Metal Artifacts Reduction in CT Scans using CNN with Ground Truth Elimination

by

Qi Mai

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

Supervisor: Justin W.L. Wan

Waterloo, Ontario, Canada, 2019

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

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

Abstract

Metal artifacts are very common in CT scans since metal insertion or replacement is performed for enhancing certain functionality or mechanism of patient's body. These streak artifacts could degrade CT image quality severely, and consequently, they could influence clinician's diagnosis. Many existing supervised learning methods approaching this problem assume the availability of clean images data, images free of metal artifacts, at the part with metal implant. However, in clinical practices, those clean images do not exist. Therefore, there is no support for the existing supervised learning based methods to work clinically. We focus on reducing the steak artifacts on the hip scans and propose a convolutional neural network based method to eliminate the need of the clean images at the implant part during model training. The idea is to use the scans of the parts near the hip for model training. Our method is able to suppress the artifacts in corrupted images, highly improve the image quality, preserve the details of surrounding tissues, without using the clean hip scans.

Acknowledgements

I would like to thank Prof. Justin W.L. Wan for his guidance and support toward this research project, and Prof. Yuying Li for taking her time to read my paper.

Dedication

This is dedicated to my mom and Frank, for their encouragement and support during my master's degree.

Special thanks to

Frank Lu

for thoroughly editing this paper

Amir Aziz
Baoxiu Chen
Zoey Chen
Yaxuan Guo
Lilian Lu
Kristin O'Brien
David Pow
Kun Shi
Rongrong Su
Zoe Tian
Mingyu Yang
Yilia Yu
Zoey Zou

for reading the paper and giving advice

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

When a patient has metallic objects implanted in their body, such as hip replacements, dental fillings, aneurysm clips and coils (Figure 1.1), their CT scans may contain different types of metal artifacts[4]. The typical appearance of metal artifacts are some bright and dark streaks expanding from or surrounding the metal pieces. The streak artifacts may obscure vital information for physicians to analyze CT scans and make diagnosis. In the past few decades, many metal artifacts reduction methods have been developed in attempts to combat this problem. However, so far there is no standard solution to this difficult problem in clinical CT imaging.

With the rapid advancement in medical technology and medical device industry, a growing number of people are receiving metal implants for the treatment of diseases or physical functionality support. This has led to an increase in the frequency of encountering metallic objects in CT scans. In 2012, 16 million out of 80 million CT scans performed in the U.S. contain metal artifacts[4]. Metal artifacts is a common and crucial issue in CT imaging.

During CT scanning, metal pieces block some x-rays from being detected and lead to errors in measurements at detectors. The errors are then amplified in the transformation from the detector measurements to a CT scan, resulting in metal artifacts[4]. The streak artifacts due to metallic objects overlay a large amount of information in the scans. Consequently, the quality of the images with artifacts degrades considerably. Physicians may then experience difficulties in detecting the abnormalities, tumor and other pathological changes in the scans with artifacts. To improve image quality and diagnosis accuracy, the presence of severe streak artifacts necessitates the need for developments in metal artifacts reduction.

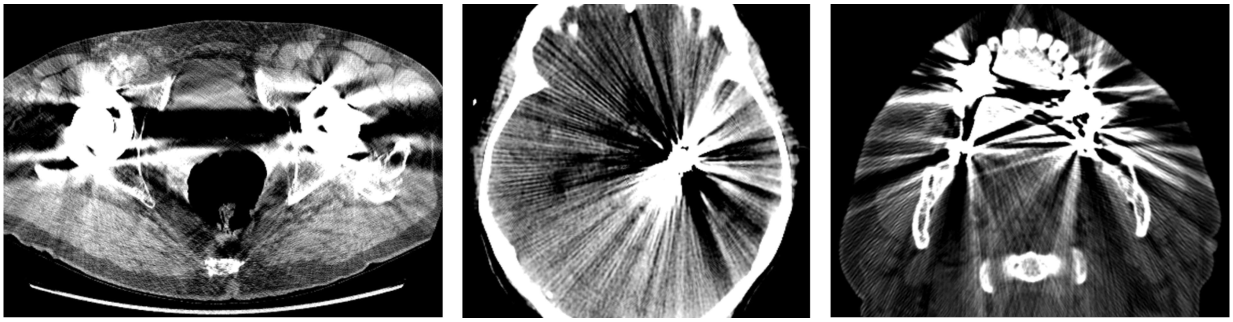


Figure 1.1: Examples of metal artifacts in the CT images of hip scan[4], brain scan[27] and dental scan[27].

The metal artifacts reduction problem is challenging. It is difficult to infer the information altered by the metallic objects. This problem has been studied for a few decades and different approaches have been developed to improve the quality of the CT scans with artifacts. The existing metal artifacts reduction methods usually belong to one of the following three categories: physical correction, iterative reconstruction and sinogram correction[1][4]. Some approaches are the combination of the methods in more than one category. In physical correction, beam hardening reduction[19] and filtered back-projection[30] are built into modern scanners, aiming to correct the errors during CT scan reconstruction. Unfortunately, the results are often not satisfactory and have visible streak residues. Iterative algorithms remove the artifacts from the corrupted CT scans directly[31]. But the computation for iterative reconstruction methods is expensive, and consequently is inefficient. Sinogram correction techniques include interpolation and inpainting. Linear and spline interpolations are intuitive and widely used interpolation methods to solve the sinogram correction problem[18]. However, linear interpolation introduces new artifacts and distorts the shape of tissue surrounded by the metal[1]. Meanwhile, cubic spline interpolation does not return a smooth sinogram view[18]. Since these two interpolation methods are usually carried out in one dimension, the spatial information in the sinogram is unused. Image inpainting[2] methods are able to make use of spatial information to make better sinogram corrections. Many inpainting methods in the literature are PDE-based. The Euler's Elastica inpainting technique corrects the values of traces in sinograms while preserving sharp edges and curvature[35, 14]. Total Variation inpainting can inpaint sinograms smoothly[34]. They can suppress most of the artifacts but require complicated mathematical calculations which are time-consuming and computationally expensive.

[1]Sinogram will be introduced in Chapter 2.
[2]Inpainting will be introduced in Chapter 3.

Convolutional Neural Network (CNN)[21] has been widely used and attains extraordinary achievements in computer vision and pattern recognition[22]. Meanwhile, CNN models have also been developed for image restoration, and many of which are adopted from the models used for image classification and image segmentation[7]. Researchers in medical imaging have attempted to take advantage of CNN and apply it to solve the problems in medical imaging. To tackle the metal artifacts reduction problem in CT scans, researchers usually take artifacts contaminated images for restoration or erroneous sinograms for inpainting. Zhang and Yu[36] proposed a two-phase CNN based method to suppress metal artifacts, by combining the information from the artifact-corrupted images and pre-corrected images obtained from the interpolation method. Using a special loss function for a deep neural network, Gjesteby et al.[12] developed a model for artifacts suppression and tested it on phantom images. Xie et al.[33] applied deep residual learning using the improved GoogleNet to reduce metal artifacts on phantom images. Ghani and Karl[11] chose to inpaint sinograms by learning the values to be filled into metal traces using CNN.

Many of the proposed CNN models can suppress metal artifacts effectively. However, there is a commonly neglected issue in the above mentioned studies – the existence and the availability of the labeled data, that is images, which are used as the target for model training. In order for a CNN to learn how to reduce artifacts in corrupted images or how to inpaint sinograms, it needs artifact-free images or correct sinograms as training targets. Without these images, the learning task cannot be completed. Unfortunately, the existing methods in the literature typically assume the existence of such clean reference images. A good model indeed plays an important role in applying supervised learning to solve a problem, but the significance of the data used for training, particularly the target data, should not be ignored.

In this thesis, we focus on reducing metal artifacts in the hip scans of patients with hip prosthesis. In clinical practices, we do not have artifact-free hip scans as targets for a model to learn from. To solve this issue, we propose an innovative method that conducts model training on the scans near the hip with simulated artifacts. Artifacts suppression is then carried out on the actual artifact-corrupted hip scans using the trained model. Our approach eliminates the need for clean hip scans in model training and produces artifact-free hip scans using model prediction.

The rest of the thesis is organized as follows. Chapter 2 provides some background knowledge for a better understanding of the metal artifacts reduction problem and our approach. Chapter 3 describes our proposed method in details and Chapter 4 validates the method by presenting different experimental results. Chapter 5 concludes the thesis and points out potential directions for future research.

Chapter 2

Background

In this chapter, we will introduce some background knowledge for a better understanding of the subsequent chapters. Section 2.1 gives a summary of the mechanism of CT machines, the CT imaging process and the formation of metal artifacts. Section 2.2 provides a short introduction to supervised learning, neural networks and optimization techniques.

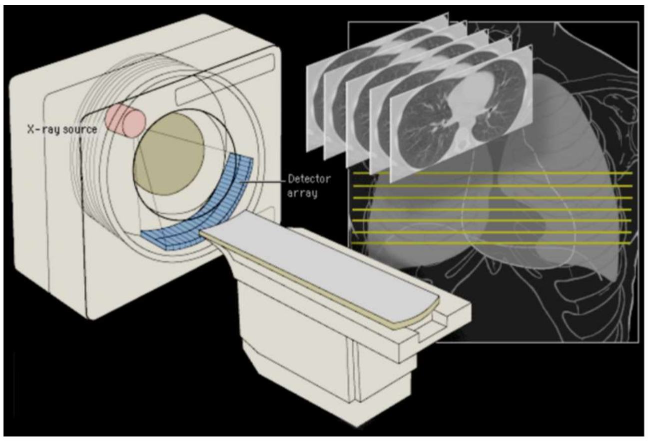
2.1 CT Imaging and Metal Artifacts

Computed Tomography (CT) is a computerized x-ray imaging procedure in which x-ray beams are generated to pass through the patient's body while rotating around. The signal collected at the detector will be processed and transformed by the machine to produce cross-sectional images, or sometimes called slices, of a body. For different diagnostic and therapeutic purposes in medical disciplines, CT scan can be conducted on different parts of the body, such as head, cardiac, abdomen, pelvis, or even the entire body. Thus, CT is widely used to detect diseases and conditions of a patient's body and help pinpoint the location of infection, tumor, blood clots, etc[4].

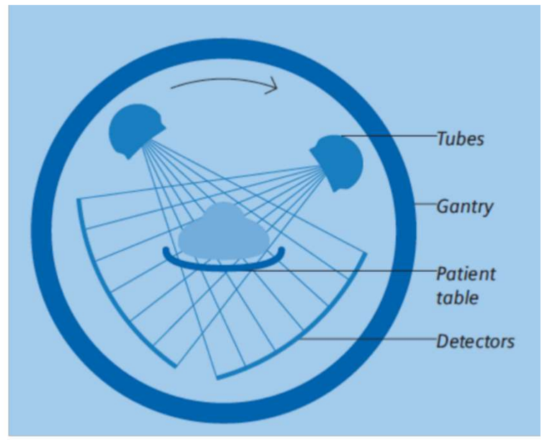
2.1.1 CT Mechanisms

CT scanner uses special x-ray equipment to obtain image data from different angles around the body and creates the cross-section images of body tissues, organs, and bones.

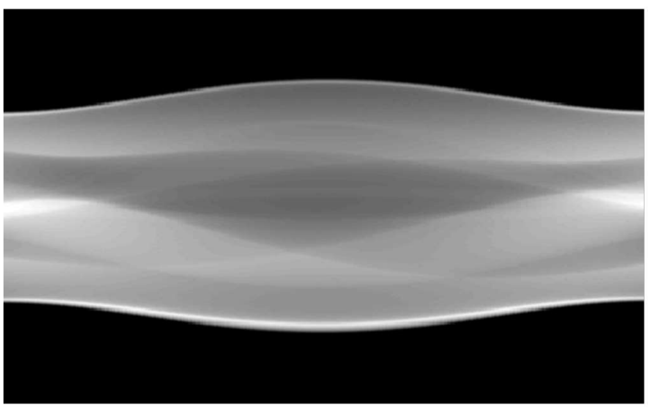
The scanning unit is called gantry (Figure 2.1 A), and consists of two components – the transmitter (also known as the x-ray unit) and the receiver (also known as the data



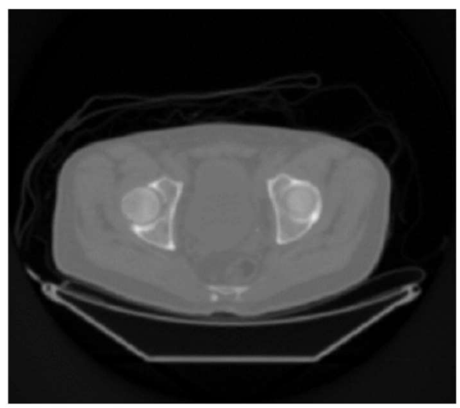
A



B



C



D

Figure 2.1: CT Mechanism. Picture A[17] shows the gantry in a CT machine and some consecutive slices of chest. Picture B[15] illustrates the x-ray emission and detection inside the gantry. Picture C is a sinogram recorded in CT machine after 180° gantry rotation. Picture D is a CT scan of the hip[13].

acquisition unit or the x-ray detector). The x-ray unit emits multiple beams which will be detected at the detector after passing through the body of a patient (Figure 2.1 B). Before whole-body CT scanning, the patient is asked to lie on a table, which will slowly move through the gantry during scanning. The patient is fixed to the table throughout the process and is transported continuously through the scanning field. As the patient being moved forward, the gantry performs multiple 360° rotations. For every rotation, at each small angle, the detector records the attenuation coefficients into a one-dimensional array. After data collection, the computer inside the scanner will concatenate the arrays of attenuation coefficients from different angles together into a 2D matrix, where each column in the matrix is the one-dimensional record from a different angle (Figure 2.1 C). For example, if the detector collects data at each 1° in one rotation, the resulting matrix will have 360 columns. Such a matrix represents an intensity record for a slice of the body, also known as a sinogram[1]. Mathematically, the process of acquiring a sinogram is called radon transform. The inverse radon transform, which is also called Filtered Back Projection (FBP), is the method of CT scan reconstruction from a sinogram (Figure 2.1 D). Different objects, such as organs, tissues, bones, and air, have different attenuation coefficients[25]. Objects with high attenuation will appear brighter and objects with low attenuation will appear dimmer. Hence, the brightness of different objects will be noticeably different in the reconstructed CT scans.

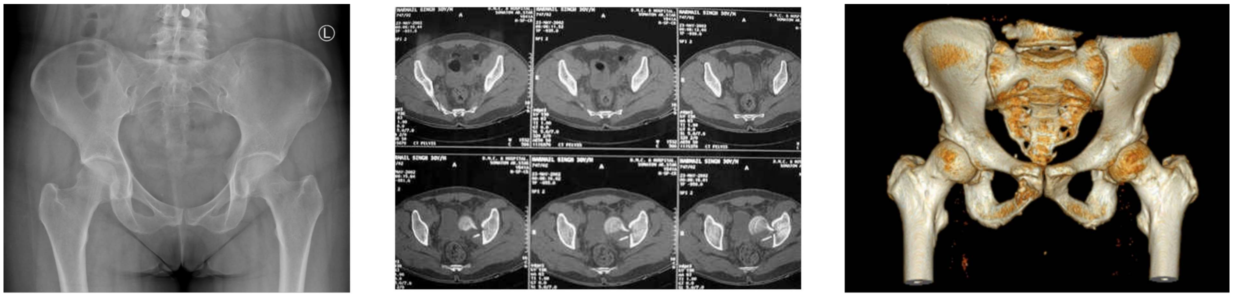
Compared to conventional X-ray images (Figure 2.2 A), which are usually taken one at a time, multiple and successive slices contain more information when they are stacked in order (Figure 2.2 B). In addition, through sophisticated mathematical calculations and transformations, the consecutive slices can be used to construct a three-dimensional digital image or even build a physical model (Figure 2.2 C). This helps physicians identify the location of possible tumors, abnormalities, and other pathological changes more accurately.

2.1.2 Radon Transform and Filtered Back Projection

Radon transform and FBP (inverse radon transform) play important roles in producing CT scans.

Suppose a CT slice of body is $[f(x, y)]$, where (x, y) is the coordinate of a pixel and $f(x, y)$ is the pixel value. Radon transform calculates the projection of a slice from different

[1]In one full rotation, the sinogram from 0° to 180° is always identical to the sinogram from 180° to 360° (see Figure 2.1 C). We usually use the data collected in the first half of the full rotation. So the number of columns in the matrix in the example can be reduced to 180.



A B C

Figure 2.2: Examples of (A) the X-ray image[23], (B) the CT scan[29] and (C) the 3D CT model of hip[9]

angles. The sinogram for a slice $[f(x,y)]$ is

$$R(\rho,\theta)[f(x,y)] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(\rho - x\cos\theta - y\sin\theta)dxdy, \tag{2.1}$$

where ρ is the perpendicular distance from a line to the origin, and θ is the angle formed by the distance vector. In CT scanner, ρ is the radius of the gantry, i.e. the half of the distance between the x-ray emitter and detector, and θ is the angle that the gantry rotates to. δ is the delta function where $\delta(x) = \infty$ when $x = 0$ and $\delta = 0$ otherwise.

The inverse radon transform is as following,

$$f(x,y) = \int_{-\infty}^{\infty}\int_{0}^{\pi} G(\upsilon\cos\theta, \upsilon\sin\theta)|\upsilon|e^{2\pi i\upsilon(x\cos\theta + y\sin\theta)}d\theta d\upsilon, \tag{2.2}$$

where $G(\upsilon\cos\theta, \upsilon\sin\theta) = \int_{-\infty}^{\infty} R(\rho,\theta)e^{-2\pi i\rho\upsilon}d\rho$. Here, $G(\upsilon\cos\theta, \upsilon\sin\theta)$, represented in polar coordinates (υ,θ), is the two dimension Fourier Transform of $f(x,y)$. $|\upsilon|$ compensates for the high density of projections near the origin in frequency domain. It is essentially a high pass filter, and hence the "filter" in Filtered Back Projection.

2.1.3 Metal Artifacts Formation

Metal streak artifacts are very common in CT scans when a patient has a metal insertion. Usually, patients are required to take off all removable metallic objects before CT scanning. For those items which cannot be removed, such as dental fillings, surgical clips and hip

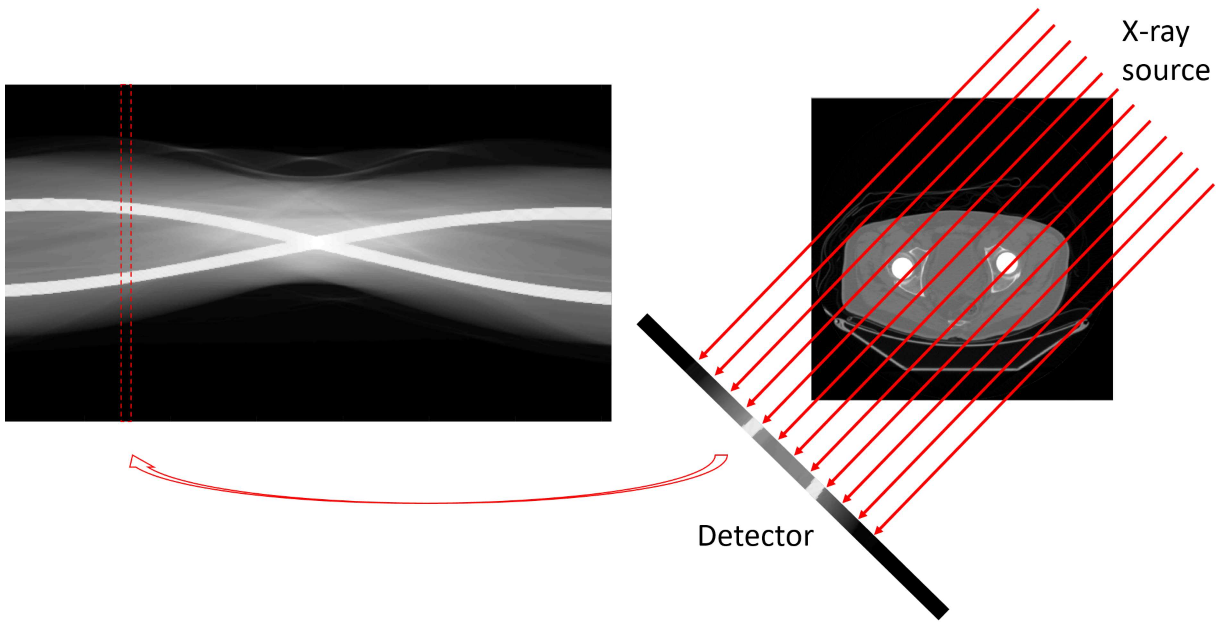


Figure 2.3: Metal artifacts formation. A column in the sinogram is the data recorded in the detector after x-ray beams are emitted and pass through the body.

replacement prosthesis, it is difficult to avoid artifacts in the scans. Metallic materials have high attenuation coefficients, much higher than that of bones and other tissues. As a result, the metal implants appear extremely bright, leading to strong artifact effects at its surroundings or even across the whole image.

Metal streak artifacts are mainly due to the errors in the sinogram and the filters used in FBP. The detector records relatively high values for those x-rays passing through metal pieces since metallic objects have higher attenuation coefficients. Since each column in a sinogram is the attenuation coefficients recorded at an angle, as the gantry rotates, the high values resulting from metal pieces form bright traces in the sinogram (see Figure 2.3). However, the sinogram of an image without artifacts has no such bright bands (Figure 2.1 C). The errors in sinograms led by inaccurate measurements from metallic objects result in the streaks in CT scans. During CT scan reconstruction, FBP uses a high pass filter, which further exaggerate the difference between detected values[4]. Since the values at the bright traces are much higher than others in sinograms, the exaggeration in value difference brings up artifacts. The filter creates great distinction in the color of metallic objects from other tissues, as well as bright and dark streaks originating from the metal in the reconstructed image.

Metallic objects and the resulting artifacts in CT scans create a large obstacle in CT scan interpretation. Artifacts may cover crucial information in CT scans and hinder diagnosis[2]. Around 20% of CT scans taken in the U.S. annually have metal artifacts[4]. It is essential and necessary to develop effective and efficient methods to solve this issue, in order to provide physicians with higher quality scans for diagnosis.

2.2 Supervised Learning, Neural Networks and Training

Supervised learning is widely used for predicting unseen or future events using existing information, and is the foundation of our proposed method for reducing metal artifacts. This section will introduce supervised learning, the supervised learning model we use in our approach, and the methods to train a model.

2.2.1 Supervised Learning

Supervised learning is one of the main branches of machine learning. Some examples of supervised learning problems could be predicting the future sales of products offered by a company or identifying the digits in images of house numbers. Typically there are three components in a supervised learning problem: an outcome measurement (label), a set of features (input) and a model that predicts the outcome using the features. Outcome measurements can be numerical (the highest temperature of the next day) or categorical (win or lose a game). Features can be of these two types as well, and usually have a relationship with the outcome. Using the data we collect, we can build a prediction model that predicts the outcomes of new situations given features.

One thing that makes supervised learning distinct from other machine learning tasks is the label of an observation. In supervised learning, we want a model to learn the transformation from input to label using the collected data, where the transformation is specified by a set of parameters. Suppose we have n observations. The model in supervised learning can be seen as a function $f(\boldsymbol{\theta}; X)$, where $\boldsymbol{\theta}$ contains the parameters of the model and $X = \{x_1, x_2, ..., x_n\}$ is the input data. The labeled data is $y = \{y_1, y_2, ..., y_n\}$. We call each (x_i, y_i) as an observation. The goal of supervised learning is to find the parameter $\hat{\boldsymbol{\theta}}$ such that the difference between $f(\boldsymbol{\theta}; X)$ and y is minimized. Without y, the parameter

$\hat{\boldsymbol{\theta}}$ cannot be found. Therefore, the labeled data y plays a significant and essential role in supervised learning.

As we explained above, we aim to minimize the difference between $f(\boldsymbol{\theta}; X)$ and y. Such difference is called loss function. Let the loss function be denoted by l. A supervised learning problem is then defined as:

$$\min_{\boldsymbol{\theta}} l(\boldsymbol{\theta}) := l\left(f(\boldsymbol{\theta}; X), y\right). \tag{2.3}$$

Specifically, the loss function is a function of $\boldsymbol{\theta}$ as well, written as $l(\boldsymbol{\theta})$. The process of minimizing the loss function to obtain $\hat{\boldsymbol{\theta}}$ is called training. By applying optimization techniques, which will be discussed in section 2.2.3, we can solve the minimization problem to find $\hat{\boldsymbol{\theta}}$.

Once we find $\hat{\boldsymbol{\theta}}$, we have a trained model and are ready to apply the it on new data X'. Let y' denote the label of new data. If $l(f(\hat{\boldsymbol{\theta}}; X'), y')$ is low and is similar to $l(f(\hat{\boldsymbol{\theta}}; X), y)$, we may conclude that the model is well trained and able to produce reliable prediction.

2.2.2 Neural Network

There are various models in supervised learning. Some model has an explicit form of f, such as linear regression, and some does not. The model, neural network, we use in our approach is defined implicitly. It is inspired by the biological nervous system in animals. When information is received in a neural network, it will be passed through different neurons in the form of signals. As the brain receives signals, it will makes commands for reactions. In this section, a brief introduction of neural network will be given, as well as the network components and structures used in the convolutional neural network (CNN) model in our approach.

Mechanism of Neural Networks

In the implementation of neural network, neurons are seen as nodes and connections are seen as edges. The signals transmitted in a network are real numbers. When a node receives signals, it processes the numbers by applying summation followed by a function. Such functions are typically non-linear functions and are called activation functions, which will be introduced in section 2.2.2. The result of the activation function is the output of the node. The output will then be passed to the nodes that are connected with the current one via outgoing edges. Every edge in the network has a weight. During transmission, the

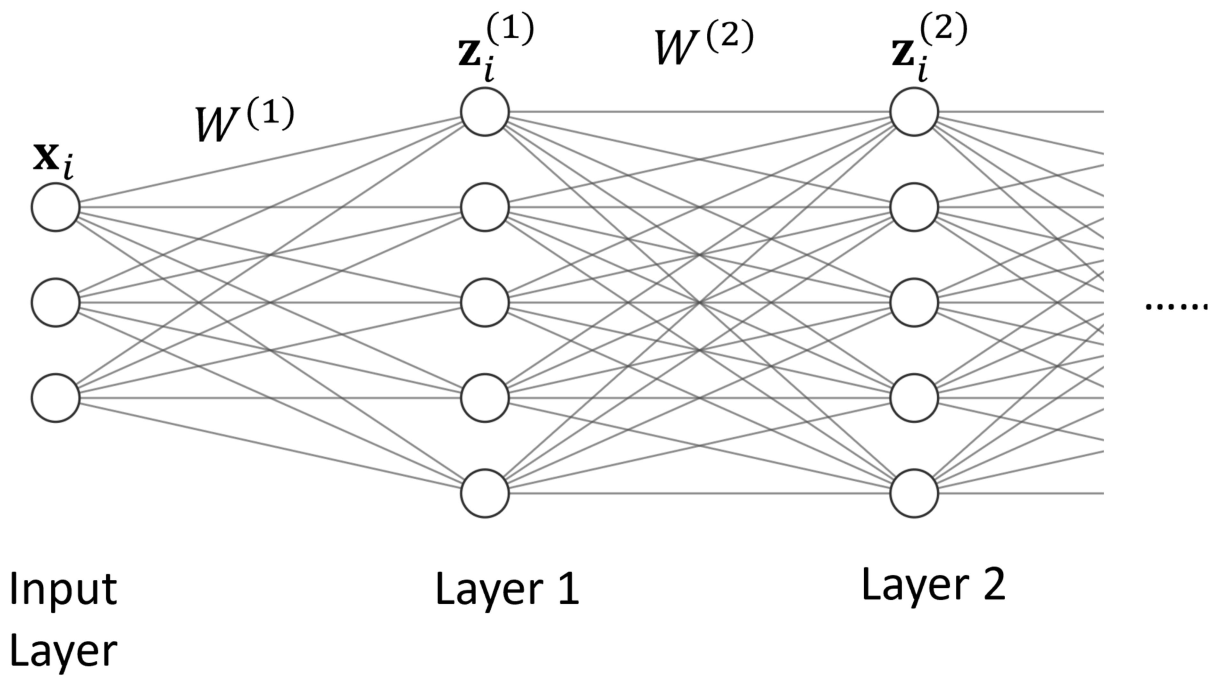


Figure 2.4: The first 2 layers of a feed-forward neural network model. Each circle represents a node and each line represents a connection edge.

output from the node will be multiplied by the weight of each connection edge. Since the weights on different outgoing edges vary, a single node will pass distinct signals to different nodes.

Since the connections among neurons in the nervous system are too complicated to imitate for data modeling, neurons are partitioned into different sets for constructing a neural network model. Each set of nodes is called a layer. An input layer, multiple intermediate layers, and an output layer collectively form a neural network model. Each node in an intermediate layer has both incoming edges and outgoing edges. One of the most simple neural networks is the sequential feed-forward neural network that consists of several layers of nodes and sequentially transmits information through layers (Figure 2.4).

Mathematically, we present a feed-forward neural network in the following way. Suppose we have a set of sample data $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$, where each $\mathbf{x}_i \in \mathbb{R}^3$. In this example (see Figure 2.4), the first two intermediate layers have 5 nodes individually. The weights of the connection edges between input layer and the first layer are stored in a matrix $W^{(1)} \in \mathbb{R}^{5\times3}$, and the weights between the first layer and the second layer are represented as a matrix

$W^{(2)} \in \mathbb{R}^{5\times5}$. Let σ_1 and σ_2 be the activation functions to process outputs from layer 1 and layer 2 respectively. To learn from a sample $\mathbf{x}_i$, for each of its 3 features, the information needs to be passed to every node in the first layer. The input to the first layer for a sample is $W^{(1)}\mathbf{x}_i + \mathbf{b}^{(1)}$, where vector $\mathbf{b}^{(1)}$ is a bias term. After applying the activation function σ_1, we obtain the output of the first layer for sample $\mathbf{x}_i$, $\mathbf{z}_i^{(1)} = \sigma_1(W^{(1)}\mathbf{x}_i + \mathbf{b}^{(1)})$. Repeating this process, we can acquire the output of the second layer for sample $\mathbf{x}_i$, $\mathbf{z}_i^{(2)} = \sigma_2(W^{(2)}\mathbf{z}_i^{(1)} + \mathbf{b}^{(2)})$, where $\mathbf{b}^{(2)}$ is the bias term in the second layer.

In a neural network model, the weights are essentially the parameters in the $\boldsymbol{\theta}$ of the loss function as introduced in section 2.2.1.

Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is a type of neural network for imagery analysis, such as image classification, image segmentation, object detection, etc. A CNN usually consists of an input layer, an output layer, and multiple convolutional layers. The mechanism of convolutional layer inherits from the convolution operation in signal/image processing. In such a layer, a sliding window, called a kernel, moves across an image to capture spatial and pattern features. Multiple kernels can be used in one convolutional layer to extract various characteristics and information.

Suppose we have an image of size 4×4 and each pixel value is x_{ij}, where row $i = 1, 2, 3, 4$ and column $j = 1, 2, 3, 4$. We want to use a 2×2 kernel to convolve the image (see Figure 2.5). Let k_{pq} be the weights in the kernel, where $p = 1, 2$ and $q = 1, 2$[2]. Convolution typically starts from the top left corner of the image. The result of the corner is $k_{11}x_{11} + k_{12}x_{12} + k_{21}x_{21} + k_{22}x_{22}$. Then the kernel slides towards the right for 1 pixel[3] and gives the resulting value $k_{11}x_{12} + k_{12}x_{13} + k_{21}x_{22} + k_{22}x_{23}$. Repeating the process, the kernel finishes convolving one horizontal line. It then moves to the first two columns in the second and the third row, i.e. 1 pixel downward[4], and repeat the convolution as previous. By stacking and concatenating the resulting values in the order we proceed convolution, we obtain a 3×3 convoluted "image". We then obtain the output of a convolutional layer with one kernel after applying an activation function.

Convolutional layer is designed to improve the efficiency in a neural network model for image data. Considering the number of pixels in an image, which could be 10,000 for a

[2]For an RGB color image with red, green and blue channels, the kernel has depth 3, the same as the number of image channels.

[3]The sliding step is not necessarily 1. If the step size is 2, then the kernel moves 2 pixels to the right. The step size is usually smaller than or equal to the kernel size.

[4]It again depends on the step size.

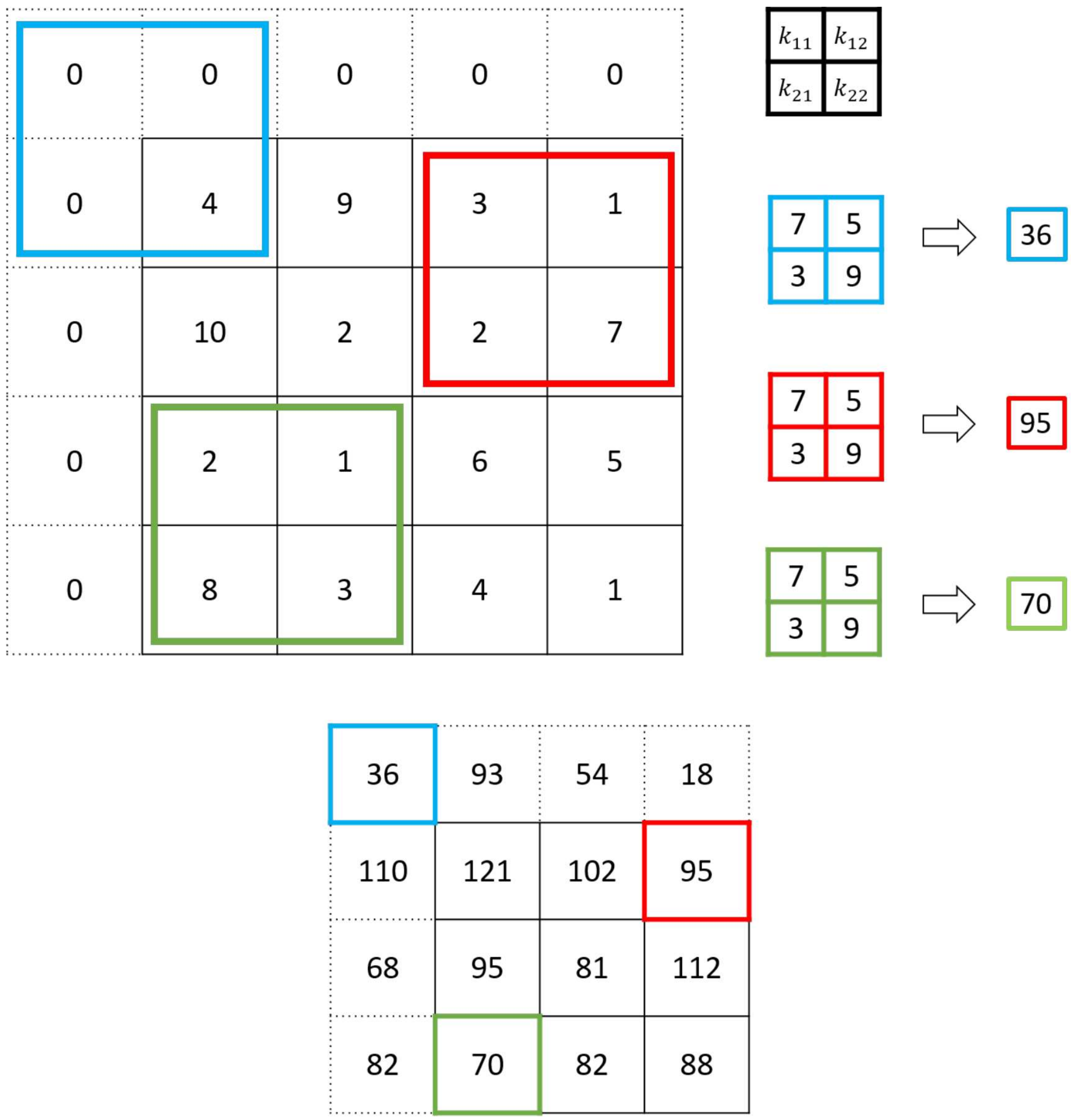


Figure 2.5: Kernel operations and margin padding in a convolutional layer. The image is of size 4×4 and is padded with zeros to the left margin and top margin. Convolution is conducted by a 2×2 kernel with weights $k_{11} = 7$, $k_{12} = 5$, $k_{21} = 3$ and $k_{22} = 9$. Blue, red and green squares illustrate the convolution at the margin and in the image. The image below is the result of convolution with the kernel.

100×100 image (small) or over 1 million for a normal size photo, it is computationally expensive to assign weights for each pixel. For the photo with 1 million pixels, if we use 20 kernels of size 5×5 each, we will have $20 \times (5 \times 5) = 500$ parameters (weights), which is much less than 1 million. Through deploying kernels, we can not only extract spatial information but also reduce a large amount of computing resource usage.

Convolutional layer is the most important component in the model we use for metal artifact reduction. Through connecting 6 consecutive convolutional layers, our model learns the way to correct sinograms during model training. The model will be introduced and explained in detail in section 3.3.

Margin Padding

In the above example (Figure 3.3), a 4×4 image becomes a 3×3 image after convolution. Sometimes it is not desirable to alter the sizes of the input and the output of a layer. To resolve this issue, we can pad some zeros around the image. For instance, we can pad a row and a column of zeros above and at the left of the image, so that the padded image now is of size 5×5. The convolution operates in the same way as before. When the window slides through the row of zeros and the column of zeros, only the weights that overlap with the pixels in the image will be used and the weights corresponding to the padded zeros will be ignored. At the top left corner, the convoluted result would be $k_{22}x_{11}$; after one step to the right, the result is $k_{21}x_{11} + k_{22}x_{12}$; after one step downward at the very left, the result is $k_{12}x_{11} + k_{22}x_{21}$. By padding some zeros along the margins, we obtain the convolution result to be a 4×4 image, which is the same size as the input.

Activation Function

Activation function is an imitation of the firing process in a neuron that determines what information will be passed to the connected neurons and when to pass them[16]. It is used after each intermediate layer in a neural network. A typical activation function is non-linear. Commonly used activation functions include sigmoid: $\sigma(x) = 1/(1 + e^{-x})$, and Rectified Linear Unit (ReLU): $\sigma(x) = x$ if $x \geq 0$ and $\sigma(x) = 0$ otherwise. A commonly used variant of ReLU is called Leaky ReLU, and is represented as $\sigma(x) = x$ if $x \geq 0$ and $\sigma(x) = \alpha x$ otherwise, where α is a hyperparameter and usually less than 1.

In our model, we adopt Leaky ReLU as the activation function after each convolutional layer.

2.2.3 Training and Optimization

As explained in section 2.2.1, we want to find the best parameter $\hat{\boldsymbol{\theta}}$ for a model. To achieve this, we need to minimize a loss function as presented in 2.3. Here, optimization techniques can fulfill the minimization goal. In this section, we will briefly introduce loss functions and three optimization techniques: gradient descent, stochastic gradient descent, and ADAM.

Loss Function

A loss function represents the difference between the predicted label and the actual label of an observation. The smaller the value of the loss, the more accurate the prediction we have. Two commonly used loss functions in the literature for training a neural network are logloss and mean squared error. Specifically, logloss is used for classification problems and mean square error is for regression problems.

For logloss in classification, the function to be minimized is

$$l(\boldsymbol{\theta}) := -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} y_{ik}\log(f(\boldsymbol{\theta};\mathbf{x}_i)_k), \tag{2.4}$$

where there are n samples and K classes; $y_{ik} = 1$ if sample i is in class k and $y_{ik} = 0$ otherwise; $f(\boldsymbol{\theta};\mathbf{x}_i)_k$ is the predicted probability of sample $\mathbf{x}_i$ to be classified to class k by the model.

For mean squared error in regression, the function to be minimized is

$$l(\boldsymbol{\theta}) := \frac{1}{n}\sum_{i=1}^{n}(y_i - f(\boldsymbol{\theta};\mathbf{x}_i))^2, \tag{2.5}$$

where y_i and $f(\boldsymbol{\theta};\mathbf{x}_i)$ are the actual value and the predicted value of sample $\mathbf{x}_i$ respectively.

For our model, which will be introduced in section 3.3, its output data and target data are both images. Logloss, which is in regard to classification probability, is not a suitable loss function when the target data is images. Thus, we choose to use mean squared error to measure the difference between outputs and targets.

Gradient Descent

With a defined loss function, the next step is training the model, i.e. minimizing the loss function. One of the most classical methods is gradient descent.

To find the minimum of a loss function $l(\boldsymbol{\theta})$ using gradient descent, we start from some point $\boldsymbol{\theta}_1$ and take a step toward the direction that gives the greatest decrease in l. This direction is the negative gradient at $\boldsymbol{\theta}_1$, denoted by $-\nabla l(\boldsymbol{\theta}_1)$. After taking the first step, we reach a new point, say $\boldsymbol{\theta}_2$. We then take another step by calculating the direction with steepest drop. The procedure of moving from $\boldsymbol{\theta}_1$ to $\boldsymbol{\theta}_2$ is called an update. The general formula of one update is $\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - \alpha \nabla l(\boldsymbol{\theta}_i)$, where α is a hyperparameter for step size and i is the iteration number. The updating process gives a decreasing sequence of $l(\boldsymbol{\theta})$, i.e. $l(\boldsymbol{\theta}_1) \geq l(\boldsymbol{\theta}_2) \geq l(\boldsymbol{\theta}_3) \geq ...$. When the difference between $\boldsymbol{\theta}_{i+1}$ and $\boldsymbol{\theta}_i$ is close[5], we can stop updating and conclude that we have reached a local minimum of function l.

Stochastic Gradient Descent

In gradient descent, the gradient $\nabla l(\boldsymbol{\theta}_i)$ is calculated from the entire data set (X, y) in an update step. This is computationally expensive, especially when there are millions or billions of observations. By deploying stochastic gradient descent[5], the gradient computation is based on a subset of observations, called a batch of data, in one update. The batches of data are obtained by random partition in the original data set. Each batch has the same size. With a lighter computation burden at each step, the training process could be accelerated.

Batch Normalization

Batch normalization is known for its effectiveness regarding performance, efficiency, and stability in training a neural network. Suppose the network is currently learning from a batch of data B of size n. For the j^{th} feature, the mean of the data in this batch is

$$\mu_{Bj} = \frac{1}{n} \sum_{i=1}^{n} x_{ij}$$

and the variance is

$$\sigma_{Bj}^2 = \frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \mu_{Bj})^2.$$

The normalized data of feature j is then $x'_{ij} = (x_{ij} - \mu_{Bj})/\sigma_{Bj}$. We perform the normalization for each feature individually, and pass the final normalized data to the next layer.

[5] The stopping threshold of the difference between $\boldsymbol{\theta}_{i+1}$ and $\boldsymbol{\theta}_i$ is user-defined. Sometimes the threshold is defined as the difference between $l(\boldsymbol{\theta}_{i+1})$ and $l(\boldsymbol{\theta}_i)$. Either way, the value is typically 10^{-9} or smaller.

ADAM

In gradient descent and stochastic gradient descent, we need to determine the step size of each update manually. At the beginning of training, we may use a relatively large step size to move towards a local minimum faster. As training proceeds, we get closer and closer to the local minimum and do not want to miss it by taking big steps, so we decrease the step size. It could be hard to adjust of the step size accordingly during training. Thus, ADAM[20], which is a variant of stochastic optimization and known as adaptive moment estimate, takes a user-defined initial step size at the beginning and modifies it as training proceeds. It can adapt the step size automatically, removing the need to manually modify the step size during training.

Chapter 3

Methodology

Metal artifacts reduction is still an unsolved problem in CT imaging. The existing approaches based on supervised learning require clean scans of patients with metallic implants. In reality, such scans do not exist. For this reason, a novel supervised learning based method is required to eliminate the need of clean images.

In this chapter, we outline our proposed method to reduce metal artifacts, and discuss each component in the method. Our approach focuses on the artifacts resulting from hip replacements only, and deploys CNN for sinogram inpainting. The key to this approach is getting rid of the requirement of clean hip scans that are not available clinically, by generating a training data set from the scans of the parts above and below the hip. After the model is trained, it can then be used for artifacts suppression on corrupted hip scans.

The ground truth assumption issue is discussed in more details in section 3.1 and our method is outlined in section 3.2. The CNN-based sinogram inpainting model is described in section 3.3. It is based on the work of Ghani and Karl[11]. The input images to the model are the sinograms with bright bands resulting from metallic objects, and the target images are the sinograms without those bands. The model will be trained to inpaint the bright belts. In order to be efficient and generalizable, the model should be applicable on multiple patients. Our strategy to create a generalizable model is discussed in section 3.4. An additional component illustrated in section 3.5 is K-means segmentation which is utilized for segmenting the metallic objects from the artifact-corrupted images. It is an important part in training data generation.

3.1 Ground Truth Assumption

To train a neural network for reducing metal artifacts in CT images, we need to have artifact-corrupted images as inputs and artifact-free images as targets for a CT scan-based model, or their corresponding sinograms as inputs and targets for a sinogram-based model. In the literature, we often call such target data the ground truth.

However, in clinical practices, the artifact-free scans at the part with inserted metallic objects are not available. When a patient with hip prosthesis takes a whole-body CT scan, the metallic object is already implanted and so it will lead to artifacts immediately. Even if the patient takes CT scans before the replacement implant and after the implant, it is not guaranteed that the patient can stay in the same position and remain in the same physical situation in the two screenings. We may obtain the error in the parts other than the locations of metal pieces. Therefore, clinically it is impossible to acquire a pair of artifact-corrupted and artifact-free CT images with everything else in the image held the same.

As explained in section 2.2.1, supervised learning requires input-target pairs for training. An inpainting model needs target examples in order to learn how to map an artifact-corrupted image to an artifact-free image, or how to map a sinogram with bright traces to a smooth sinogram without traces. Ground truth images are strict requirements in order to actually apply neural networks for metal artifacts reduction. However, for the proposed methods in the literature, the target images are the clean hip scans of patients with metallic hip replacements. As previously discussed, these scans do not exist. Therefore, it is necessary to come up with a new method of training CNN models for metal artifacts reduction that does not rely on the existence of clean images at the hip.

3.2 Our Proposed Method

To tackle the ground truth assumption issue, we propose an innovative method to remove the need for clean images at the hip. In a series of CT scans, the scans from the abdomen to the thigh are similar in terms of body shapes, bone shapes and tissues. By taking advantage of the similarities, our idea is to simulate streaking artifacts in the scans near the patient's hip and train a CNN model, which will be introduced in section 3.3, on the sinograms of these images. After the model is trained, it can then be used to reduce artifacts in the scans of the hip. In this situation, the training data, including the input images and the target images, is generated from the scans that are available in clinical

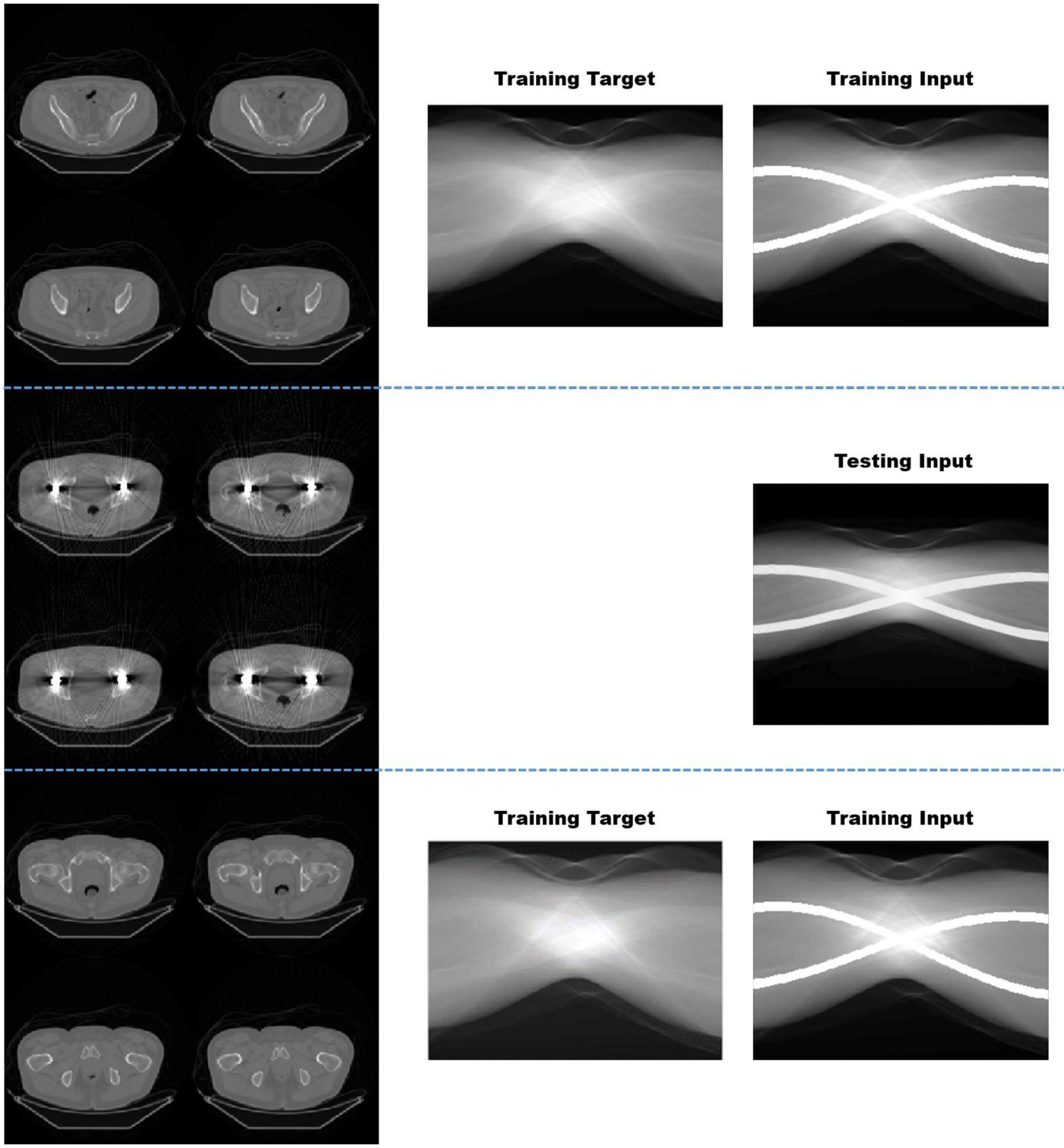


Figure 3.1: The data used for creating training set and testing data set for modeling. Top: the scans of the abdomen – the part above the hip. These images are used as training data. Middle: the scans of the hip. These are testing data. Only input images are available. Bottom: the scans of the thigh – the part below the hip. These images are also used for training.

practices. The clean images of the artifact-corrupted scans at the hip are not required for model training any more.

In a whole-body CT screening, the contrast and brightness of all slices for a patient are usually similar since radiation doses[1] are identical in one screening. Also, the shapes of the parts near the hip, such as abdomen and thigh, are similar to the ones at the hip. Most importantly, the pixel value distributions of the images from the abdomen to the thigh are similar as well. Additionally, radon transform transfers the similarity of the pixel distributions to sinograms, so that the sinograms of the images with similar backgrounds also have similar pixel distributions. Therefore, we think that a model can learn artifacts reduction from the near hip scans, which have similar backgrounds to hip scans.

A natural approach to metal artifacts reduction is to remove the streaks directly from artifact-corrupted images. However, this is difficult since the locations of streaks are hard to acquire. Instead, we will correct the errors caused by the metal pieces when generating the sinograms. Specifically, the values that need to be changed are the ones in the traces resulting from metallic objects (Figure 2.3). Together with the idea described above, the training data consists of the sinograms of artifact-corrupted near hip scans as inputs and the sinograms of clean near hip scans as targets. Artifact-corrupted near hip scans will be generated from clean near hip scans using artifacts simulation. The testing data consists of the sinograms of artifact-corrupted hip scans as inputs only. As previously discussed, artifact-free hip scans that would serve as target images in the testing set do not exist (Figure 3.1).

To verify our idea, we prepare the training data for model training in the following way (see Figure 3.2). Given a series of artifact-corrupted hip scans, we first segment the metal pieces from these images. This segmentation will be useful for simulating artifacts on the scans near the hip. Since the trained model will be used to reduce artifacts on the CT slices of the hip, we hope the simulated artifacts are as similar as possible to the artifacts on the hip scans. Here, metal segmentation is used to approximate the shapes and locations of the metallic objects. We apply the K-means clustering method, which will be introduced in section 3.5, for metal pieces segmentation. After obtaining the segmented metallic objects at the hip scans, we store the results as masks for the subsequent streak simulation. We can overlay a mask on a scan near the hip and apply radon transform to get a sinogram with bands in light colors. To simulate artifacts, we fill the bands with the largest existing value in a sinogram[2]. We then apply FBP on the trace-filled sinograms to reconstruct CT scans. Due to the modification in sinograms, the reconstructed scans near

[1] Radiation doses: the amount of x-rays.

[2] This step is for the computational simulation of artifacts only.

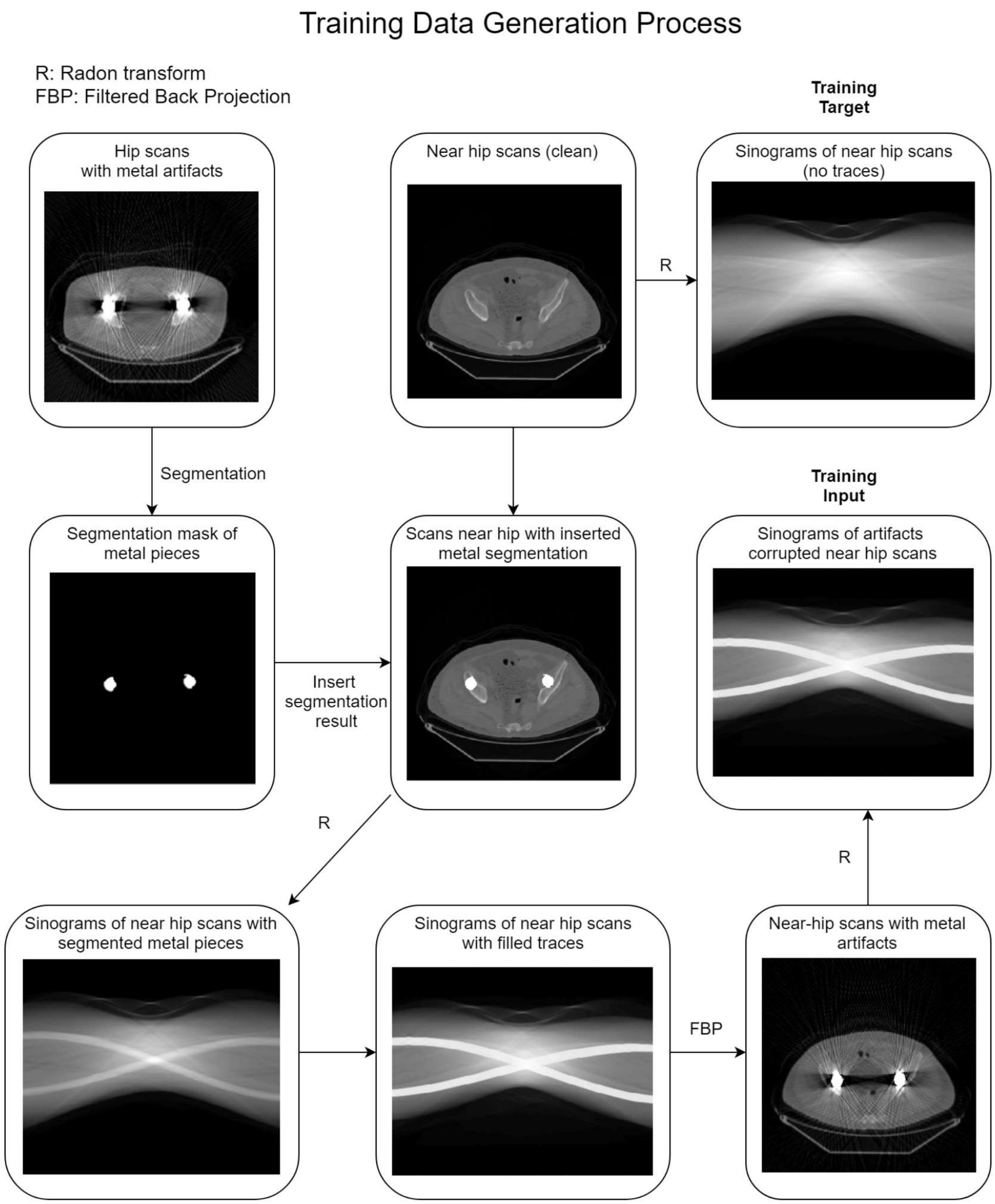


Figure 3.2: Training data generation process. This process can be viewed as a segmentation step and an artifacts simulation process.

the hip now have very bright metal pieces as well as streak artifacts. Lastly, by applying radon transform on the artifact-corrupted near hip scans, we acquire their corresponding sinograms, which are the input data to our model. The targets are the sinograms of the clean images near the hip.

When our model finishes training using the sinograms obtained from the above procedure, it can correct the sinograms by inpainting the bright bands. We use the trained model to correct the sinograms of artifact-corrupted hip scans. The corrected output sinograms will then be used to reconstruct artifact-free CT scans of the hip by FBP.

Model training often requires a large amount of data, especially when the model is complicated and contains many parameters. The number of the abdomen scans and the thigh scans of a patient, which is usually around 20, is insufficient for a model to learn sinogram inpainting. We, therefore, conduct perturbations on the segmentation results regarding the sizes and locations of the metallic objects to obtain additional masks[3]. Then, we randomly choose a near hip scan and overlay one of the perturbed masks on it, to simulate artifacts. Through perturbation, we can acquire sufficient training data and make the model learn to correct a variety of different traces in sinograms.

3.3 Convolutional Neural Networks for Sinogram Inpainting

Because we want to replace the values of traces with correct ones, we can view this process as filling the gaps in sinograms. The values of traces can be seen as missing values. In the literature, the completion of an image with missing parts is called inpainting. Conventional approaches for sinogram inpainting are often based on partial differential equation (PDE). For instance, the Euler's Elastica inpainting technique corrects the values of traces in sinograms and preserves sharp edges and curvature at the mean time[35, 14]. Total variation is deployed by Duan et al. for sinogram inpainting to suppress artifacts[10]. The PDE-based inpainting methods can reduce most of the metal artifacts but are complicated and time-consuming. It is more desirable to have an efficient approach to solve the artifacts reduction problem.

In recent years, neural network models have been developed for image restoration, and many of them are modified from the models used for image classification and image

[3] The number of masks depends on the model size and architecture. In our case, 1000 masks are sufficient for model training.

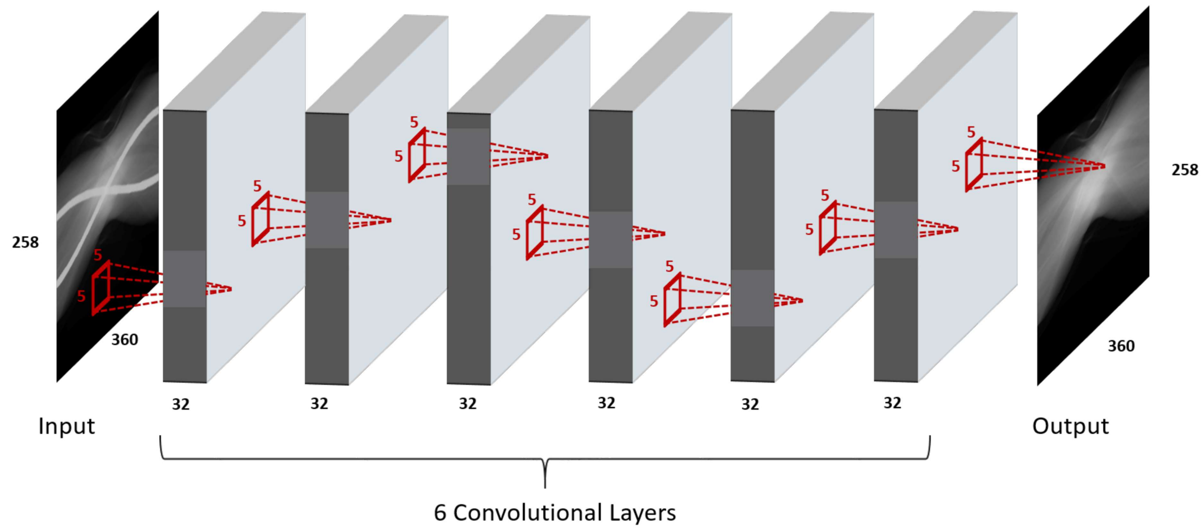


Figure 3.3: Model architectures.

segmentation[24]. Since image data contains spatial structures and information, these models usually adopt convolutional layers to learn patterns in images. The training data generated by Zhang and Yu[36] is the combination of pre-corrected images obtained from the interpolation method and the artifact-corrupted images. Due to the insufficiency of clinical samples, images are separated into small grids which are used as the inputs to their CNN model. Gjesteby et.al[12] developed a CNN model with a special loss function and validated its performance on phantom images. Xie et.al[32] used an improved GoogleNet for residual learning to suppress artifacts. The model was also only applied on phantom images. These three metal artifacts reduction models all take artifact-corrupted images as inputs and clean images as targets. Ghani and Karl corrected the values of the traces in sinograms using a neural network model with convolutional layers only[11]. The sinograms with inpainted traces are then used to reconstruct artifact-free images by FBP. In general, these models performed well in reducing metal artifacts in either phantom images or clinical images and produced better results than conventional approaches.

Training a CNN model could take hours to days. But using a trained model to reduce artifacts could be done less than one second. In this way, using neural network models for metal artifacts suppression is more efficient than the traditional methods.

The model we use in our approach is a CNN model (Figure 3.3), inspired by the model developed by Ghani and Karl[11]. Only convolutional layers are used in the network. The kernels used in convolutional layers capture the spatial information in sinograms to correct the values of the bright traces. The kernel is of size 5×5 and is moved 1 pixel at a time. We pad zeros along margins so that the input image size is the same as the output image size. Each convolutional layer is followed by an activation function – Leaky Rectifier Linear Unit (Leaky ReLU) with slope 0.2. Additionally, batch normalization is performed after activation. The last convolution layer is just the output layer and does not have activation or batch normalization. Ghani and Karl's model has 10 intermediate convolutional layers. We reduce the number to 6 for better computational efficiency and find no obvious degradation in model performance.

3.4 Single Patient & Multiple Patients Scenarios

Based on our proposed method, we design two model training processes for metal artifacts reduction in clinical applications. The first process trains models using the scans of a single patient and will be used on the hip scans of the same patient only. The other process trains model on the hip scans of multiple patients and is thus more robust. The trained model can be applied on the patients it trained on and potentially also applied on future patients.

For the first process, we observe that consecutive slices from one CT screening are consistent in body shapes and similar in brightness, contrast and pixel value distributions. Their corresponding sinograms also share a large amount of resemblance. When a model is being trained on these sinograms, it does not need to adapt to background variations and thus, it can focus on inpainting traces. However, the generalizability and transferability of the trained model will be relatively low. The model may work effectively for the patient used for training, but may produce poor artifacts suppression results for other patients. Therefore, training a model on every single patient's sinograms is inefficient.

To resolve this concern, we consider developing a model that can be applied to multiple patients who have various physical circumstances and characteristics. We initially deployed our model directly on the sinograms obtained from multiple patients' scans. Due to the diversity in body shapes and image properties, the input sinograms from different patients have more variances in pixel value distributions than the sinograms from a single patient. Unfortunately, our model trained on multiple patients fails to adapt to the dissimilarities in the inputs and produces sinograms that output blurred reconstructed CT images. Consequently, we realize we need to use normalization[4] techniques to make the inputs obtained from multiple patients similar.

[4]The "normalization" here is different from the one in batch normalization (section 2.2.3).

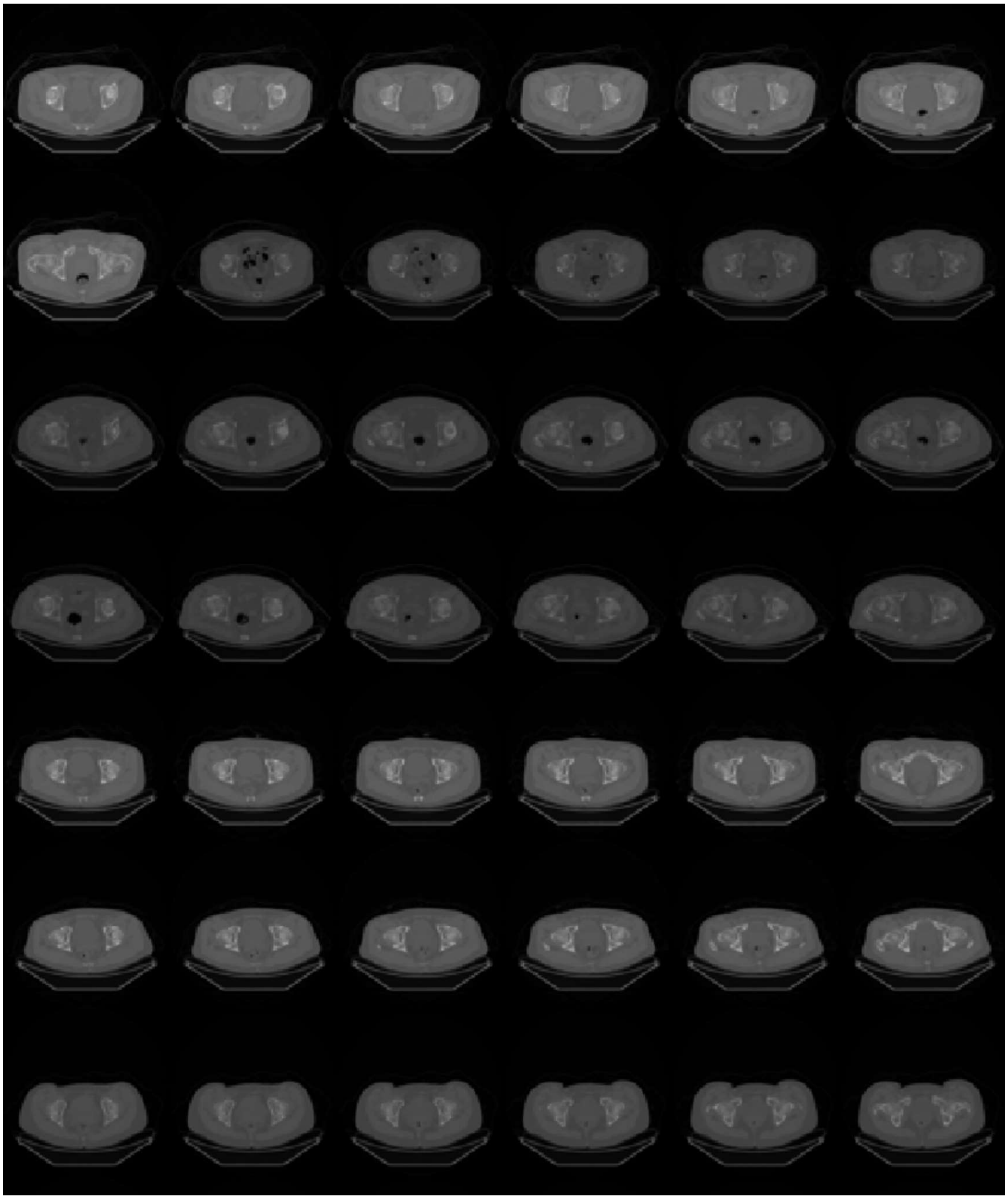


Figure 3.4: The scans of the hip of different patients. The images have different brightness, contrast and body shapes.

Through observing the CT scans of different patients, we notice the inconsistencies in body shapes as well as the brightness and contrast of the scans (Figure 3.4). Since it is inappropriate for us to alter any physical property or condition of a patient, we ignore the body shape inconsistencies during input normalization. Between brightness and contrast, we observe that brightness is the main cause of the variance in pixel value distributions of the scans and the corresponding sinograms. Thus, we focus on normalizing the brightness.

First, we randomly choose a patient and use one of his/her near hip scans as the standard[5]. Then, we want to make the near hip scans of other patients similar to the standard.

The first method we try is histogram matching, since it is a direct way of matching pixel distributions. The resulting scans from histogram matching are able to achieve consistent brightness. However, we also need to adjust the brightness of hip scans back to original levels after artifacts are removed. The reversion is aimed to keep the consistency of brightness in the scans of a patient. Therefore, the normalization method we use should be easily reversible. Unfortunately, we fail to find a way to reverse histogram matching.

We next consider a much simpler normalization method, scaling, which can be reversed by dividing the scaling factor. Given an image S as the standard and an image I to be scaled, we find a scalar a such that the value distribution of aI is similar to the value distribution of S. We use the same scaling factor for all the scans of a single patient and different scaling factors for scans of different patients. After scaling the near hip scans of different patients and applying radon transform, we can acquire the sinograms with similar distributions. By applying FBP on the output sinograms from the model, we obtain artifact-free images with similar distributions. We then divide the images by their corresponding scaling factors to get the final images without artifacts, but with background pixel value distributions similar to the original scans.

3.5 K-means Clustering for Metal Segmentation

As we introduced in section 3.2, we need to simulate metal artifacts in the CT scans of the parts near the hip so that we can create the input data to our model. We will use the shape and location of metal artifacts extracted from corrupted scans at the hip as a basis for artifact simulating. Thus, we need to segment the metal artifacts in corrupted hip scans so that the artifacts' approximate locations and shapes can be obtained.

[5]Whether the scans are bright or dim does not matter. The goal of normalization is to have similar brightness among all patients' scans.

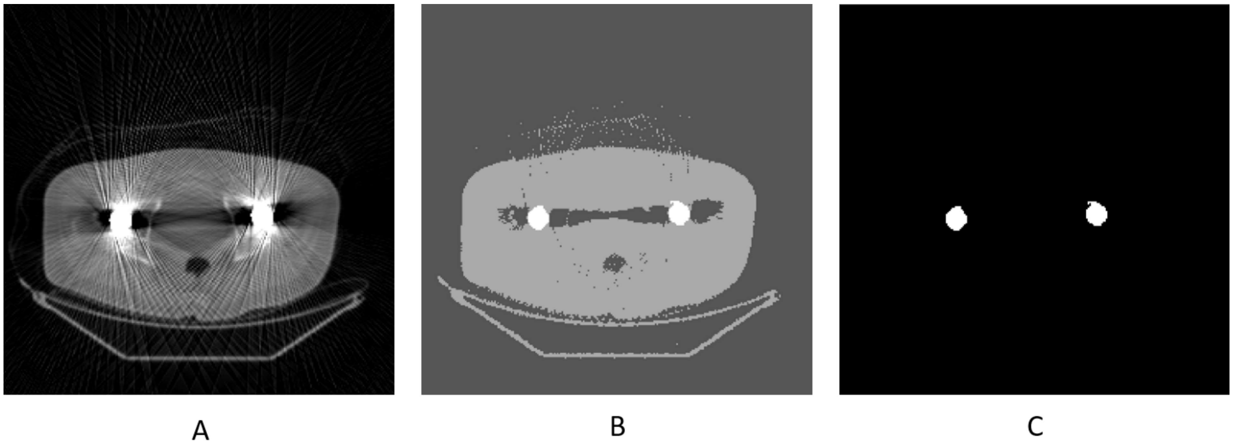


Figure 3.5: Metal segmentation using K-means clustering. Segmenting (A) using $k = 3$ gives (B). The segmentation mask (C) is generated by merging the black and grey segments to one black segment. So the mask is a binary image with metal pieces segmentation result being white and the rest being black.

As objects with higher density appear brighter in images and have higher pixel values, metal artifacts appear very bright and attain high values. In some simple cases, separating metal parts from the background can be easily done by thresholding. However, most clinical images have complicated backgrounds. It is not easy to predetermine thresholds to perform metal segmentation for hip CT scans.

In order to be more robust, we apply the K-means clustering method to segment metallic objects from artifact-corrupted images. Consider a greyscale image that has N pixels. We first sort these N pixels by their values in increasing order, where lower values appear darker and higher values appear brighter in images. Then, we randomly choose k values without replacement from the given values, and assign them as the k center points, $c_1, c_2, ..., c_k$[6]. In the first iteration, we calculate the difference between each pixel value to each center point. We label a pixel as in class i, where $1 \leq i \leq k$, if the difference of the pixel value and c_i is the smallest. After the first iteration, each pixel has a label that indicates which class the pixel belongs to. For each class i, we compute the mean pixel value and assign the computed mean to c_i. This ends the first iteration with new center points. Iterations will be repeated until two consecutive iterations give identical center points. After the final iteration, we choose the class with the highest center point value to represent the metal artifacts.

[6]We have observed that $k \geq 3$ gives stabler segmentation results. In our experiments, we use $k = 3$.

Chapter 4

Experiment and Result

In this chapter, we perform 4 experiments to validate our idea and approach. The data used in our experiments include phantom images and clinical images. The metal artifacts reduction results will be displayed in image form and evaluated using quantitative measurements.

4.1 Model Training and Data source

The model we introduce in Section 3.3 is used for all the experiments outlined in this chapter. We also apply identical settings for training in all experiments. We deploy ADAM optimizer with learning rate 5e-3 and decay 2.5e-5. Due to computational resources limitation and the large image size, we use a batch 16 sinograms when training our model. Training time varies with different inputs, ranging from 1 to 2 hours. The loss function is mean squared error, measuring the Euclidean distance between outputs and targets.

The data used in Experiment 1 are artificial phantom images. The data used in the rest of the experiments are obtained from Grossberg et al.[13]. Due to memory limitations, we resize the images from 512×512 to 256×256. Additionally, all the images in the dataset, including hip scans, are metal-free and artifact-free[1]. Therefore, we need to simulate artifacts on hip scans and then validate our proposed method. We will use the clean hip images as reference images to perform both qualitative and quantitative evaluations.

[1]Unfortunately, we are unable to find clinical whole-body CT scans that have artifacts at the hip part, i.e. real world scans of the patients who have hip replacements. Such scans can make a huge contribution to this research in the future.

4.2 Performance Evaluation Metrics for Metal Artifacts Reduction

To evaluate the metal artifacts reduction outcome, we conduct qualitative and quantitative analysis on the reconstructed scans using the outputs of our model. The evaluation will be conducted only on the test data in each experiment.

Qualitative evaluations are performed by showing the artifact-corrupted images, the artifact-reduced images and the reference images.

Quantitative evaluations are conducted using the following measurement metrics:

- MSE: Mean Squared Error, the average squared difference between pixels. Smaller values imply smaller discrepancies between the artifact-reduced image and the reference image.
- PSNR: Peak Signal-to-Noise Ratio. Larger values indicate better reconstruction from the artifact-corrupted image.
- SSIM: Structural Similarity, ranged from 0 to 1. Higher values suggest higher similarity between the artifact-reduced image and the reference image.

4.3 Experiment 1: phantom images

Before applying our method on clinical images, we first test the model performance on phantom images to have an idea of the robustness and the performance of our model.

4.3.1 Experiment Setup

Each phantom image is in greyscale $0 - 1$ and of size 128×128. In all the phantom images we generate, we use pure black color with greyscale value 0 as background since air usually appears very dark in CT scans. Then, we insert a big white ring, with greyscale value 0.9, to represent the outer body skeleton. We also insert 3 to 10 different shapes in random levels of grey between 0.1 to 0.8 at random locations within the outer white ring, to simulate various tissues and organs. Shapes could vary from circles, ellipses, squares, and rectangles. Similarly, 1 to 4 random shapes with greyscale value of 1 are inserted at various locations to represent simulated metallic objects.

	Input	Target
Training	1000 sinograms (have bright traces) of phantom images with metal artifacts	1000 sinograms (no bright traces) of phantom images without metal and artifacts
Testing	100 sinograms (have bright traces) of phantoms images with metal artifacts	100 sinograms (no bright traces) of phantom images without metal and artifacts

Table 4.1: Training and testing data in Experiment 1

The artifact simulation procedure is similar to the one in Figure 3.2. The only difference is in the method used to obtain the masks of metal pieces. In this experiment, since all metallic objects are simulated, we can acquire the masks when we insert the simulated metals into the images. Two examples of the generated phantom images can be seen in Figure 4.1.

Most of the objects in this experiment are randomly generated. Thus, we can create as many phantom images as we want. Considering the size of our model and the number of parameters needed, we generate 1100 phantom images, 1000 for training and 100 for testing (Table 4.1). The partition between the training set and the testing set is completely random. The target images for training and testing are the sinograms of the phantom images without simulated metallic objects. These sinograms have no bright traces. After the model finishes learning from the training data set, it can be used to make predictions on the testing set. The predicted results, which are the corrected sinograms, will then be used to reconstruct images using FBP to give the final artifact reduction results.

4.3.2 Result

The qualitative results in Figure 4.1 and the quantitative results in Table 4.2 both indicate reconstructed images have significant improvement in image quality compared to artifact-corrupted images. Most of the artifacts are successfully suppressed by correcting the values in traces in sinograms using our trained model, and they can be barely seen in the reconstructed images. Even though the input images have a variety of shapes and shades in the background and simulated metal pieces, our model is capable to handle the variance and provide stable prediction results on par with the performance on the training set.

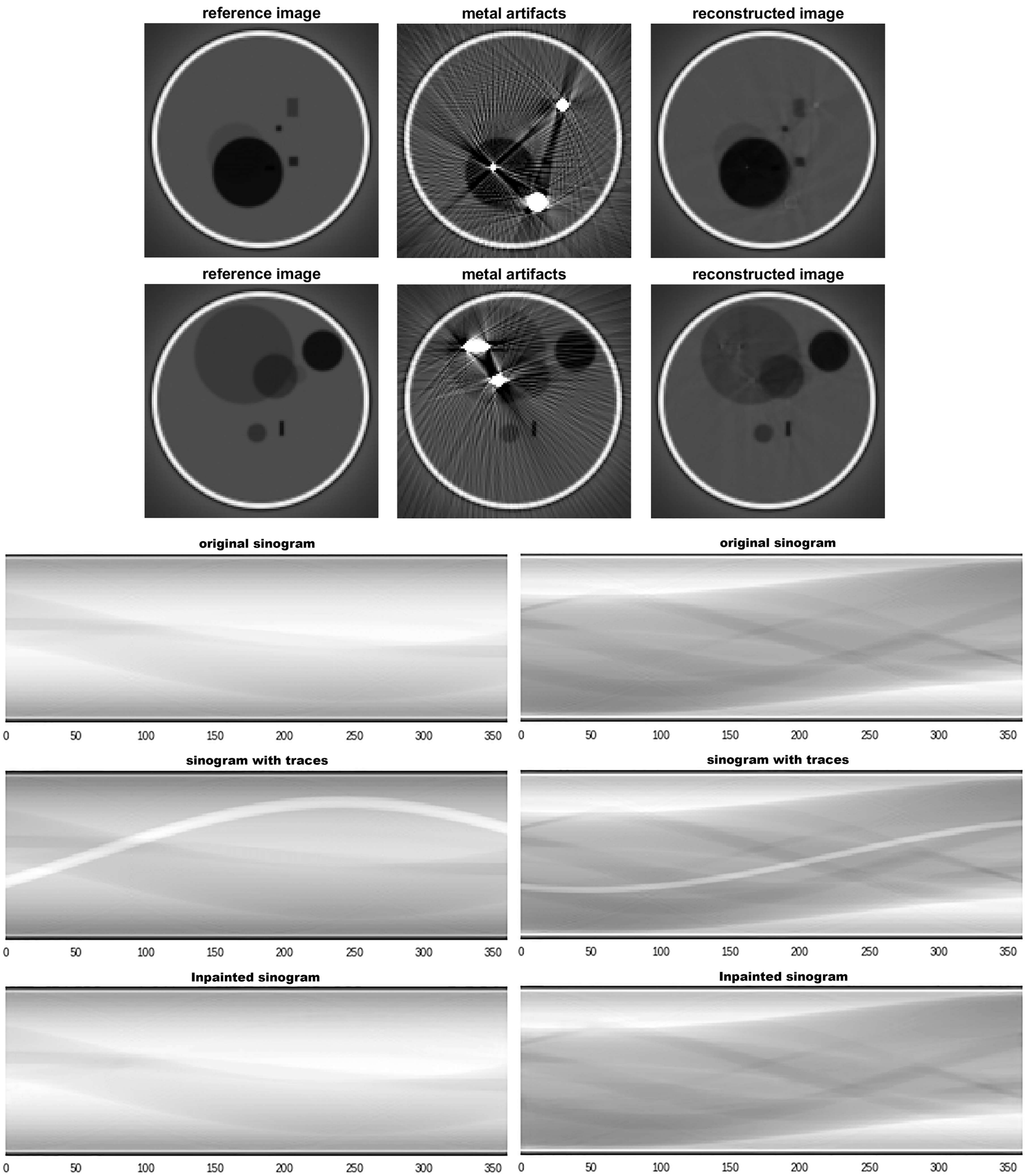


Figure 4.1: Experiment 1: phantom images. Above: Two examples of phantom images with simulated artifacts and the corresponding metal artifact reduction results. Below: the corresponding targets, inputs and outputs in our model.

Comparing with the experiment performed by Ghani and Karl[11], our input data has much more variance stemming from a larger variety of background shapes and simulated metal pieces. The phantom images they used in their experiment all have the same background and their simulated metallic objects are filled circles only. Our experiment demonstrates the robustness of the model and shows its capability for reducing artifacts even given different image backgrounds.

Images	MSE	SSIM	PSNR
Corrected Image	7.727e-5	43.289	0.9817
Uncorrected Image	7.721e-3	27.221	0.5583

Table 4.2: Quantitative Comparison Results of Phantom Images

4.4 Experiment 2: hip scans for training and testing (one patient)

In the second experiment, we want to see if our model could perform well on real CT hip scans, which have more variances in object shapes and brightness than phantom images. The tissues and organs in our body have complicated shapes and curves. Additionally, the attenuation coefficients vary in different parts of an organ or a tissue, while the attenuation coefficients are consistently identical in generated shapes in phantom images. In this experiment, we further test our model's capabilities, including robustness and adaptability, on more complicated inputs.

4.4.1 Experiment Setup

Since no patients in the data set have existing metal implants and therefore no CT slices contain streaking artifacts, we need to simulate streaking artifacts to generate inputs for both the training set and the test set. A patient with 250 slices of hip CT scans is singled out from the data set, and 11 consecutive slices from the scans are chosen. Using these 11 images, we identify the sizes and the locations of hip bones. Then, we use these information to generate 1100 masks, by perturbing the location and size of the metal pieces. The rest of the artifacts simulation process is similar to the one introduced in Figure 3.2. After obtaining 1100 corrupted hip scans, we randomly partition them into a training set and

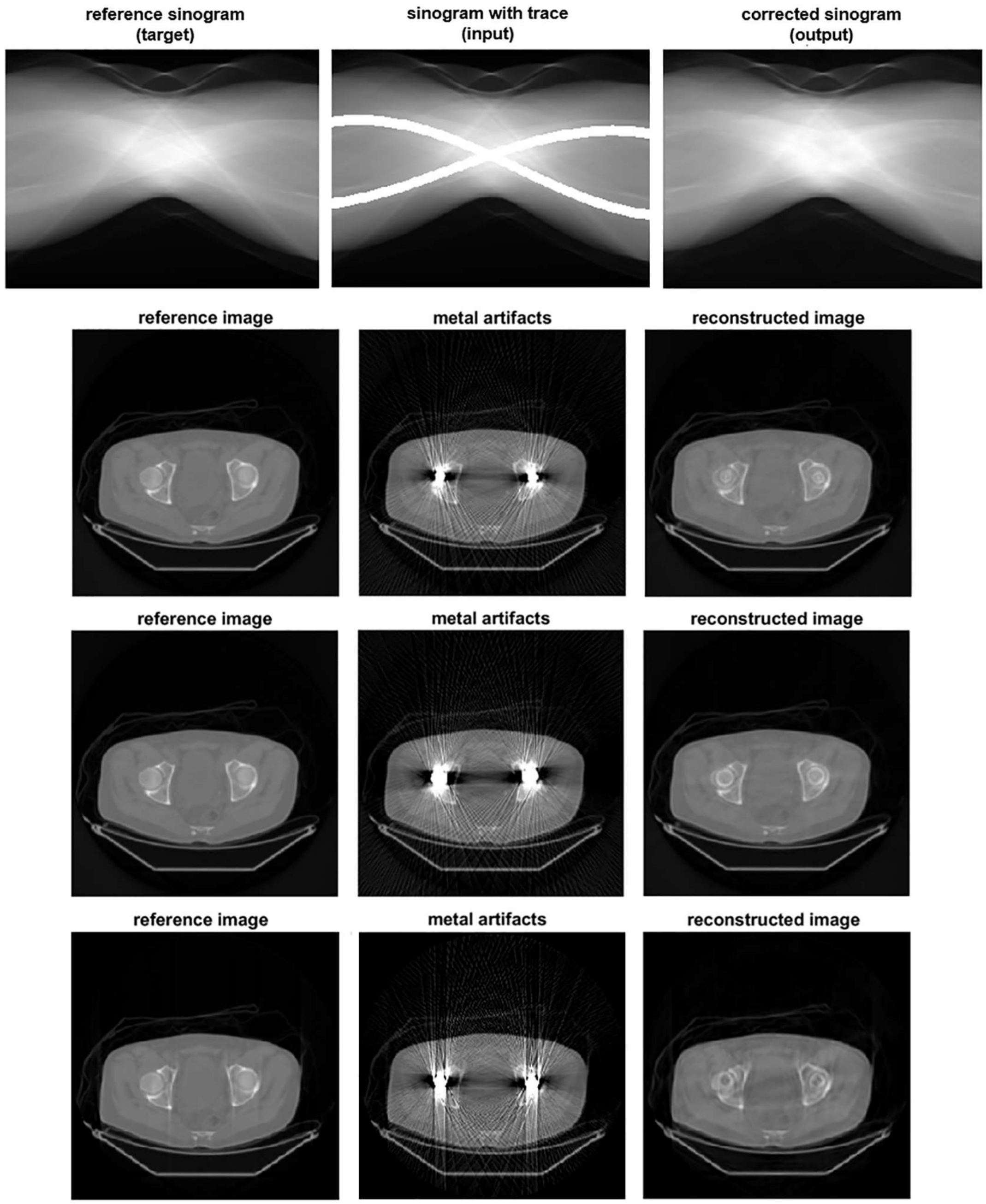


Figure 4.2: Experiment 2: hip scans for training and testing. Above: an example of the target, the input and the output in our model. Below: three testing examples to illustrate metal artifacts reduction result.

	Input	Target
Training	1000 sinograms (have bright traces) of hip scans with metal artifacts	1000 sinograms (no bright traces) of hip scans without metal and artifacts
Testing	100 sinograms (have bright traces) of hip scans with metal artifacts	100 sinograms (no bright traces) of hip scans without metal and artifacts

Table 4.3: Training and testing data in Experiment 2

a test set with 1000 and 100 images respectively. By applying radon transform again, we acquire the inputs for both data sets (Table 4.3). The targets are correspondingly the sinograms of the original clean slices of the parts above and below the hip.

4.4.2 Result

As shown in Figure 4.2 and Table 4.4, in this experiment where training and testing data both obtained from real hip scans, our model does an excellent job in reducing streak artifacts. Both the quantitative measurement and the qualitative analysis indicate the image quality of the reconstructed images is significantly improved compared to the artifact-corrupted images. The parts, which used to be covered by streaks and artifacts, is visible after correction. Although the dark streaks in the artifact corrupted images resulted in slightly darker shades in the reconstructed image, we can still see the parts beneath the shades fairly clearly. This experiment reveals the adaptability of the model to clinical data, which contains not only complicated shapes and curves but also diverse attenuation coefficients in different parts of body matter.

Images	MSE	SSIM	PSNR
Corrected Image	9.180e-5	40.523	0.9752
Uncorrected Image	6.623e-2	20.793	0.3514

Table 4.4: Quantitative Comparison Results of hip scans for training and testing.

4.5 Experiment 3: training on near hip scans, testing on hip scans (single patient)

In the previous two experiments, we assume that the clean images of the hip are available and we use them as the target for both model training and testing. However, as explained in section 3.1, it is impossible to obtain the artifact-free hip scans if the patient has metal implants at the hip. Since we have clean slices at the parts other than hip, we can simulate artifacts on these slices, and let the model learn how to reduce artifacts by inpainting traces in the sinograms of these slices. Once the model learns the pattern, it can be applied to the artifact-corrupted slices of the hip to suppress streaks.

In this experiment, we validate our idea by training our model on the sinograms generated from the near hip scans and testing the trained model on the sinograms of artifact-corrupted hip scans. All images in the training set and testing set come from a single patient.

4.5.1 Experiment Setup

The patient and artifacts simulation procedure on hips scans are identical to the ones described in Experiment 2. We generate 1000 hip scans with artifacts, then perform K-means segmentation with $k = 3$ to extract the masks of metallic objects. For generating training data, we pick 22 near hip scans, 11 consecutive slices above the hip and 11 consecutive slices below the hip from the same patient. We overlay each of the 1000 masks on a random near hip scan and then simulate artifacts as described in Figure 3.2. From this we acquire 1000 near hip scans with various artifacts. We use the corrupted sinograms of these 1000 scans as training input and the corresponding clean sinograms as training targets. The 11 slices with manually inserted circles are used to make 11 artifact corrupted images, which are then used as the testing set. In model prediction, the 11 sinograms of the hip scans with artifacts will be used as inputs (Table 4.5).

For experimental purposes only, we compare the reconstruction results of the 11 hip scans with the clean ones which are obtained from the data source directly. In clinical practices, we do not have the clean images of the hip to compare with. The only available comparison, in reality, is between artifact-corrupted scans and artifact-reduced scans.

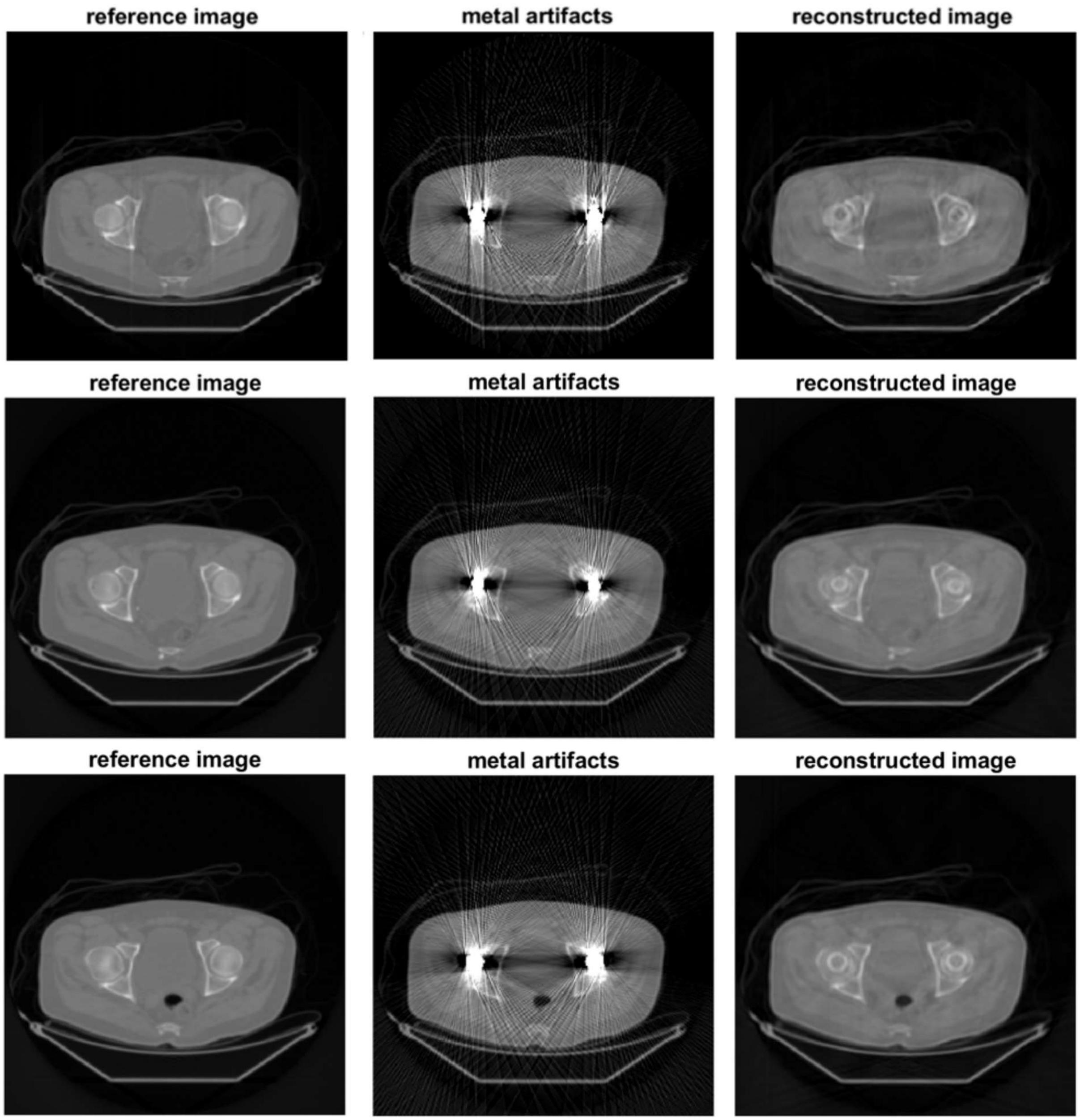


Figure 4.3: Experiment 3: training on near hip scans and testing on hip scans from a single patient. Three testing examples to illustrate metal artifacts reduction result in the scans of hip.

	Input	Target
Training	1000 sinograms (have bright traces) of near hip scans with metal artifacts obtained from one patient	1000 sinograms (no bright traces) of near hip scans without metal and artifacts obtained from one patient
Testing	11 sinograms (have bright traces) of hip scans with metal artifacts	11 sinograms (no bright traces) of hip scans without metal and artifacts

Table 4.5: Training and testing data in Experiment 3

4.5.2 Result

As observed from Figure 4.3 and Table 4.6, the artifacts are significantly suppressed in the hip scans transformed by FBP from the corrected sinograms. Unlike the data sets in Experiment 2, the training data and the testing data are generated from different images. Our model successfully learns the way to correct the values in the traces in sinograms regardless of the difference in the backgrounds in the images. Even though the model has never seen the hip sinogram, by learning from the abdomen and the thigh sinograms, the model is able to adapt the knowledge learned and fill the traces in the hip sinograms with appropriate values. This experiment further demonstrates the flexibility of our model for handling discrepancies between the training data and testing data.

Images	MSE	SSIM	PSNR
Corrected Image	3.564e-4	40.990	0.9473
Uncorrected Image	6.623e-2	20.793	0.3514

Table 4.6: Quantitative Comparison Results of Experiment 3: training on near hip scans and testing on hip scans from a single patient.

4.6 Experiment 4: training on near hip scans, test on hip scans (multiple patients)

Similar to the experiments in many other studies about metal artifacts reduction, Experiment 2 and 3 are carried out on the scans of a single patient. As explained in section 3.4,

	Input	Target
Training	1000 sinograms (have bright traces) of near hip scans with metal artifacts obtained from 10 patients	1000 sinograms (no bright traces) of near hip scans without metal and artifacts obtained from 10 patients
Testing	102 sinograms (have bright traces) of hip scans with metal artifacts	102 sinograms (no bright traces) of hip scans without metal and artifacts

Table 4.7: Training and testing data in Experiment 4

building one model per patient is inefficient in practice. A trained model might work well for that specific patient but might fail to produce adequate artifact suppression results for the CT scans of other patients. We hope to generalize our approach so that it can used on multiple patients with various physical circumstances and characteristics.

4.6.1 Experiment Setup

In this experiment, we generate our training data set and testing data set using the scans from 10 different patients. As illustrated in section 3.4, the scans of different patients have variances in brightness, contrast and body shapes. Among these variants, brightness leads to the greatest variance in pixel values. To reduce the variance in pixel values in scans and sinograms, we need to normalize the data regarding brightness before generating the data set for modeling. Normalization is carried out using scaling functions, which make the pixel value distributions of all scans similar. After the above data preprocessing, we generate 1000 corrupted images from the near hip slices from the 10 patients' scans for training using the same process in Experiment 3. The testing data in this experiment is 102 artifact-corrupted hip scans obtained from the 10 patients (Table 4.7). Pixel value distribution normalization is conducted on these images before the input sinograms are generated. The reconstructed results of hip scans will then be re-scaled using the inverses of the scaling functions, and compared with the clean images in the same way as in Experiment 3.

4.6.2 Result

As shown in Figure 4.4, without brightness adjustment, our model fails to produce clear reconstructed artifact-free images. The reconstructed images appear blurry and cloudy,

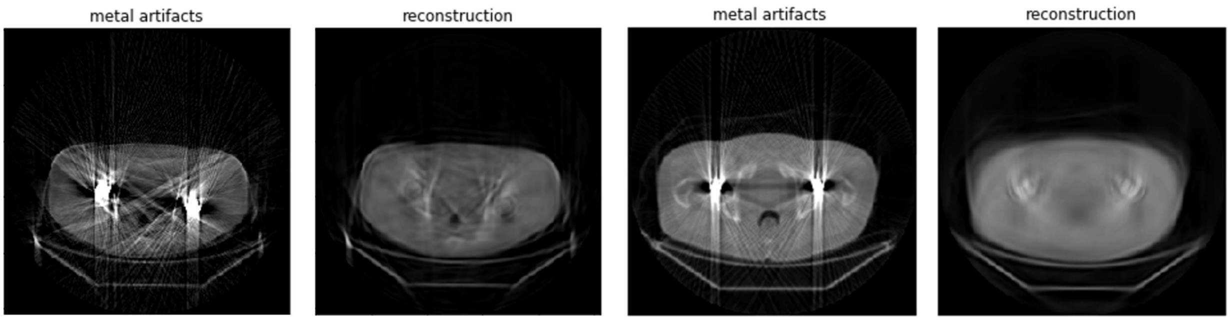


Figure 4.4: Two failed cases in Experiment 4 without brightness adjustment.

containing insufficient structural details at the hip. Due to the difference among the pixel value distributions of the corrupted images, the values in the corresponding sinograms also have dissimilar distributions. The model is not robust enough to perform appropriate adjustments regarding the discrepancy in value distributions. Therefore, it is essential to preprocess the images from different patients so that the input sinograms possess similar value distributions.

We obtain the results shown in Figure 4.5 and Table 4.8 by normalizing the pixel value distributions as a preprocessing step. We see a greater reduction of streak artifacts when comparing the reference images and the reconstructed ones. With the normalization of pixel value distributions using scaling functions, our approach can adapt to the variances in body shapes and physical conditions at the hip, and be applied on multiple patients.

Images	MSE	SSIM	PSNR
Corrected Image	2.324e-4	36.915	0.9094
Uncorrected Image	6.271e-2	21.438	0.4329

Table 4.8: Quantitative Comparison Results of Experiment 4: training on near hip scans and testing on hip scans of multiple patients.

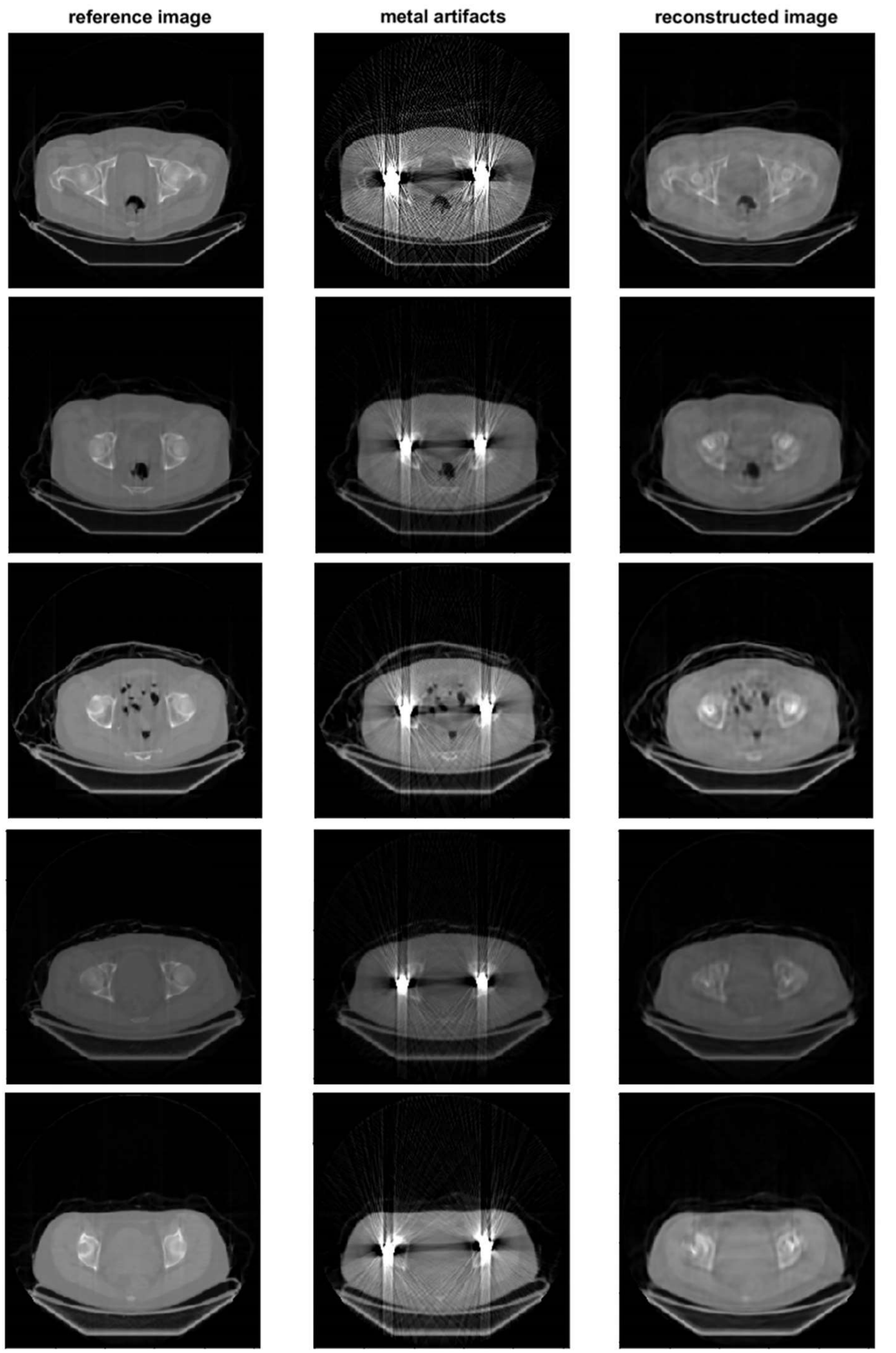


Figure 4.5: Experiment 4: training on near hip scans and testing on hip scans of multiple patients. Five testing examples to illustrate metal artifacts reduction results in hip scans.

Chapter 5

Conclusion and Future Work

In this paper, we propose an innovative approach to address the issue of metal artifacts in CT scans. In particular, we focus on removing artifacts at the hip, by training a neural network on the data generated from the scans near the hip. Our method eliminates the need for ground truth images at the hip and produces reconstructed hip scans without metal artifacts while preserving the details in the background. In our experiments, we test our approach on the scans obtained from a single patient as well as the scans obtained from multiple patients. Our approach is further generalized compared to the existing methods in the literature.

However, there is still room for future improvements in our method. One potential step could be acquiring a series of whole-body CT scans with artifacts at the hip and using the CT machine for artifacts simulation. Because the data used in our experiments are all clean images, the artifacts in the hip scans are simulated computationally. These generated artifacts might attain different properties and characteristics from the artifacts formed in the CT machine. When deploying our approach in clinical practices, we should to train our model using the scans with real artifacts. It might be beneficial to simulate artifacts on near hip scans in alternative ways, such as using the CT machine and phantom objects, in order to attain streaks with similar properties to the streaks formed in CT scanners. Another direction of potential future work could be improving the architecture of the neural network model. In the scenario of multiple patients, without brightness adjustment, our model fails to adapt to the variance in the brightness of images. It may be possible to modify the model architecture and improve the robustness against different pixel value distributions.

References

[1] 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.

[2] E Barmeir, B Dubowitz, and M Roffman. Computed tomography in the assessment and planning of complicated total hip replacement. *Acta Orthopaedica Scandinavica*, 53(4):597–604, 1982.

[3] F Edward Boas and Dominik Fleischmann. Evaluation of two iterative techniques for reducing metal artifacts in computed tomography. *Radiology*, 259(3):894–902, 2011.

[4] F Edward Boas and Dominik Fleischmann. Ct artifacts: causes and reduction techniques. *Imaging Med*, 4(2):229–240, 2012.

[5] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[6] Yang Chen, Yinsheng Li, Hong Guo, Yining Hu, Limin Luo, Xindao Yin, Jianping Gu, and Christine Toumoulin. Ct metal artifact reduction method based on improved image segmentation and sinogram in-painting. *Mathematical Problems in Engineering*, 2012, 2012.

[7] Yunjin Chen, Wei Yu, and Thomas Pock. On learning optimized reaction diffusion processes for effective image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5261–5269, 2015.

[8] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[9] Wikimedia Commons. File:pelvis ct 3d front 01.jpg — wikimedia commons, the free media repository, 2017. [Online; accessed 5-December-2019].

[10] Xinhui Duan, Li Zhang, Yongshun Xiao, Jianping Cheng, Zhiqiang Chen, and Yuxiang Xing. Metal artifact reduction in ct images by sinogram tv inpainting. In *2008 IEEE Nuclear Science Symposium Conference Record*, pages 4175–4177. IEEE, 2008.

[11] Muhammad Usman Ghani and W Clem Karl. Deep learning based sinogram correction for metal artifact reduction. *Electronic Imaging*, 2018(15):472–1, 2018.

[12] Lars Gjesteby, Hongming Shan, Qingsong Yang, Yan Xi, Bernhard Claus, Yannan Jin, Bruno De Man, and Ge Wang. Deep neural network for ct metal artifact reduction with a perceptual loss function. In *In Proceedings of The Fifth International Conference on Image Formation in X-ray Computed Tomography*, 2018.

[13] A Grossberg, A Mohamed, H Elhalawani, W Bennett, K Smith, T Nolan, S Chamchod, M Kantor, T Browne, K Hutcheson, G Gunn, A Garden, S Frank, D Rosenthal, J Freymann, and C Fuller. Data from head and neck cancer ct atlas. The Cancer Imaging Archive, 2017.

[14] Jianwei Gu, Li Zhang, Guoqiang Yu, Yuxiang Xing, and Zhiqiang Chen. X-ray ct metal artifacts reduction through curvature based sinogram inpainting. *Journal of X-ray Science and Technology*, 14(2):73–82, 2006.

[15] Siemens Healthcare. Ct history and technology.

[16] Alan L Hodgkin and Andrew F Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500–544, 1952.

[17] Yale Internal Medicine. Cardiac ct.

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

[19] Richard A Ketcham and Romy D Hanna. Beam hardening correction for x-ray computed tomography of heterogeneous natural materials. *Computers & geosciences*, 67:49–61, 2014.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[22] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[23] Matthew Lukies. Normal pelvis and both hips: Radiology case.

[24] Xiaojiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Advances in neural information processing systems*, pages 2802–2810, 2016.

[25] Marlene H McKetty. The aapm/rsna physics tutorial for residents. x-ray attenuation. *Radiographics*, 18(1):151–163, 1998.

[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] ReVision Radiology. Revision radiology: Ct metal artifact reduction using the metal deletion technique (mdt).

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[29] Harpal Singh Selhi. *Central Hip Fracture Dislocation*. May 2002.

[30] Sarabjeet Singh, Mannudeep K Kalra, Jiang Hsieh, Paul E Licato, Synho Do, Homer H Pien, and Michael A Blake. Abdominal ct: comparison of adaptive statistical iterative and filtered back projection reconstruction techniques. *Radiology*, 257(2):373–383, 2010.

[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] Junyuan Xie, Linli Xu, and Enhong Chen. Image denoising and inpainting with deep neural networks. In *Advances in neural information processing systems*, pages 341–349, 2012.

[33] Shiyu Xu and Hao Dang. Deep residual learning enabled metal artifact reduction in ct. In *Medical Imaging 2018: Physics of Medical Imaging*, volume 10573, page 105733O. International Society for Optics and Photonics, 2018.

[34] Hui Xue, Li Zhang, Yongshun Xiao, Zhiqiang Chen, and Yuxiang Xing. Metal artifact reduction in dual energy ct by sinogram segmentation based on active contour model and tv inpainting. In *2009 IEEE Nuclear Science Symposium Conference Record (NSS/MIC)*, pages 904–908. IEEE, 2009.

[35] Xiaochen Zhang and Justin WL Wan. Image restoration of medical images with streaking artifacts by euler's elastica inpainting. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 235–239. IEEE, 2017.

[36] Yanbo Zhang and Hengyong Yu. Convolutional neural network based metal artifact reduction in x-ray computed tomography. *IEEE transactions on medical imaging*, 37(6):1370–1381, 2018.