiting a scanning material is a function of the material thickness and the attenuation coefficients. When there are metallic objects, whose attenuation value is much higher than human tissues in the scanning body, most of the low-energy beams can not pass. And high-energy beams can pass through the body and be received by detectors, so that the energy of beams received will be much higher than the normal range, and the effect of beam hardening will be extremely severe. The linear relation between the

amount of beams and tissues's density no longer holds, causing a non-linear error at the region with metallic objects.

In the next procedure, the information of X-ray beams that collected by the detectors are used to perform a back-projection to form the reconstructed image. The reconstruction algorithm use the intensity of collected beams to calculate the tissue's density and assumes that the collected X-ray beams' intensity follow a linear relation with tissues's density, which is not the fact due to the non-linear error caused by beam hardening. Since the lightness of reconstructed images should be determined by the densities of human's tissues, when we use the intensity of collected beams to calculate the tissue's density, the non-linear error mentioned above will affect calculation results, which leads to streaking artifacts in the constructed images.

To fix the artifacts directly on the image is not feasible. According to Figure 1.1, the regions that has been influenced by artifacts and the surrounding regions are very blurry and complicated. As a result, it is hard to distinguish metallic objects, artifacts and body tissues on the image. Therefore, instead of fixing the original image directly, projection completion MAR methods tend to restore the corrupted regions on the forward projection of the original image, which is the sinogram domain.

Sinogram is the radon transform of the original image. It can be regarded as a simulation of the tomography process. One single projection can provide information of the object at one direction. If we perform projections to the object from several different directions, we are able to compute the shape of the object. A matrix of projections is called a sinogram, each column corresponding to the information from one direction. Figure 2.1 shows the original image and its corresponding sinogram .

From Figure 2.1, we can see that on the sinogram domain, the metallic objects and its artifacts are shown as several light traces. Comparing with correcting the metal artifacts on the image domain, it is much easier to locate and fix the metal trace on the sinogram domain. The main idea of a projection completion MAR method is to replace those light traces with values calculated from surrounding reliable information. Then, the corrected sinogram is transformed back into an image, which is the final restored image we desire.

A general procedure of projection completion MAR method often consists of two steps. Firstly, identify and locate the pixels that contain wrong information, and secondly make use of different interpolation methods to fix them. Among all the projection completion methods, we will implement three of them in our project, including two 1-D interpolation methods and the normalized MAR method.

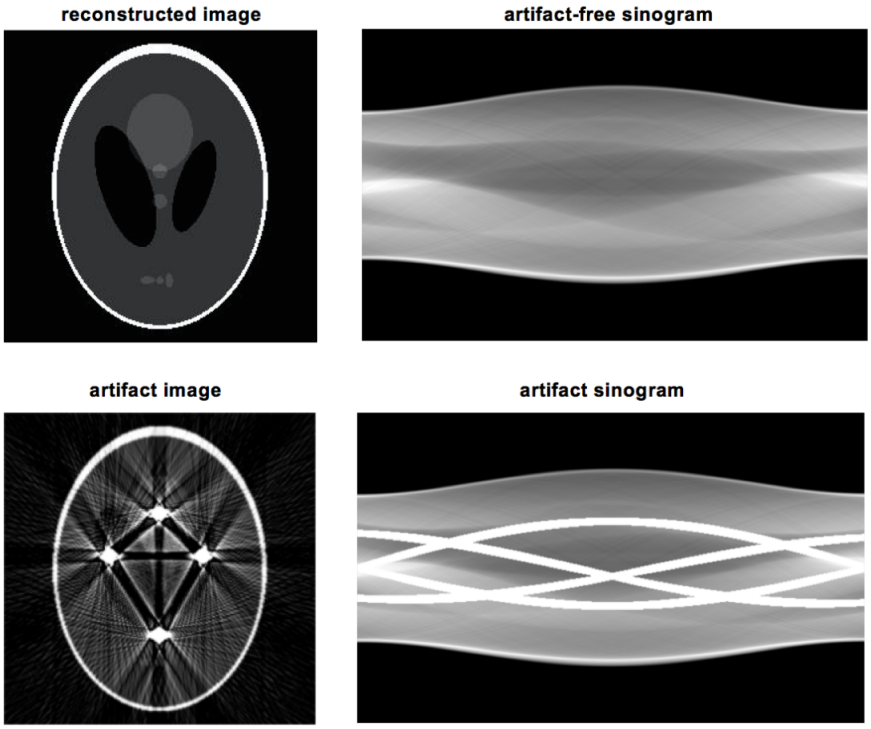

Figure 2.1: Image and corresponding sinogram

2.1.1 One Dimensional Interpolation MAR Method

Two of the one dimensional interpolation methods we will introduce are linear interpolation and cubic interpolation. Linear interpolation technique was firstly made use of to reduce the artifacts in CT images in [20]. As one of the most effective and commonly used algorithms so far, linear interpolation MAR is simple and easy to implement. A general procedure is as follows. Raw data is reconstructed into a 2-D image with artifacts on the background. Then metallic objects are segmented by thresholding. Most of the time, this value is set at the maximum value of the whole reconstructed image, for the reason that metallic material often has a large attenuation value. After that, the segmented metal is forward projected to get a metal trace sinogram. Comparing the metal trace with the original sinogram pixel by pixel, the location of the corrupted raw data can be identified. Two points are chosen a few pixels away from the top and bottom of the metal trace, followed by a linear interpolation. The corrected sinogram is reconstructed again into a restored image. Segmented metal can be inserted back onto the restored image.

Cubic spline interpolation method (CIMAR) almost follows the exactly same procedures except that the interpolation used for replacing the metal pixels is cubic Hermite spline . The cubic spline has a first continuous derivative, therefore is smoother and more continuous comparing with linear interpolation.

Advantages of these two interpolation approaches are that they are easy and fast to implement, and can be combined with other methods. The drawback is also significant. Since both linear interpolation and cubic interpolation only use the information from one direction, it is very sensitive to the location of the interpolation points. With the wrong information, the interpolation results will be inconsistent with the surrounding pattern; thereby generating extra artifacts on the final reconstructed image.

2.1.2 Normalized MAR Algorithm

Another type of the MAR method we compare with is the normalized MAR, or shorten as NMAR. Normalized metal artifact reduction (NMAR) algorithm was recently introduced in [26]. The main idea of NMAR is to take advantage of a prior image, which is obtained by a multi-threshold segmentation of the initial image. The prior image is used to normalize the sinogram projections before interpolation. The sinogram from artifact image is divided by the sinogram from the prior image and, after interpolation, are de-normalized again.

Normalized MAR has only moderate extra computational costs compared with one-dimensional interpolation methods, but could largely improve the image quality. In addition, it can be used as an additional step in other conventional sinogram-completion based MAR methods.

However, NMAR still suffers from a loss of details close to the metal implants as most of the interpolation completion algorithms. Moreover, neither of the LIMAR nor NMAR considers the curvature of the pattern, which will break the continuity of the whole image and affect the visual quality of the restored image.

Generally speaking, most of the classical projection completion methods use only one dimension information. There are some methods proposed recently making use of the optimization techniques, such as choosing optimal pixels from the neighborhood of the corrupted region [34]. The performance of the reconstructed image has been improved, but still lose the continuity of curvature in the background. However, this issue is greatly addressed in the proposed approach which is based on an inpainting algorithm.

2.2 Euler Elastica's Inpainting

In this essay, we propose a method using an image inpainting technique to restore the sinogram. Firstly, a brief review of inpainting and concepts will be described, and then mathematical formulas and discretization will be given.

2.2.1 Inpainting

The term "inpainting" was firstly introduced into image processing area in [8]. The objective of a image inpainting problem is to fill in the missing regions with proper values, so that the whole image looks natural to human eyes.

In some previous inpainting methods, total variation (TV) inpainting [10] aims to minimize the total variation norm in the inpainting domain, while penalizing the changing in other parts outside the inpainting domain. TV inpainting works well for some simple images with sharp edges and homogeneous patches, but it fails to capture the curvature in curved edges. This problem is overcomed by introducing the curvature term into the model, making the whole image more continuous. This kind of inpainting is called curvature driven diffusion (CDD) inpainting [28]. A comparison of the two inpainting methods is shown in Figure 2.2.

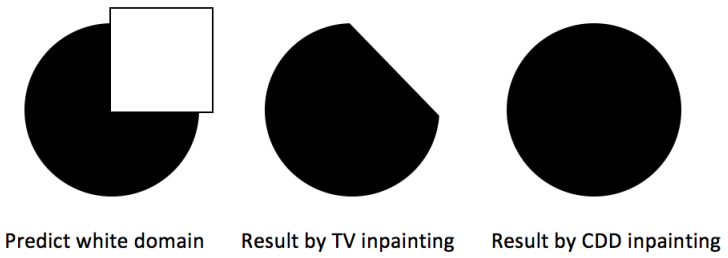

Figure 2.2: Simulated Inpainting result

Then Euler elastica inpainting model is a method that incorporates curvature information into the elastic energy [29]. Firstly, we will show the elastic energy function and the general PDE formulation for this inpainting problem. Then we give the numerical discretization of the Euler elastica inpainting model.

2.2.2 Elastic Energy Function

The elastic energy function describes the energy stored in a flexible rod when there is a changing of its shape in two directions. The model tries to find a curve, Γ, that minimizes the elastic energy function described as follows:

$$E[\Gamma] = \int_{\Gamma} (a + b\kappa(s)^2) ds, \tag{2.1}$$

where $\kappa(s)$ is the scalar curvature of every pieces along the curve with length ds, a, b are two positive constants, and the ratio of b/a determines the elastica of the curve.

Consider Figure 2.3, in an inpainting problem, the missing region to be filled is called "inpainting domain", D. The region that contains correct and reliable pixel values for inpainting is called "extended domain", E, and the whole image is Ω.

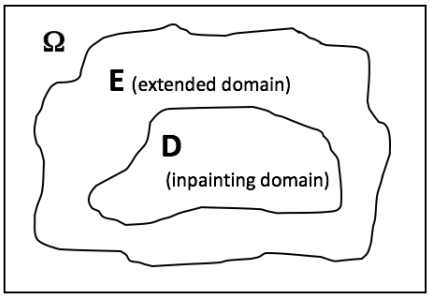

Figure 2.3: Inpainting regions

Let the original image be $u(x)$, $x \in \Omega$, and the curvature of u be defined as the divergence of the normal, which is $\nabla \cdot \frac{\nabla u}{|\nabla u|}$, and $ds = \nabla u dx$. Then the elastica energy function in the inpainting domain becomes

$$E[u] = \int_D (a + b(\nabla \cdot \frac{\nabla u}{|\nabla u|})^2) |\nabla u| dx. \tag{2.2}$$

Then we incorporate the elastica energy function into the inpainting model. Similar to TV inpainting, we add a penalty term to the energy function. But instead of minimizing total variation term, we minimize the elastica energy function. Also in practice, we apply the energy function on $E \cup D$ instead of only the inpainting domain D in order to obtain a smoother result. The PDE formulation of Euler elastica inpainting is

$$J[u] = \int_{E \cup D} (a + b(\nabla \cdot \frac{\nabla u}{|\nabla u|})^2) |\nabla u| dx + \frac{\lambda}{2} \int_E (u - u^0)^2 dx, \tag{2.3}$$

where u^0 is the true image value that we use to compare with the inpainting value at the extended region E; λ is the Lagrangian multiplier which controls the weight on the inpainting term and the penalty term. In [28], the authors show that solving (2.3) is equivalent to computing the steady state solution of the following equation:

$$\frac{\partial u}{\partial t} = |\nabla u| \nabla \cdot \vec{V} - |\nabla u| \lambda_E (u - u_0), \tag{2.4}$$

where

$$\vec{V} = (a + b\kappa^2)\vec{n} - \frac{2b}{|\nabla u|} \frac{\partial(\kappa |\nabla u|)}{\partial \vec{t}} \vec{t}$$

and

$$\lambda_E = \begin{cases} \lambda, & u \in E \\ 0, & u \in D \end{cases}$$

Here $\vec{t}$ is the tangential direction and $\vec{n}$ is the normal direction. In the normal direction, the model propagates the elastic energy directly, while in the tangential direction, it propagates the derivative of the elastic energy.

2.2.3 Numerical Discretization

In this section, we will discretize (2.4) by finite difference method. We use the explicit scheme in [29] which yields the following time stepping method,

$$u_{i,j}^{n+1} = u_{i,j}^n + \Delta t \left[|\nabla u_{i,j}^n| F(u_{i,j}^n) - |\nabla u_{i,j}^n| \lambda_E (u_{i,j}^n - u_{i,j}^0) \right]. \tag{2.5}$$

where (i, j) represents a pixel of the inpainting image, timestep is from $n = 0, 1, \cdots$ with a time step size Δt. Then $u_{i,j}^n$ is the value of u at pixel (i, j) at time step n. Also $F(u_{i,j}^n) = \nabla \cdot \vec{V}_{i,j}^n$, and u^0 is the value in region E that we already know.

There are two unknowns $\nabla u_{i,j}^n$ to be discretized in the right-hand side of (2.5). According to [23], the first one is in front of $F(u_{i,j}^n)$, and is approximated by central differencing

$$|\nabla u_{i,j}| = \frac{1}{2} \sqrt{(u_{i+1,j} - u_{i-1,j})^2 + (u_{i,j+1} - u_{i,j-1})^2} \tag{2.6}$$

and the second one is discretized by upwind scheme

$$|\nabla u_{i,j}| = \sqrt{(upwind\ D_x u_{i,j})^2 + (upwind\ D_y u_{i,j})^2}, \tag{2.7}$$

$$upwind\ D_x u_{i,j} = \begin{cases} u_{i,j} - u_{i-1,j} & \text{if } (u_{i+1,j} - u_{i-1,j})(u_{i,j} - u^0_{i,j}) > 0, \\ u_{i+1,j} - u_{i,j} & \text{if } (u_{i+1,j} - u_{i-1,j})(u_{i,j} - u^0_{i,j}) < 0. \end{cases}$$

$$upwind\ D_y u_{i,j} = \begin{cases} u_{i,j} - u_{i,j-1} & \text{if } (u_{i,j+1} - u_{i,j-1})(u_{i,j} - u^0_{i,j}) > 0, \\ u_{i,j} - u_{i,j+1} & \text{if } (u_{i,j+1} - u_{i,j-1})(u_{i,j} - u^0_{i,j}) < 0. \end{cases}$$

Now is the discretization of $F(u^n_{(i,j)}) = \nabla \cdot \vec{V}^n_{(i,j)}$. It is carried out in two dimensions based on half-point central differencing.

$$\begin{aligned} F(u^n_{i,j}) &= \nabla \vec{V}_{i,j} = D_x V^1_{i,j} + D_y V^2_{i,j} \\ &= \left(V^1_{i+\frac{1}{2},j} - V^1_{i-\frac{1}{2},j}\right) + \left(V^2_{i,j+\frac{1}{2}} - V^2_{i,j-\frac{1}{2}}\right) \end{aligned} \tag{2.8}$$

where V^1 and V^2 represent the x and y components of $\vec{V}$. For the curvature term κ, we use the "min-mod" method to calculate the values at the half-point:

$$\kappa^n_{i+\frac{1}{2},j} = minmod(\kappa^n_{i+1,j}, \kappa^n_{i,j}), \quad minmod(\alpha, \beta) = \frac{sign(\alpha) + sign(\beta)}{2} min(|\alpha|, |\beta|) \tag{2.9}$$

D_x at half-point $(i + \frac{1}{2}, j)$ is approximated by central differencing of $(i + 1, j)$ and (i, j)

$$D_x u_{i+\frac{1}{2},j} = u_{i+1,j} - u_{i,j}, \tag{2.10}$$

$$D_x(\kappa|\nabla u|)_{i+\frac{1}{2},j} = \kappa_{i+1,j}|\nabla u|_{i+1,j} - \kappa_{i,j}|\nabla u|_{i,j} \tag{2.11}$$

These are all the discretized terms in (2.5). To the consideration of stability, we use the implicit scheme provided in [9]. They have shown that the implicit scheme is unconditionally stable, which means that it converges with any timestep. This allows an implementation with a relatively larger timestep, which speeds up the algorithm. The approach used in the implicit scheme is based on the "convexity splitting" technique [15]. The main idea is to divide the energy function into a concave part and a convex term, and solve the concave component explicitly and the convex component implicitly. During the division, there is a C_1 parameter, which is a constant added to ensure the convexity. Then all the other discretization methods used for other terms are the same with those in the explicit method.

The final discretization formula for Euler elastica inpainting model is

$$\begin{aligned} & u^{n+1} - \Delta t \frac{1}{\lambda} \nabla \left((a + C_1 + b\kappa^2)\vec{n}^{n+1} + u^{n+1} \right) \\ = \quad & u^n + \Delta t \left[\frac{1}{\lambda} \nabla \left(-C_1 \vec{n}^n - \frac{2b}{|\nabla u^n|} \frac{\partial(\kappa|\nabla u^n|)}{\partial \vec{t}^n} \vec{t}^n \right) + u^0 \right], \end{aligned} \tag{2.12}$$

with all the parameters to be assigned: λ is the the Lagrangian multiplier; C_1 is a term in implicit scheme to ensure the convexity; Δt is the timestep; a and b are the elastica parameters that determine the stiffness of the curve.

(2.12) can be rewrite into a linear system

$$Au^{n+1} = f,$$

where u^{n+1} is the prediction value at the next timestep. A is the coefficient matrix of u^{n+1}, and f is the right-hand sides with all the terms based on u^n. Then u^{n+1} of next timestep can be calculated by solving the above linear system.

Chapter 3

Methodolody

The proposed model is based on the projection correction MAR approach. The main idea is that instead of using a one-dimensional interpolation, we use the Euler elastica inpainting to restore the corrupted region.

In order to do the inpainting, we need to know the location of the corrupted region on the sinogram domain. However, due to the reason that it does not work well to capture the region boundary directly on the sinogram, we decide to segment the metal part on the image domain, followed by a forward projection of the segmented metal. Comparing the projected sinogram with the original one, we can locate the metal trace region. After implementing the inpainting algorithm, we reconstruct the sinogram back into a 2-D image, which is the final restored image that we desire. The whole procedure can be divided into the following stages.

1. **Metal Segmentation**
   Apply K-means clustering method to segment the metal part on the initial artifact image.

2. **Forward Projection and Metal Trace Identification**
   Forward project the segmented metal to get a metal trace; compare the metal trace with original sinogram to locate the inpainting domain.

3. **Contrast Improvement and Sinogram Inpainting**
   Use Gamma Correction to increase the contrast of the sinogram; then apply the Euler Elastica inpainting algorithm onto the inpainting domain of the sinogram.

4. **Reconstruction and Metal Insertion**
   Obtain the reconstructed image by FBP algorithm and insert the segmented metal back into the corrected image.

5. **Iteration Correction**
   Apply steps 1 - 5 iteratively until the image has been repaired to an acceptable degree.

A flowchart of the whole procedure is shown in Figure 3.1.

3.1 Image Segmentation

We will apply K-means clustering on the artifact image to segment the metal objects. Actually, what we want is to locate the region that contains the wrong information affected by the metallic objects on the sinogram. Intuitively, the light trace on the sinogram should be the metal trace. However, sometimes the boundary of the trace is very blurry, or the background is very complicated, then it fails to identify the metal trace and using thresholding alone is not able to segment it out.

Then we consider to segment the metal on the image domain first, and then project the segmented metal to obtain a metal trace sinogram. With the same size as the original sinogram, we can compare these two and locate the inpainting domain on the original sinogram.

In order to segment the metal object, the most commonly used method is applying thresholding on the reconstructed image domain, since the pixel intensity values of the metal region are typically much larger than the rest of the image. Thresholding works well on images with small amount artifacts. However, when the image is badly damaged by a large amount of artifacts, thresholding is not able to identify the metal piece. In this thesis, we use K-means clustering instead of thresholding to segment the metal part on the reconstructed image.

K-means clustering is an algorithm to classify the objects based on their features into K groups. The aim of the clustering is to group data in a way that similar objects are in one cluster and objects of different clusters are dissimilar. The grouping is done by minimizing the distances between data and the corresponding cluster central point. The objective function is defined as below:

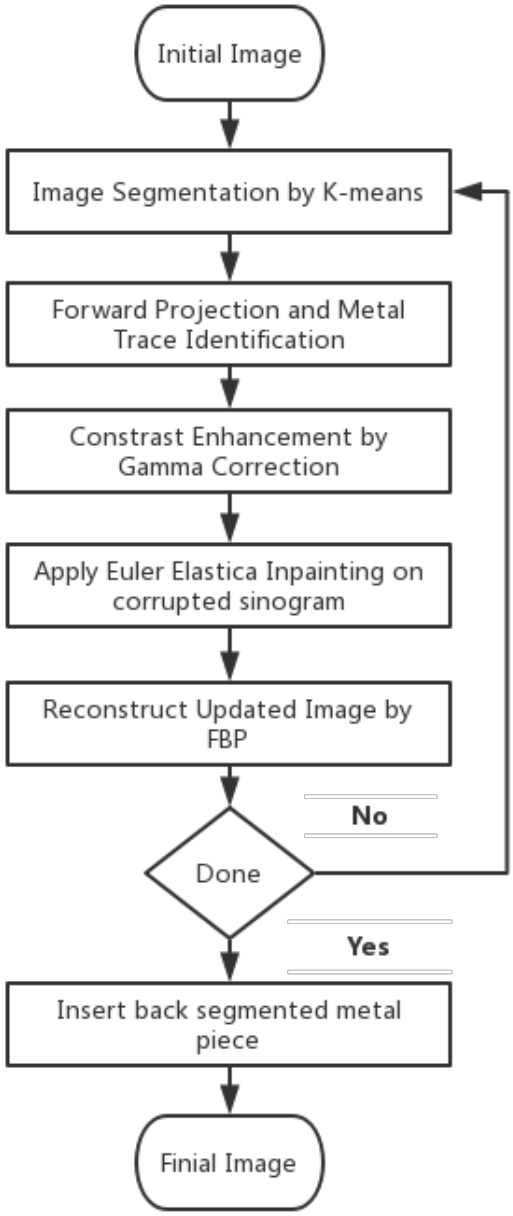

Figure 3.1: Flowchart of the whole methodolody

$$F = \sum_{i=1}^{K} \sum_{x_j \in S_i} (x_j - c_i)^T (x_j - c_i), \tag{3.1}$$

where $S_i, i = 1, 2, \cdots, K$ are K clusters, and c_i is the central point of all the points $x_j \in S_i$.

In general, K can be selected manually at first. Since we just want to segment the metal, then setting $k = 2$ would be the first choice. Sometimes, larger K will yield better result when the background is complicated. After the image has been segmented into K pieces, we manually choose the one from the K pieces to check which one is the metal piece. Normally, the Kth piece is the one containing the brightest patterns, so is the metal piece. However, it still requires an extra work to check for that. Figure 3.2 shows some experimental results applying K-means clustering on medical CT images.

3.2 Forward Projection and Metal Trace Identification

In the previous section, we used K-means clustering to segment the metallic part on the reconstructed image. We will make use of the metal information to locate the inpainting domain on the sinogram. The segmented metal image is needed as an "input" image for the forward projection, and projected under the same parametric conditions to get a metal sinogram as left Figure 3.3.

Normally, the inpainting domain is set slightly wider than the exact metal trace location. Since the segmentation may not be accurate, a wider region will ensure that the segmented region will include the metal trace. However, setting the inpainting domain too large will also damage the correct information. Therefore, a proper thresholding value σ for the metal trace identification is important. Based on our experience, a σ is often in the range of a quarter to a half of the maximum of the metal sinogram. Then a mask of the metal trace, M, can be obtained by

$$M(i,j) = \begin{cases} 1, & \text{if} \quad S(i,j) \geq \sigma \\ 0, & \text{if} \quad S(i,j) < \sigma \end{cases} \tag{3.2}$$

Since the metal sinogram and the original sinogram have the same size, the non-zero pixels on the metal sinogram correspond to the metal trace on the original sinogram. The inpainting domain is then located, as the red region in the right of Figure 3.3.

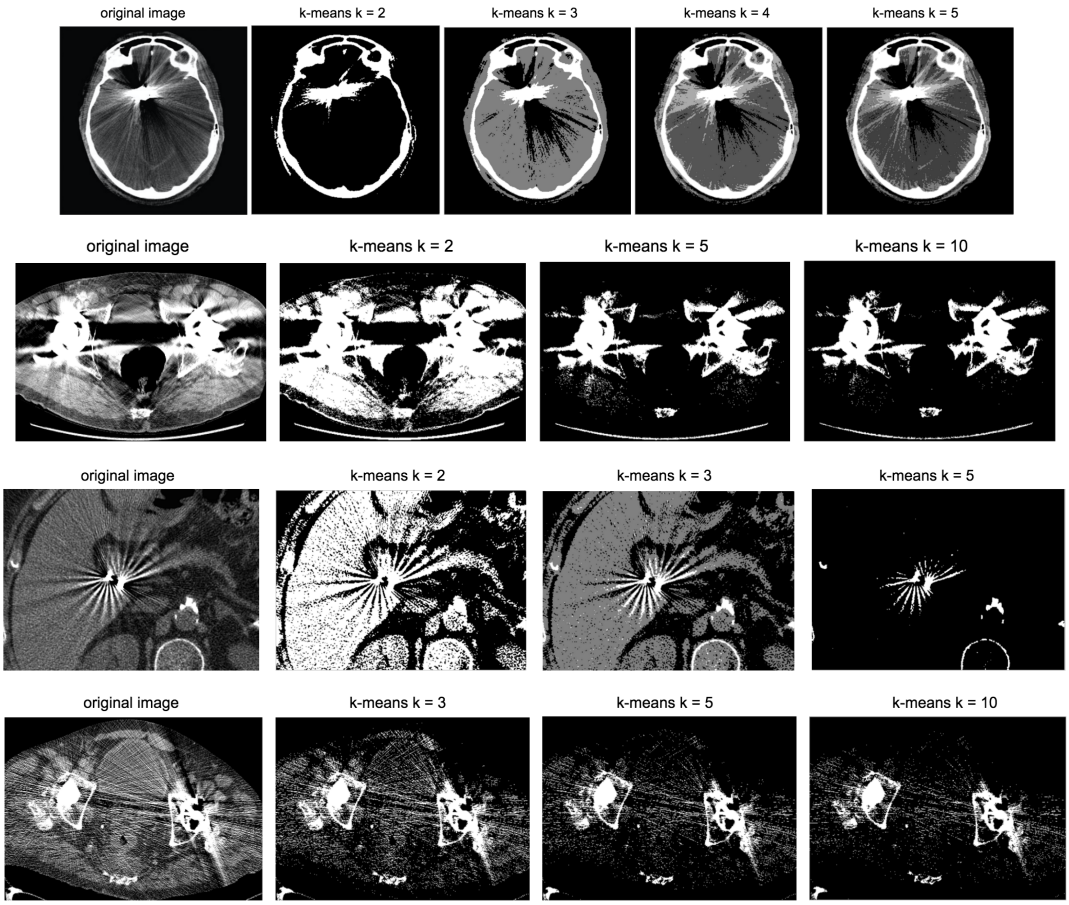

Figure 3.2: K-means Segmentation

3.3 Contrast Enhancement and Sinogram Inpainting

Sinogram images are normally gray-scale images, often in low contrast with shades and lights in the background. We apply a contrast enhancement technique to improve the quality of the sinogram before inpainting. Increasing the contrast helps calculating gradient values so that capturing the curvature more accurately. Accordingly, when inpainting completed, an inverse correction will be applied with the same parameter to recover the image into the same value range as before.

The method we use in this project is called Gamma Correction, or often simply gamma. It is a nonlinear algorithm used to encode and decode luminance in images. Gamma

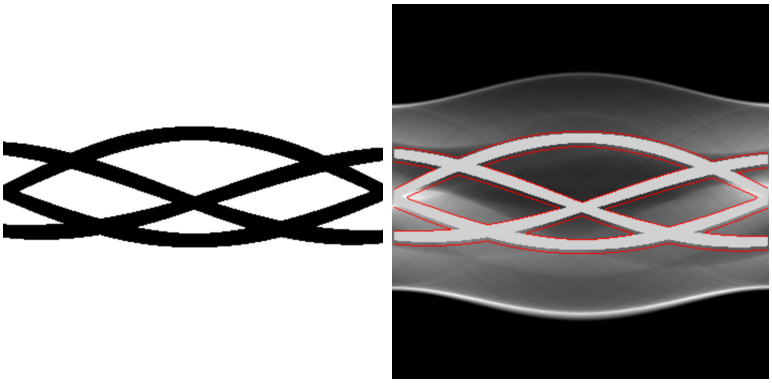

Figure 3.3: Metal mask and the identified metal trace inpainting domain on the sinogram

Correction is defined by the following mathematical expression:

$$V_{out} = A \cdot V_{in}^{\gamma} \tag{3.3}$$

where A is a constant, V_{in} is the image to be processed, and V_{out} is after-processing image. In the common case, we set $A = 1$ and γ is in the range of $[0, 1]$. When $\gamma < 1$, it is called an encoding gamma, and the contrast of the image will decrease; conversely, $\gamma > 1$ is called a decoding gamma and will increase the contrast of the image. A comparison of the effect of applying Gamma Correction is shown in Figure 3.4.

Now we can implement the Euler elastica inpainting algorithm proposed in the previous section onto the corrected sinogram. In practice, there are several issues to be discussed before implementation: initial condition, boundary condition and parameter choices.

3.3.1 Initial Condition and Boundary Condition

All PDE-based algorithms require setting the initial condition and the boundary condition. For a 2-D image, an initial condition is the default value of its inpainting domain, and a boundary condition is the intensity values of the pixels at four edges of the image.

For the initial condition, we set the pixel values to be zero on the inpainting domain. In some of the inpainting literature, it is also common to use random noise values as an initial condition [29]. Experiments have shown that different initial conditions do not show much influence to the final effect as well as to the speed. Thus we just use zero as an initial value for all the implementations in this project.

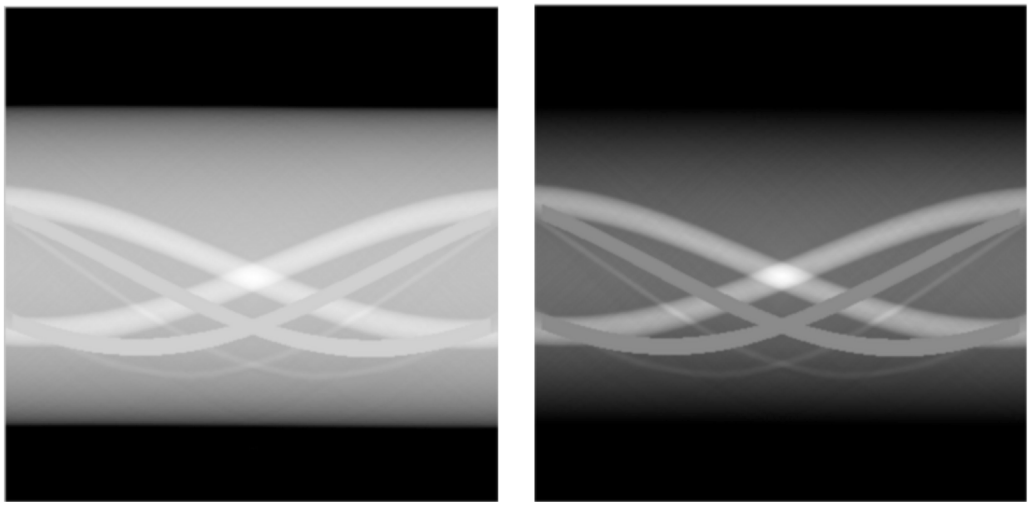

Figure 3.4: Increase the contrast using Gamma Correction

Boundary condition is relatively more complicated. In this thesis, inpainting domain will not spread to the top and bottom edges, therefore, we only need to consider the values at the left and right boundaries. The method we use is to apply linear interpolation to the most left and right five columns of the image. From Figure 3.5, we can see that the linear interpolation produces reasonable results at the boundaries.

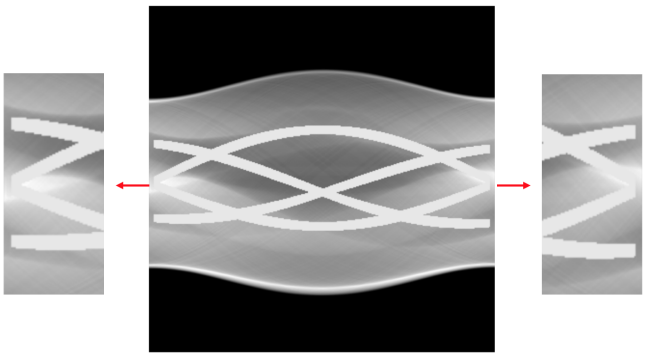

Figure 3.5: Boundary conditions given by linear interpolation

3.4 Reconstruction

The corrected sinogram is reconstructed by inverse Radon transformation after applying the inverse Gamma Correction on the inpainted sinogram. Normally, the original metal location will turn into a gray or black hole on the corrected image. Therefore, the segmented metal is inserted back into the image to make the final image more complete and consistent.

3.5 Iterative Correction

When an image is dramatically damaged by the artifacts, an iteration correction is required. The iteration correction is to repeat the previous step 1 to step 4 more times until a stopping criteria has been reached. After each iteration, the image quality gets better, which will benefit the whole procedures for next inpainting; thereby we can expect a better restoration for next iteration.

3.5.1 Stopping Criteria

A stopping criterion is used for both inpainting step and iteration step. It is measured by the changes between two steps. The quantitative parameters we use include a root-mean-square error ($RMSE$) and a peak signal-to-noise ratio ($PSNR$). They both measure the similarity of two items, but in different scales.

$RMSE$ is one of the most commonly used measures to calculate the error between two items. It represents the sample standard deviation of the differences between predicted values and observed values. The formula is as below:

$$RMSE = \sqrt{\frac{\sum_{j\in\Omega}(\hat{f}_j - f_j)^2}{N_\Omega}} \tag{3.4}$$

where N_Ω denotes the number of pixels in the chosen region Ω. $\hat{f}_j$ is the inpainted value of a pixel and f_j is its original value of that pixel.

Another metric is $PSNR$, which is commonly used to measure the quality of reconstruction loss of image or signal in engineering, based on the calculation of $RMSE$. Generally speaking, $PSNR$ is preferred over $RMSE$ because the measure increases when the error

decreases, meaning that a better inpainting result has a higher $PSNR$ value.

$$PSNR = 20 \times log_{10} \left(\frac{1}{RMSE(\hat{f}_j, f_j)} \right). \tag{3.5}$$

Figure 3.6 shows the absolute value of $PSNR$, $|PSNR^t|$, and the changing difference of $PSNR$, $|PSNR^{t+1} - PSNR^t|$, during a inpainting process. In Figure 3.6, we can see that there is a rapid decreasing at the beginning and then gradually decreases to convergence. The changing difference has declined to zero over time, ensuring the convergence again. A common iteration number for convergence is around 2000 timesteps and the stopping criterion of the difference of $PSNR$ is smaller than 10^{-6}.

As to the stopping criterion for the whole model iteration, the stopping criteria of the difference of $PSNR$ is set at 10^{-2}, which means that the difference between two iterations is less than 0.01.

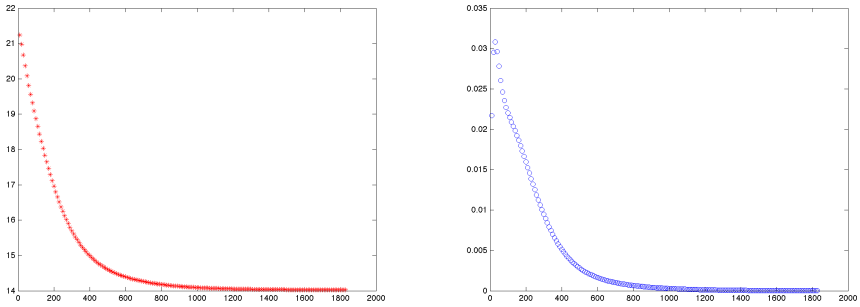

Figure 3.6: PSNR and changing of PSNR over time

Chapter 4

Experiments

In this chapter, we will apply our model to four types of image data. First, we will briefly introduce the testing images. An experiment of the parameters used in the model is carried out. Then, we test the model on the sinogram domain. Next on the image domain, we compare the results with standard methods, LIMAR, CIMAR and NMAR. Finally, iteration correction scheme is tested on type C and D images, and some difficult cases are considered.

4.1 Image Data Sets

Four image data sets are used for the numerical experiments (Figure 4.1) : Shepp-Logan phantom (A), jaw phantom (B), artifact-free CT phantom images (C) and clinical CT images with streaking artifacts (D).

A. The first type is the Shepp-Logan phantom, which is a gray-scale intensity image that consists of one large ellipse (representing the brain) and several smaller ellipses (representing features in the brain). We use this Shepp-Logan phantom to test the inpainting algorithm on accuracy, efficiency and consistency. The Shepp-Logan phantom has its advantages such as small size, simple and easy to edit.

B. The second type is called the jaw phantom. It simulates tooth CT image with dental fillings. This origin of the jaw phantom is in [25], where they simulated the teeth with different size of circles. We use this original jaw phantom image to simulate artificial CT images suffering from beam hardening and streaking artifacts in a similar way as in real

CT scanning. The aim of using jaw phantom is to compare the inpainting performance with LIMAR and NMAR.

C. The third type is artifact-free CT images obtained from the open sources on the Internet [1]. Comparing with manually created phantom images, CT images have more complicated background and patterns. We use this image data set to implement LIMAR, CIMAR and iteration tests. Quantitative evaluation will also be carried out between the corrected images and the original artifact-free CT images.

D. The last type is the clinical CT images that are characterized by serious metal artifacts. These will be the worst case to be considered and we do not know the original clean images. Iteration tests will be implemented with this type of images.

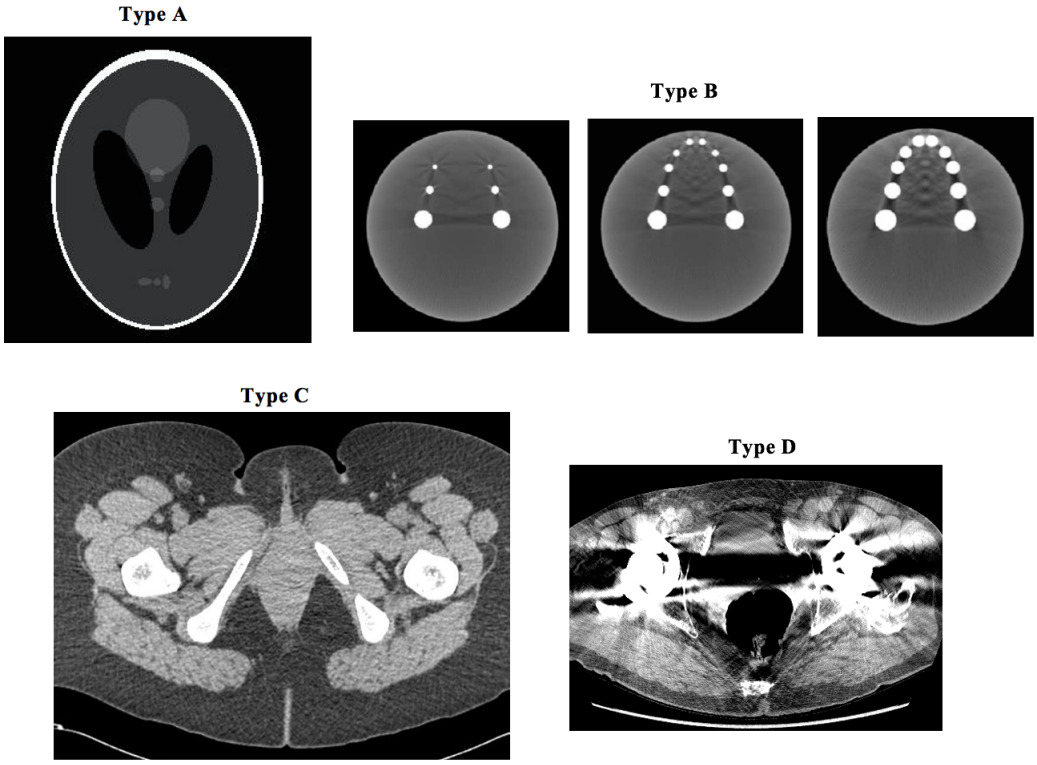

Figure 4.1: Four types of image data used for testing

4.2 Parameter Selection

Before implementation, we carry out simulated tests for all the five parameters used in implicit scheme, and discuss how poor parameter choices can affect the inpainting results.

There are totally four parameters to be determined. They are timestep dt, elastica parameters a, b, convergence constant C_1, and Lagrangian multiplier λ.

We did all the tests on a 100×100 gray image cut from the jaw phantom sinogram. Basic information of the test image is shown in Figure 4.2.

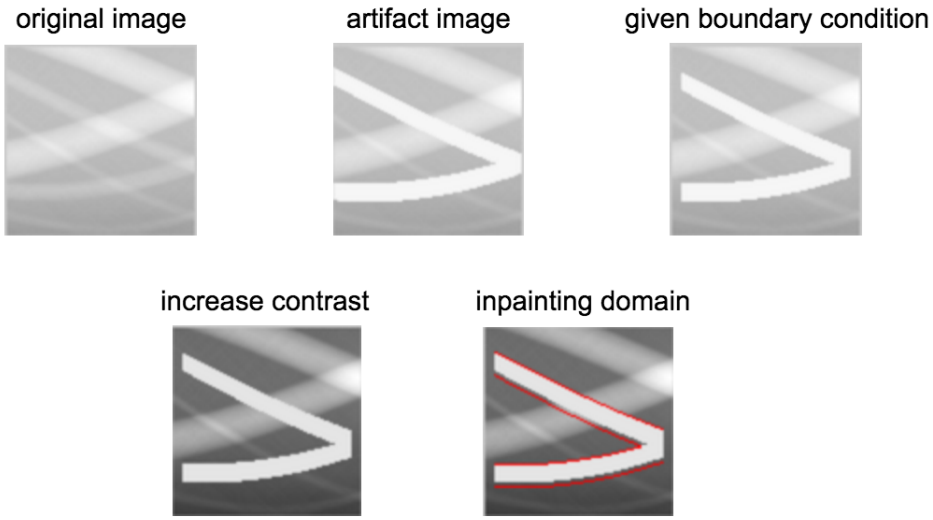

Figure 4.2: Test image

4.2.1 Timestep

Basically, an implicit scheme will always converge with any timestep. However, our experience is that a large timestep does not necessarily guarantee fast convergence. The most important thing in the implicit algorithm is the length of the timestep, rather than the number. From Figure 4.3, we can see that the results with a large Δt do not always outperform the ones with a small Δt.

With $\Delta t = 1$, the algorithm will converge after $step = 300$. The $PSNR\_change$ fluctuates around 2×10^{-6}. When $\Delta t = 10$, the $PSNR\_change$ keeps around 2×10^{-5} after $step = 200$. It shows that both choice will converge. When Δt is larger, convergence

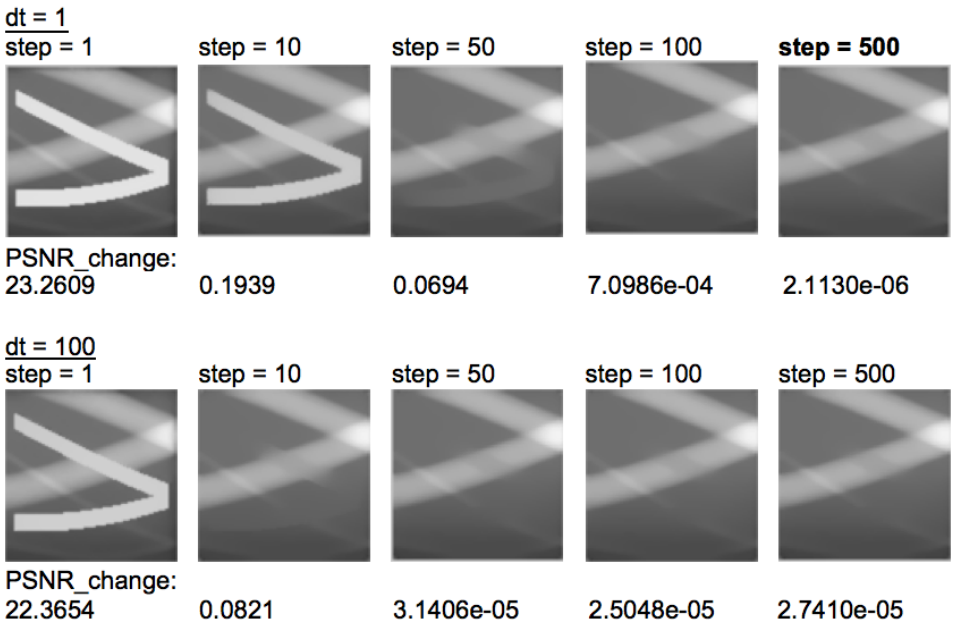

Figure 4.3: Test on timestep

happens faster, but the result is not very accurate. It is important to balance time and accuracy. However, we do have more interest in the quality of the result than time, so a smaller Δt is desired. In our implementation, we usually take $\Delta t = 1$.

4.2.2 Elastica parameter

Elastica parameter refers to the ratio between a and b. In general, larger b/a ratio will lead to a stiffer inpainting result, because the inpainting function has put more weights to the curvature term in the elastica model. As a result, the inpainted edges become sharper.

On the other side, increasing the b/a ratio could result in a side-effect. In order to maintain the curvature, the inpainting will spread outside the inpainting domain, causing extra noises. However, this effect can be offset by increasing the Lagrangian multiplier λ, which will penalize any changes to the image outside the inpainting domain.

When $b = 0$, which means there is no attention to the curvature while inpainting, the $PSNR\_change$ parameter keep decreasing. When the b/a ratio between 1-2, the algorithm converges after about 200 steps. When b/a ratio is more than $5(b > 25)$, the image begins to appear noisy.

In our experimentation with the medical image sinogram, the inpainting domain is mainly refer to the metal trace. The trace is relatively straight with low-curvature. Therefore, adjusting the b/a ratio does not affect much to the results. Therefore in practice, we just set the ratio of b/a to be 1.

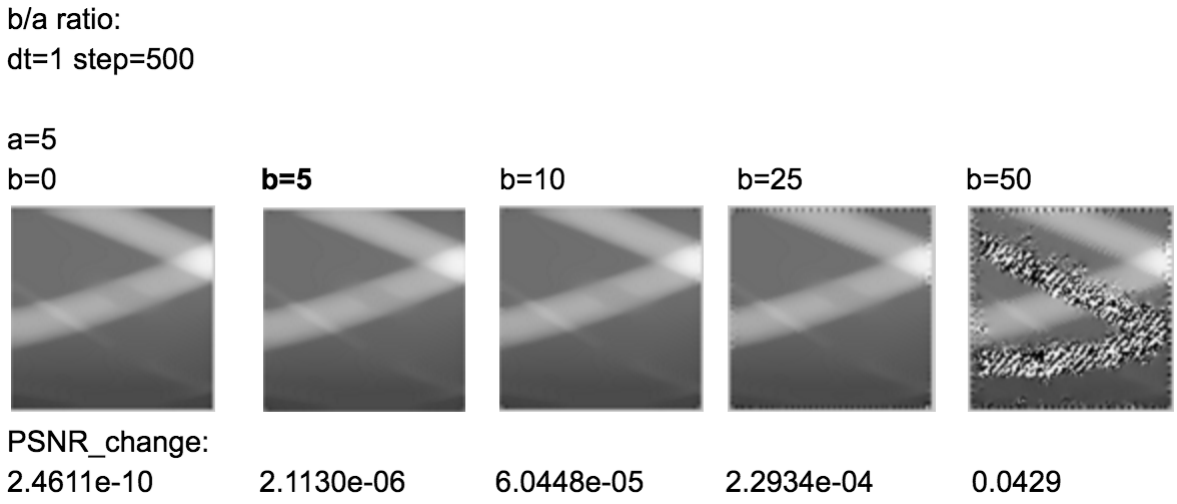

Figure 4.4: Test on b/a ratio

4.2.3 Convergence constant

The USTM constant is the term C_1 in the implicit scheme [9]. Theoretically speaking, C_1 is a convexity term that controls the convergence of the algorithm. However in practice, C_1 does not show any effect on the convergence no matter how small C_1 is. Therefore, we just set C_1 as 1 in our implementation.

4.2.4 Lagrangian multiplier

The Lagrangian multiplier, λ, is a penalty term that adds weight on the boundary term in (2.3). As introduced before, it has a inverse effect against the Elastica parameter. Increasing λ will emphasize the weight on the "changing" component during inpainting. As shown in Figure 4.5, when it is too small, blur and artifacts will appear surrounding the inpainting domain. Large λ, however, significantly slows down the speed of algorithm for convergence.

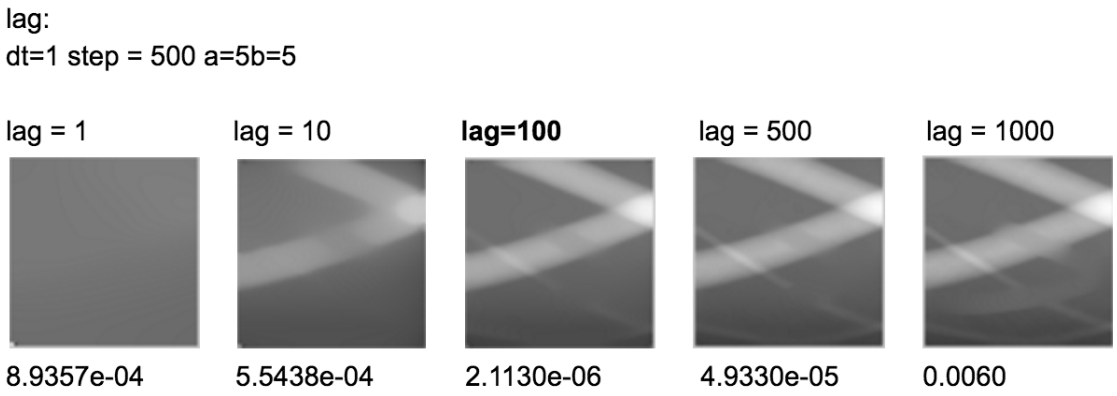

Figure 4.5: Test on lagrangian multiplier

4.2.5 Zero gradient

When discretizing the PDE formulas, some of the terms need to be divided by the magnitude of the gradient. Therefore, there are some cases that when the gradient is zero, the dominator is zero. Then a small positive constant, ϵ, is added to avoid this situation. In practice, this term could dramatically influence the result as shown in Figure 4.6.

Setting ϵ too small will affect the convergence of the algorithm, and when ϵ is too large, the algorithm will not even converge. In practice, we have found that it works well using an ϵ around 10^{-2} for simple test images and 10^{-3} for complicated clinical CT images.

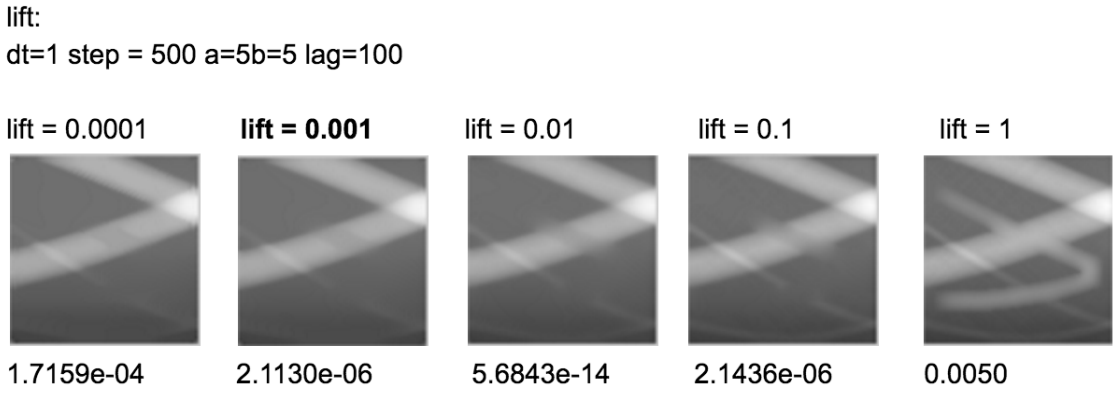

Figure 4.6: Test on lifting factor

4.3 Inpainting Results on Sinogram Domain

The sinogram shown in Figure 4.7 is obtained from the projection of a jaw phantom. The medium gray trace between the wider and thinner ones is the metal trace region.

From Figure 4.7, it can be seen that inpainting works well in the central part of the image, preserving clear edges as well as the curvature of the light trace in the background. However, at the corner of the image where three traces intersect with each other, the continuity is broken, and the Euler elastica's inpainting will lose its curvature preserving feature; see red arrows in Figure 4.7

Figure 4.8 is cut from a clinical head CT phantom. The comparison clearly shows the "curvature preserving" principle of Euler elastica inpainting. The inpainting domain is the white trace bounded by the red lines in the first artifact image. After inpainting, both linear and cubic interpolation create inconsistent edges at the boundary of the inpainting domain. The Euler elastica inpainting gives a smoother output as expected, which connects the background pattern together.

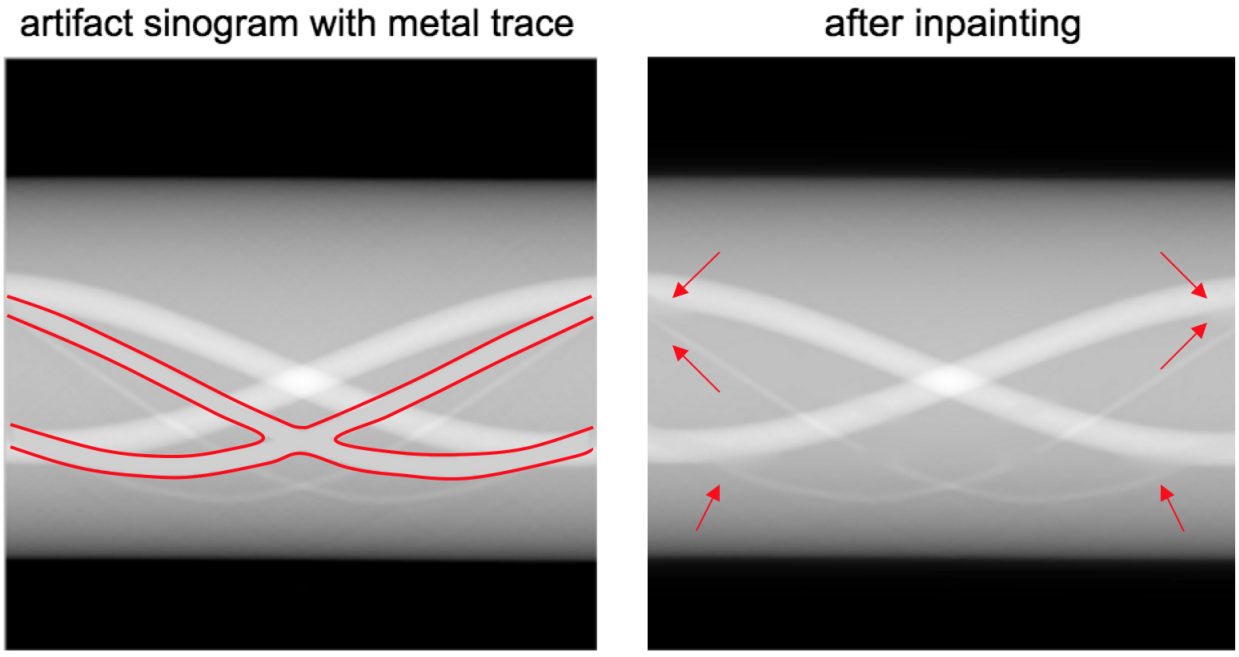

Figure 4.7: Correction of the artifact sinogram

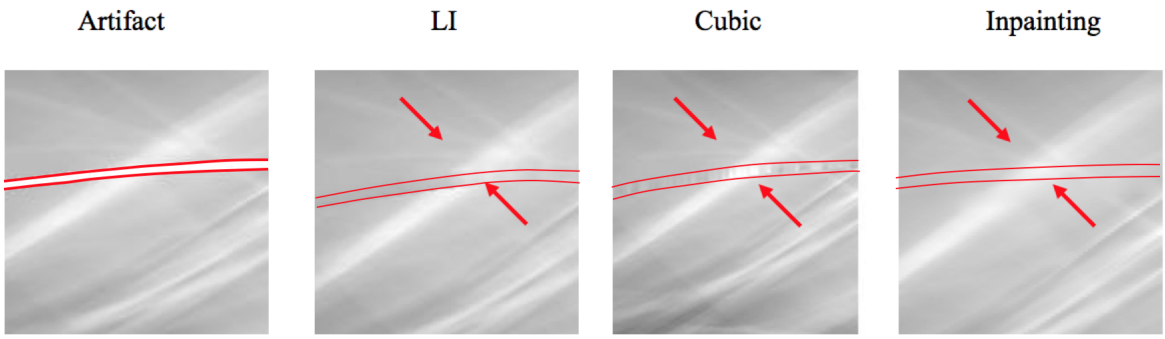

Figure 4.8: Correction of the artifact sinogram

4.4 Conparison with LIMAR and NMAR

Figure 4.9 shows the comparative performance of different MAR algorithms using the simulated jaw phantom image data (Type B). In order for comparison, we simulated similar three set of artifact jaw images as in paper [25], and compare our inpainting restoration results with the LIMAR and NMAR results shown in the paper.

The first column are three original artifact-free reference images. The spheroids represent teeth of different sizes.The second column are artifact images, with two simulated dental fillings. We implemented our inpainting model on these three types of dataset and compare the inpainting results with the LIMAR and NMAR results in [25]. All the methods have noticeably reduced streaking artifacts surrounding the tooth area. LIMAR results in the paper performs worst and introduces a large amount of new artifacts, resulting in a result even worse than the uncorrected image.

Both NMAR in the paper and Inpainting method reduce the streaking artifacts while introducing moderate new artifacts. Actually, there are only a few differences between NMAR and Inpainting from the outputs of these three set of comparisons. In the first set of comparison, where the metal pieces are relatively small, NMAR performs slightly better with fewer new artifacts than Inpainting. But with larger metal implants such as in the third set of comparison, Inpainting shows better details than NMAR. We remark that the simulated artifacts in the uncorrected image here is more severe than the ones in the original paper. This example illustrates that Euler's elastica inpainting method is superior to LIMAR and as good as, or slightly better than NMAR.

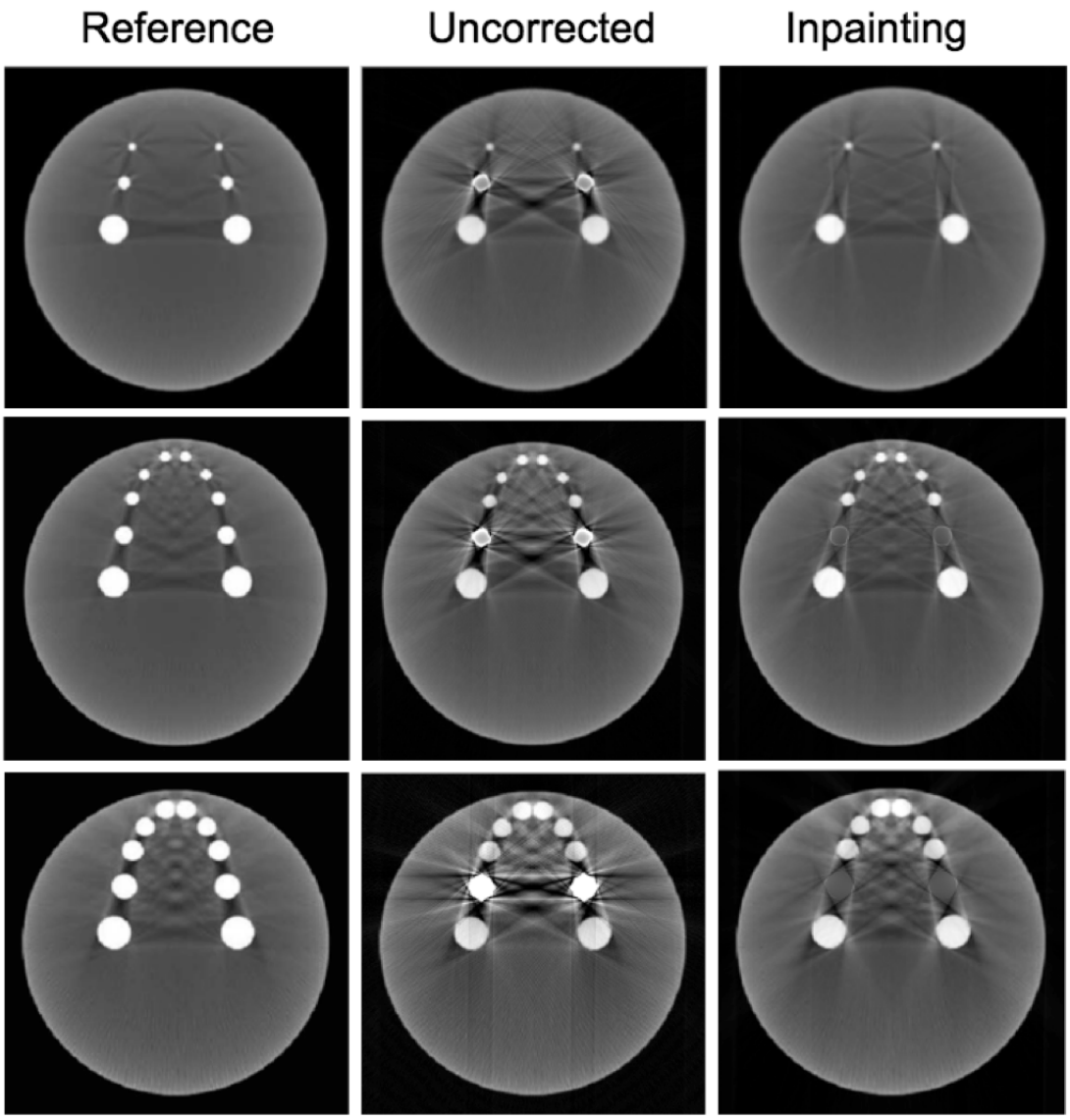

Figure 4.9: Euler's elastica inpainting results with jaw phantom image

4.5 Comparison Experiments with LIMAR and CIMAR

In this section, we apply LIMAR, CIMAR and Inpainting methods on three typical CT phantom image data (Type C) and conduct a quantitative comparison.

The first two columns are reference and manually-created artifact images respectively. Severe streak artifacts can be observed spreading from the metallic object location. Correction results on both image and sinogram are displayed in Figure 4.10 and Figure 4.11. In Figure 4.10, the first column is the original artifact-free images. In the second column, we can see the artifacts origin from the metallic objects. And the comparison in the last three columns demonstrates that inpainting results outperform those by LIMAR and CIMAR with better preservation of structure and details, especially in regions close to metallic objects. Also in Figure 4.11, the restored sinogram after LIMAR and CIMAR contain significant inconsistency in the inpainting domain, which result in the extra artifacts in the reconstructed images. In conclusion, experiment results from both image and sinogram domain show that the proposed inpainting method provide a superior correction than LIMAR and CIMAR.

A quantitative comparison is then carried out between the reconstructed images and the original artifact-free phantom images. The measurements are RMSE and PSNR parameters that we introduced before. From Table 4.1, it shows that for all the three types of CT phantoms, inpainting algorithm has the highest PSNR, which means the best restoration score. CIMAR performs better than LIMAR with head and tooth phantoms, but less than LIMAR in the hip phantom which suffers the largest metallic region among these three.

		LIMAR	CIMAR	Inpainting
Hip	PSNR	21.1491	17.4287	27.1354
	RSME	0.0876	0.1345	0.0440
Head	PSNR	22.4392	34.5161	35.5229
	RSME	0.0531	0.0597	0.0132
Tooth	PSNR	30.0728	31.4894	36.9748
	RSME	0.0314	0.0266	0.0142

Table 4.1: Quantitative measurement comparison with LIMAR and CIMAR

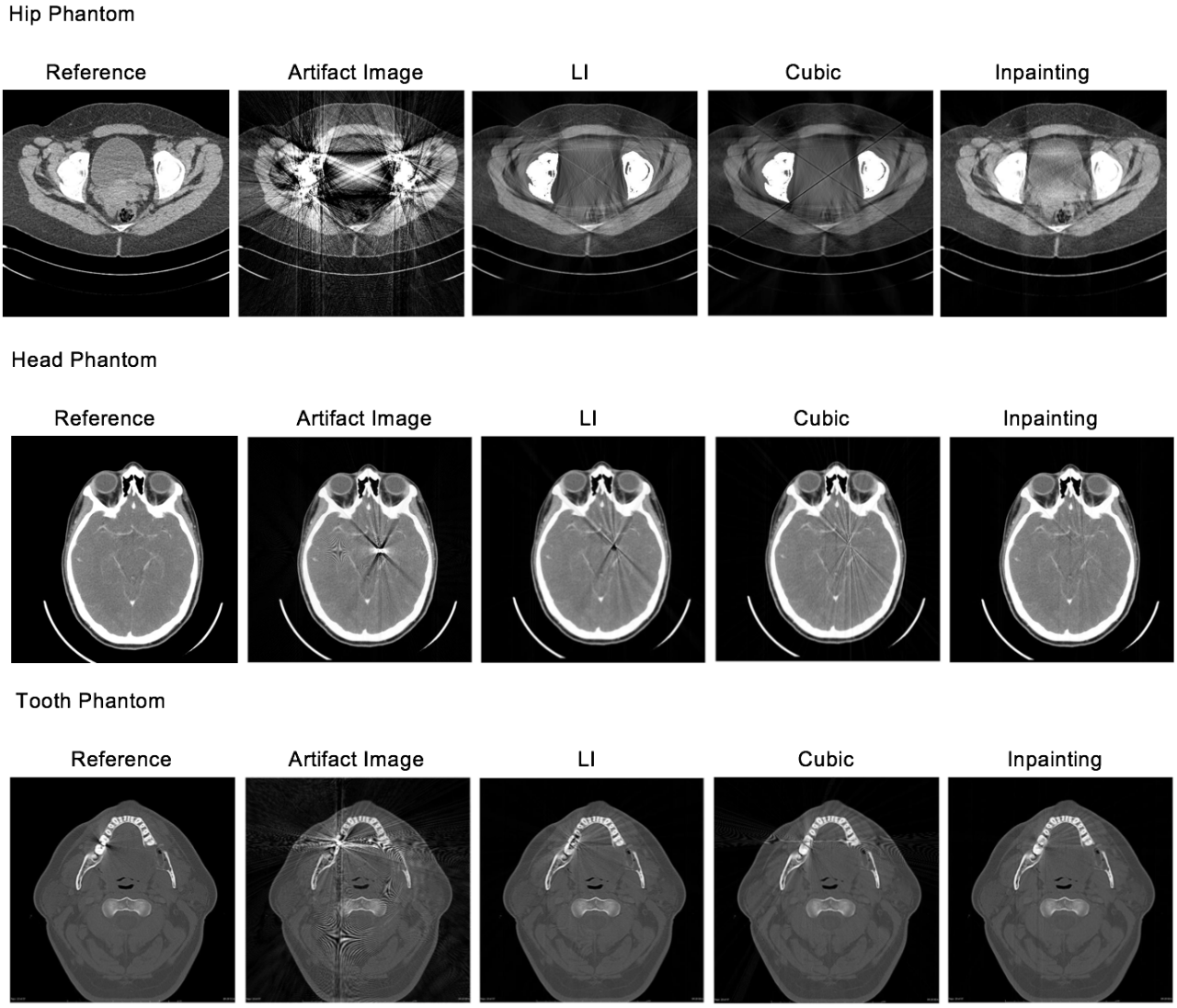

Figure 4.10: Clinical CT phantom image comparison with LIMAR and CIMAR

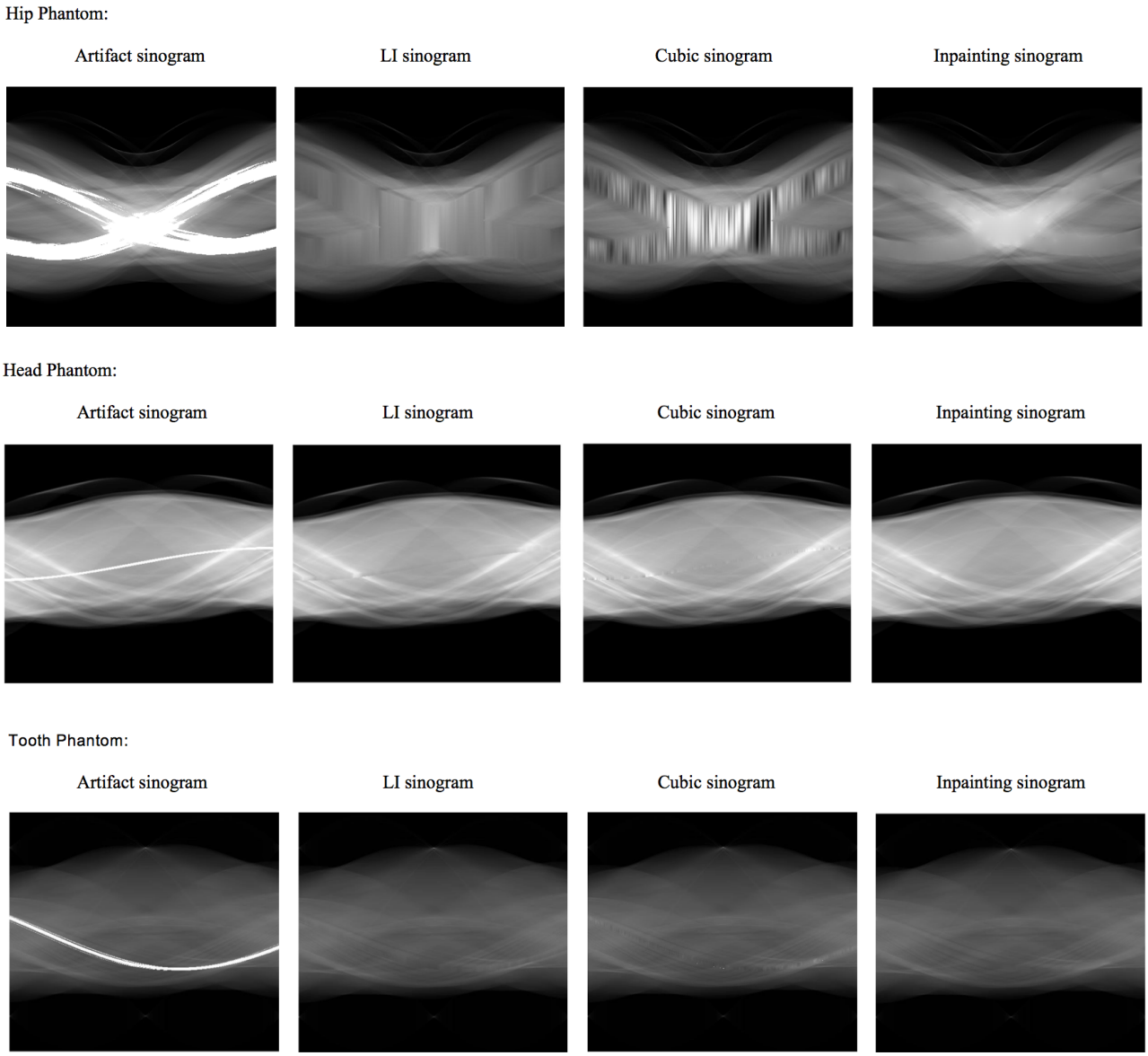

Figure 4.11: Clinical CT phantom sinogram comparison with LIMAR and CIMAR

4.6 Iteration Experiments

In this section, we apply the iteration correction on both CT phantom images (type C) and real clinical CT images (type D). Images used in this section are often suffered from more severe artifacts and we expect iterative correction can yield better results after each iteration. A stopping criterion, which is referred as PSNR in our thesis, is set approximately 10^{-2}.

It can be shown from both imaging results (Figure 4.12) and numerical results (Table 4.2) that each iteration has an improvement for correction. In Figure 4.12, we can tell that the artifacts in three images have been decreased each time than last time. And in Table 4.2, the PSNR values for three images are increasing with the iteration times. The improvement from the original artifact image to the first iteration is the largest. The subsequent improvement decreases until convergence.

Figure 4.13 are real clinical CT images, including a head, a total hip arthroplasty and a hip prostheses. It illustrates the results given by the Euler elastica inpainting applying on the real CT images with different times of iteration. It has improved the visual quality of the artifact images in a significant degree.

Finally, we carry out the experiment on two extreme cases as shown in Figure 4.6, which is also a head and a hip CT images with severe artifacts. The first one has damaging artifacts spreading on the whole image, and the second one is under an extremely low contrast. No matter how many number of iterations, the final visual results are still not satisfying.

		Iteration0	Iteration1	Iteration2
Hip	PSNR	21.7267	24.8536	28.3136
	RSME	0.07305	0.0571	0.0383
Tooth	PSNR	28.3125	29.0002	29.8274
	RSME	0.0384	0.0354	0.0322
Head	PSNR	24.0052	28.7754	29.7568
	RSME	0.0630	0.0364	0.0265

Table 4.2: Quantitative measurement for iterations

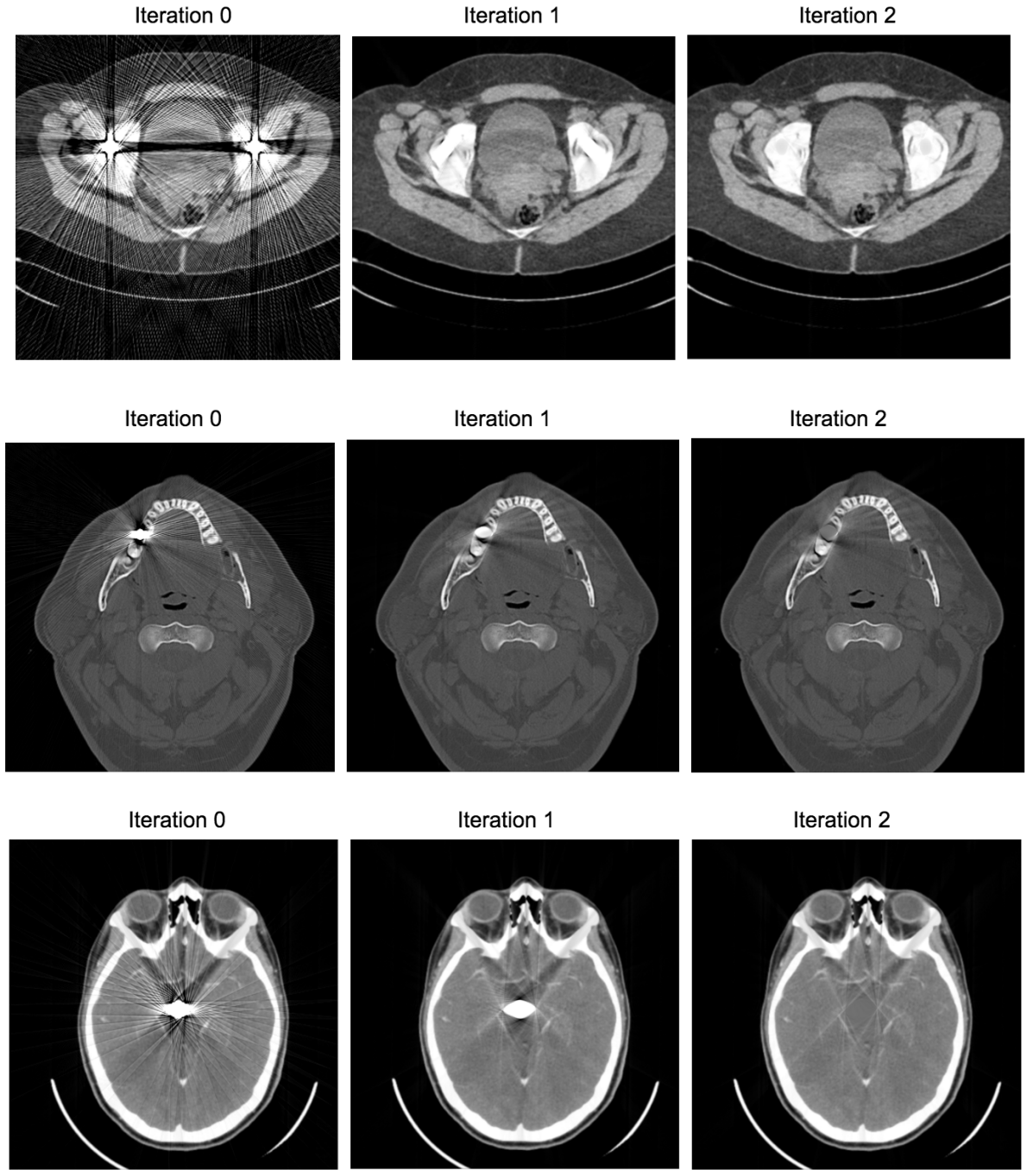

Figure 4.12: Iteration results for clinical CT phantom images

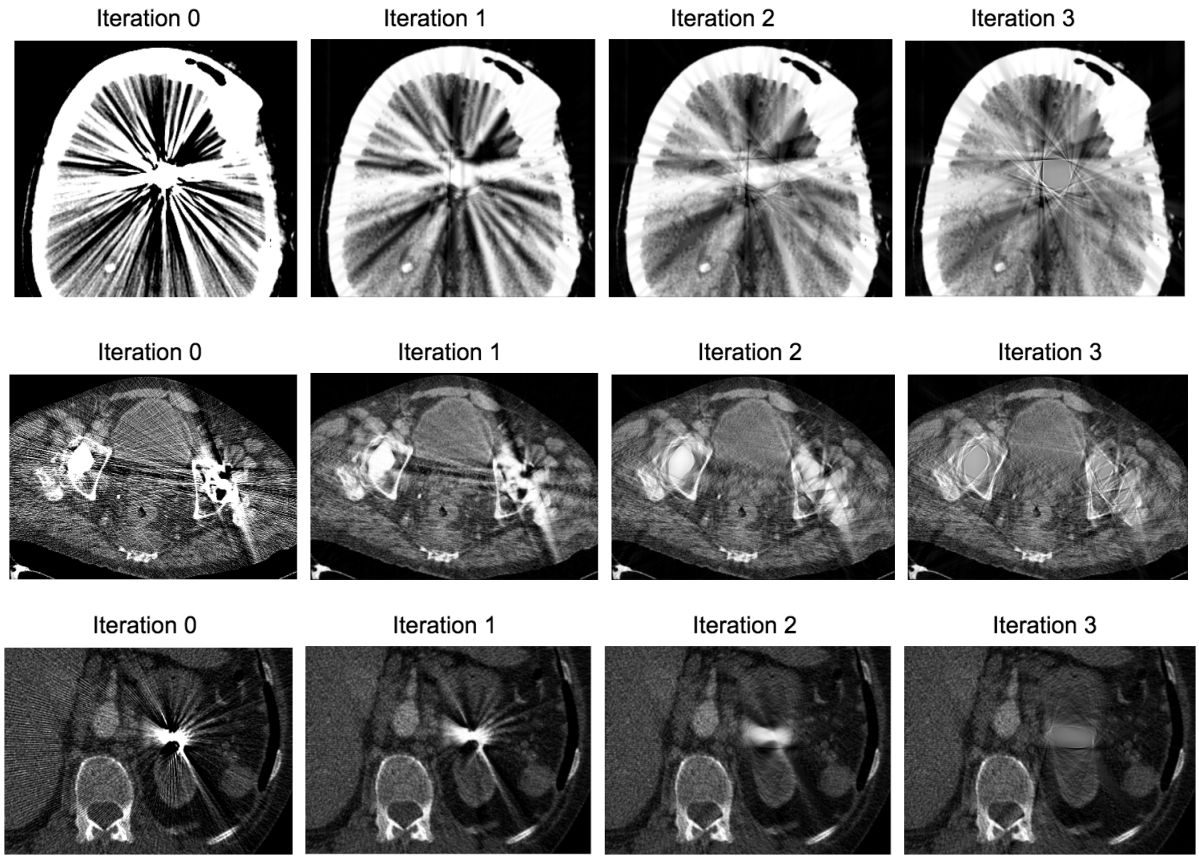

Figure 4.13: Iteration Experiment for real CT artifact images

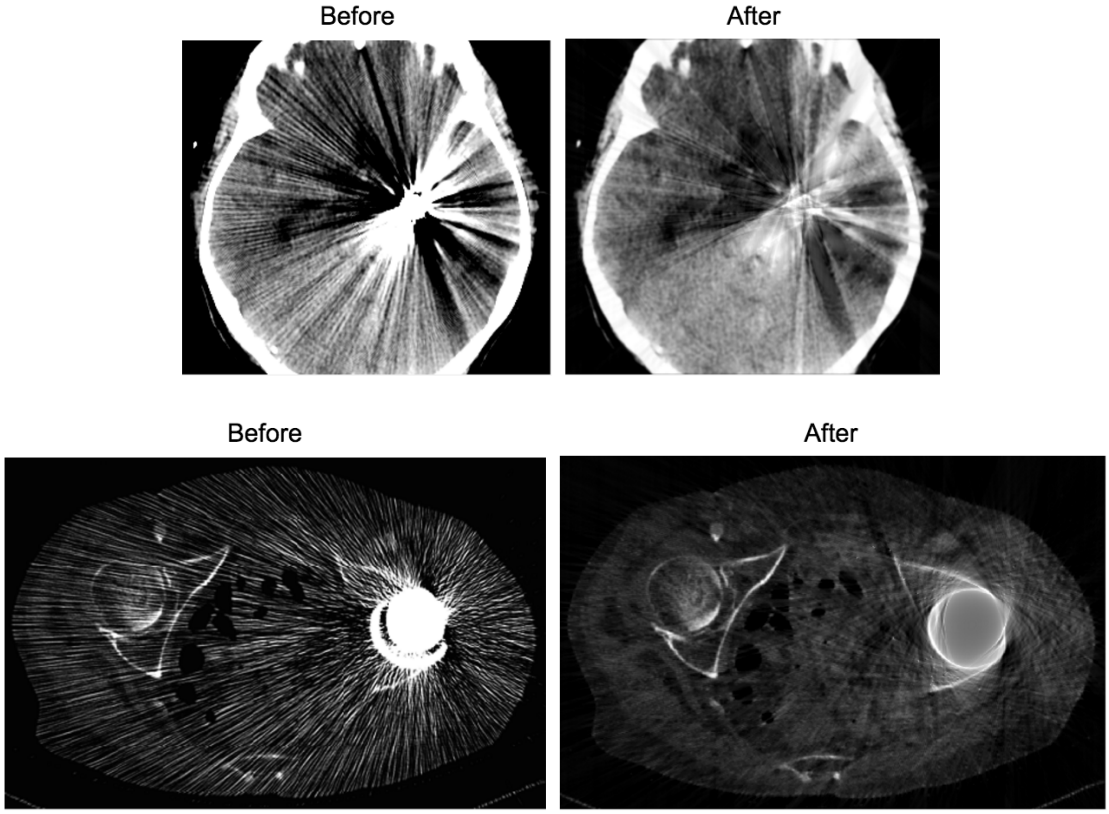

Figure 4.14: Experiment on extreme cases

Chapter 5

Conclusion

Metal artifacts on CT images often deteriorate quality in reconstructed CT images drastically. In this project, we propose and verify a novel method for metal artifact reduction based on Euler elastica inpainting technique. We observed a significant reduction of streak artifacts without introducing further artifacts in other regions; thereby, effectively improving the visual quality of CT images. Both simulated and clinical images have been tested on with a quantitative evaluation as well. Comparisons with commonly used linear interpolation MAR and normalized MAR have illustrated that Euler elastica inpainting provides a smoother and more accurate restoration than those conventional one-dimensional interpolation methods.

Future work will be devoted to the improvement to a better segmentation, more efficient inpainting discretization and the evaluation of the algorithm.

References

[1] https://radiopaedia.org/cases.

[2] Mehrsima Abdoli, Mohammad Reza Ay, Alireza Ahmadian, and Habib Zaidi. A virtual sinogram method to reduce dental metallic implant artefacts in computed tomography-based attenuation correction for PET. *Nuclear medicine communications*, 31(1):22–31, 2010.

[3] Mehrsima Abdoli, Johan R de Jong, Jan Pruim, Rudi AJO Dierckx, and Habib Zaidi. Reduction of artefacts caused by hip implants in CT-based attenuation-corrected PET images using 2-D interpolation of a virtual sinogram on an irregular grid. *European journal of nuclear medicine and molecular imaging*, 38(12):2257–2268, 2011.

[4] Mehrsima Abdoli, Rudi AJO Dierckx, and Habib Zaidi. Metal artifact reduction strategies for improved attenuation correction in hybrid PET/CT imaging. *Medical physics*, 39(6):3343–3360, 2012.

[5] Emran Mohammad Abu Anas, Soo Yeol Lee, and Md Kamrul Hasan. Removal of ring artifacts in CT imaging through detection and correction of stripes in the sinogram. *Physics in medicine and biology*, 55(22):6911, 2010.

[6] Matthieu Bal and Lothar Spies. Metal artifact reduction in CT using tissue-class modeling and adaptive prefiltering. *Medical physics*, 33(8):2852–2859, 2006.

[7] Coloma Ballester, Marcelo Bertalmio, Vicent Caselles, Guillermo Sapiro, and Joan Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE transactions on image processing*, 10(8):1200–1211, 2001.

[8] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424. ACM Press/Addison-Wesley Publishing Co., 2000.

[9] Carlos Brito-Loeza and Ke Chen. Fast numerical algorithms for Eulers elastica inpainting model. *Int. J. Mod. Math*, 5:157–182, 2010.

[10] Tony F Chan and Jianhong Shen. Nontexture inpainting by curvature-driven diffusions. *Journal of Visual Communication and Image Representation*, 12(4):436–449, 2001.

[11] Andrew Croxford, Patricio Fajnwaks, Crystal Botkin, Dana Oliver, Nghi Nguyen, and Medhat Osman. Prevalence and patterns of metal artifacts in FDG PET/CT. *Journal of Nuclear Medicine*, 51(supplement 2):2123–2123, 2010.

[12] Bruno De Man, Johan Nuyts, Patrick Dupont, Guy Marchal, and Paul Suetens. Metal streak artifacts in X-ray computed tomography: a simulation study. *IEEE Transactions on Nuclear Science*, 46(3):691–696, 1999.

[13] Michel Defrise and Grant T Gullberg. Image reconstruction. *Physics in medicine and biology*, 51(13):R139, 2006.

[14] André J Duerinckx and Albert Macovski. Nonlinear polychromatic and noise artifacts in X-ray computed tomography images. *Journal of computer assisted tomography*, 3(4):519–526, 1979.

[15] David J Eyre. Unconditionally gradient stable time marching the Cahn-Hilliard equation. In *MRS Proceedings*, volume 529, page 39. Cambridge Univ Press, 1998.

[16] Gary H Glover and Norbert J Pelc. An algorithm for the reduction of metal clip artifacts in CT reconstructions. *Medical physics*, 8(6):799–807, 1981.

[17] GH Glover. Compton scatter effects in CT reconstructions. *Medical physics*, 9(6):860–867, 1982.

[18] GH Glover and NJ Pelc. Nonlinear partial volume artifacts in X-ray computed tomography. *Medical physics*, 7(3):238–248, 1980.

[19] Godfrey N Hounsfield. Computerized transverse axial scanning (tomography): Part 1. description of system. *The British journal of radiology*, 46(552):1016–1022, 1973.

[20] Willi A Kalender, Robert Hebel, and Johannes Ebersberger. Reduction of CT artifacts caused by metallic implants. *Radiology*, 164(2):576–577, 1987.

[21] Raph Levien. The elastica: a mathematical history. *Electrical Engineering and Computer Sciences University of California at Berkeley*, 2008.

[22] RM Lewitt and RHT Bates. Image-reconstruction from projections. 3. projection completion methods (theory). *Optik*, 50(3):189–204, 1978.

[23] Antonio Marquina and Stanley Osher. Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal. *SIAM Journal on Scientific Computing*, 22(2):387–405, 2000.

[24] Simon Masnou and J-M Morel. Level lines based disocclusion. In *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, pages 259–263. IEEE, 1998.

[25] Abolfazl Mehranian, Mohammad Reza Ay, Arman Rahmim, and Habib Zaidi. X-ray CT metal artifact reduction using wavelet domain sparse regularization. *IEEE transactions on medical imaging*, 32(9):1707–1722, 2013.

[26] Esther Meyer, Rainer Raupach, Michael Lell, Bernhard Schmidt, and Marc Kachelrieß. Normalized metal artifact reduction (NMAR) in computed tomography. *Medical physics*, 37(10):5482–5493, 2010.

[27] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[28] Jianhong Shen and Tony F Chan. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, 62(3):1019–1043, 2002.

[29] Jianhong Shen, Sung Ha Kang, and Tony F Chan. Euler's elastica and curvature-based inpainting. *SIAM Journal on Applied Mathematics*, 63(2):564–592, 2003.

[30] Henrik Turbell. Cone-beam reconstruction using filtered backprojection. 2001.

[31] Ge Wang, Donald L Snyder, Joseph A O'Sullivan, and Michael W Vannier. Iterative deblurring for CT metal artifact reduction. *IEEE transactions on medical imaging*, 15(5):657–664, 1996.

[32] Oliver Watzke and Willi A Kalender. A pragmatic approach to metal artifact reduction in CT: merging of metal artifact reduced images. *European radiology*, 14(5):849–856, 2004.

[33] Ching-Ching Yang, Fong-Lin Chen, and Yeh-Chi Lo. Improving image quality of on-board cone-beam CT in radiation therapy using image information provided by planning multi-detector CT: A phantom study. *PloS one*, 11(6):e0157072, 2016.

[34] Mehran Yazdi and Luc Beaulieu. A novel approach for reducing metal artifacts due to metallic dental implants. In *2006 IEEE Nuclear Science Symposium Conference Record*, volume 4, pages 2260–2263. IEEE, 2006.

[35] Mehran Yazdi, Meghdad Asadi Lari, Gaston Bernier, and Luc Beaulieu. An opposite view data replacement approach for reducing artifacts due to metallic dental objects. *Medical physics*, 38(4):2275–2281, 2011.

[36] Shiying Zhao, DD Robeltson, Ge Wang, Bruce Whiting, and Kyongtae T Bae. X-ray CT metal artifact reduction using wavelets: an application for imaging total hip prostheses. *IEEE transactions on medical imaging*, 19(12):1238–1247, 2000.