Universiteit Leiden
The Netherlands

Master Computer Science

Virtual Reconstruction Analysis: Ring Artifact and Noise Removal in a breast CT geometry

Name:	Guillem Casals Ruesga
Student ID:	s3741486
Date:	25/07/2025
Specialization:	Artificial Intelligence
1st supervisor:	Dr. Joost Batenburg
2nd supervisor:	Dr. Daan Pelt

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Acknowledgements

I would like to express my gratitude to all the people that have helped me during this project. First, I would like to thank my supervisors, Dr. Joost Batenburg and Dr. Daan Pelt, for the opportunity to work on this thesis and the guidance and insight given during the process.
I would also like to thank to Dr. Berend C. Stoel and Dr. Martin N. J. M. Wasser at the Leids Universitair Medisch Centrum for the valuable insight and information they gave for this project.
Furthermore, I would like to give a special thanks to Dr. Veikko Ruth from the AB-CT team. Your understanding of the scanner and valuable data allowed me to continue and further improve the results of this thesis.
Lastly, many thanks to Alice O'Donnell for her encouragement, support, and meticulous proofreading and feedback that improved the clarity and coherence of this thesis.

Abstract

Early detection of breast cancer significantly improves patient outcomes. AB-CT aims to improve early detection through their CT scanner machine, the nu:view. Unfortunately, high-resolution breast CT imaging often suffers from reconstruction artifacts and noise that can obscure critical details. This thesis investigates the origin of the ring artifacts which appear in the AB-CT's nu:view machine and removes the artifacts and noise. This thesis proposes two solutions to remove the artifacts and one solution to remove the noise. To achieve this, a simulated breast CT geometry was made to mimic the scanner behavior. We implemented two complementary solutions to remove artifacts: (1) a lightweight pre-processing pipeline applying targeted stripe filters to eliminate the artifacts, and (2) a multi-detector reconstruction made via the simultaneous iterative reconstruction technique (SIRT), which inherently avoids artifact formation at high resolution when detector alignment errors are corrected. Finally, to remove the noise, we employed the self-supervised Noise2Inverse deep neural network, achieving a PSNR of 21.68dB and an SSIM of 0.912 in respect to the phantom, therefore effectively removing the noise whilst preserving fine detail structures. Together, these approaches enable artifact-free and noise reduced breast CT reconstruction and can be integrated into the nu:view workflow.
GitHub repository: `https://github.com/Will-Ruesga/breast-ct-geometry`

1 Introduction

Breast cancer remains one of the most prevalent health challenges worldwide, and early detection plays a pivotal role in improving patient outcomes. Diagnosing breast cancer as quickly as possible is essential, as early diagnosis can ensure a patient's well-being even without the need for heavy treatments such as chemotherapy. AB-CT, a high-tech medical company specialized in research and development of solutions for diagnostic breast imaging, aims to improve the early diagnosis of breast cancer by developing a new breast CT scanner machine, the nu:view.

This scanner, that uses spiral CT and single photon-counting technology, is already in use in universities and radiology practices around Europe. It offers good resolution despite the low radiation dose and short scan times of up to 12 seconds. The scanning is performed without any breast compression which ensures full comfort for the patient. Every rotation around the breast creates up to 2,000 image projections, the total scan does several rotations, acquiring images of the entire breast in one scan, resulting in a 3D image with soft tissue differentiation and avoiding the use of superimposition, being the first CT scan machine that accomplishes it.

In October 2021, LUMC (Leids Universitair Medisch Centrum), the hospital affiliated with Leiden University, became the first hospital in the Netherlands to use the nu:view machine in clinical routine. In Spring 2024, the LUMC radiology and imaging teams showed us the machine, and we discussed some improvements that could be made to the reconstructed 3D images. Despite its advantages to the nu:view machine, the reconstruction of such a complex images does indeed bring challenges. Generally, generated reconstructions are affected by artifacts, and, of course, by noise, as it is impossible to avoid in any environment. These artifacts and noise may compromise the overall quality of the reconstruction, hence possibly altering a later diagnosis. Along with the professionals at the LUMC, we determined that reducing or removing the artifacts and noise would be the most beneficial course of action to take within the scope of a Master's thesis.

In summation, the LUMC's nu:view breast CT scan machine performs state-of-the-art reconstructions while also maximizing comfort for the patient, resulting in a good quality reconstruction. However, these reconstructions have some artifacts and noise, bringing us to the center-point of this research which is to explain the appearance of these artifacts in the real-world reconstructions.

After some testing and observation of the results produced by the CT scan machine, we determined that the main artifacts that appear in the reconstruction are ring artifacts, which are circular patterns that arise due to inconsistencies in the detector performance or from calibration errors, as well as noise which obscures fine details and diminishes the overall clarity of the reconstructed image.

To determine their origin of the artifacts, we worked with a model of their cause, replicating a similar environment as in the nu:view machine and adjusting different settings to observe and understand how these artifacts came to be.

Furthermore, we wanted to find a solution to diminish the appearance of these artifacts and the noise. We used the Noise2Inverse, a deep learning method that does not make use of a dataset, for noise reduction. We also used filtering techniques during pre-processing to mitigate artifacts, all while preserving critical diagnostic details of the reconstruction images.

2 Notions of Computational Imaging and Tomography

Computational Imaging and Tomography play an important role in understanding the inside of an object or scene without needing to alter it. Computational imaging can be defined as the process of forming images from measurements, whilst tomography is the visual representation of an object form by sections, using a penetrative wave.
In order to observe such an object, it is necessary to have:

- A signal generation or probing mechanism generally called a **source** that probes the structure of the object. These signals are usually called **sinograms** in the 2D plane and **projections** in the 3D plane;
- A digital detection mechanism called a **detector** which measures the signal;
- A **mathematical model** that explains and translates how the signal relates to the structure of the object or scene.

The source and the detector describe a movement which is called **projection geometry** or **geometry**, although in some cases the one describing this movement can be the object itself whilst the source and detector remain static. This geometry is used alongside the mathematical model to compute the **reconstruction**, which is the result of computing the inverse problem. This aims to express the structure of the object as a function of the measurements (projections), which creates the need for a **reconstruction algorithm** that mathematically transforms the projections into the object which also contains information about its structure.
Tomography has many uses in various disciplines such as radiology [1], archaeology [2], biology [3], and many other scientific areas for research purposes or even for industrial purposes [4] to detect flaws in the products and more. Nevertheless, all share the same principle: to obtain detailed internal images of an object. This present project is centered in the medical imaging discipline, concretely in the Computed Tomography scan (CT scan) which uses X-ray as the penetrative wave to obtain detailed images of the internal parts of the body.
Additionally, to avoid risks on living organisms, an **imaging phantom** or simply **phantom** is used in the medical field. These are objects which are specially designed to be scanned in order to evaluate, analyze, and tune the performance of the imaging device.
Generating the sinograms from the phantom or object is known as the **forward problem**, and the opposite process, where the object and its structure is obtained from the sinogram, is known as the **inverse problem**. These two concepts are fundamental in tomography, and thus will be further explained in the following sections along with the common problems that can arise during the reconstruction.

2.1 Forward Problem

The forward problem is the first step in the reconstruction pipeline. As shown in Figure 1, the forward problem can be described as the process of obtaining the projections of an object. First, the source emits a penetrative wave that passes through the object. The wave is then captured by the detector mechanism and finally encoded and stored as data,

usually called projections. The penetrative wave sent and the detector mechanism vary depending on the object and the medium in which the waves travel.

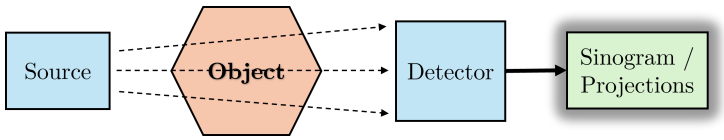


Figure 1: Forward Problem

In the case of CT scan, an X-ray tube is used as the source which is moved around the object at the same time as the detectors, creating the geometry. Normally, conventional scanners use energy-integrating detectors which measure the total x-ray energy deposited in the detector during each measurement interval. This allows for the computation of the attenuation of such rays which vary depending on the object composition (different tissues in the body). On the other hand, AB-CT's nu:view machine makes use of photon-counting technology [5] to measure the data. The photon-counting detectors convert individual x-ray photons into an electric signal proportional to the energy deposited by the photon [6]. This helps to keep track of individual photon interaction and the respective deposited energy, therefore gaining resolution in the measurements.

2.2 Inverse Problem

As the name implies, the inverse problem refers to performing the inverse procedure than the forward problem. This happens by reconstructing the object using the projections obtained in the forward step. Figure 2 outlines the process.

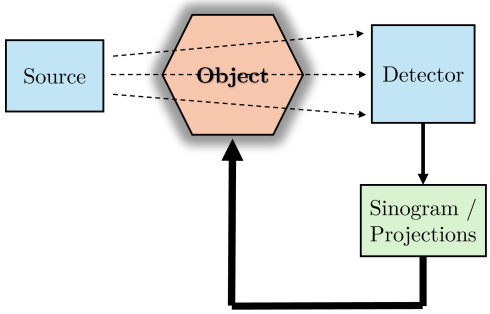
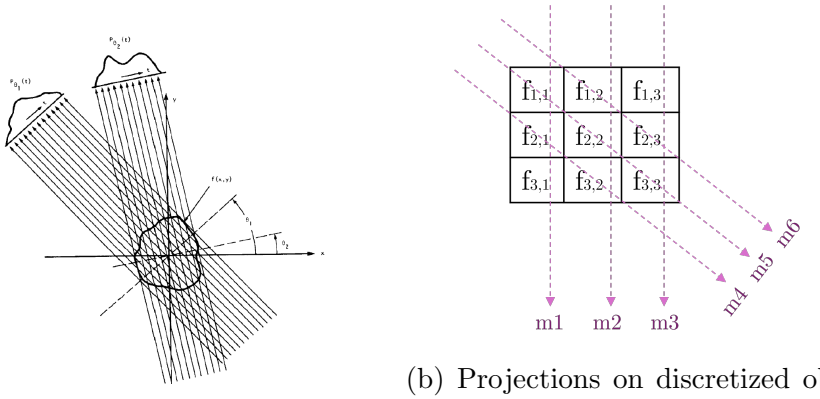


Figure 2: Inverse Problem

Through the inverse problem, the composition of an object can be analyzed and observed without altering its structure. This gives rise to the question of how to express the structure of the object as a function of the measurements.
The inverse problem is first examined in 2D to later extrapolate the findings in 3D. As found in Principles of Computerized Tomographic Imaging by Kak and Slaney [7], Figure 3a depicts a slice of an object parametrized by the function $f(x, y)$ that expresses the density of the object in any (x, y) coordinate. In tomography, the line integrals of this function f are measured across stacks of lines in different angles with respect to the object. Equation 1 shows the mathematical representation of this measure in accordance to Figure 3a. This mathematical representation is known as the Radon Transform. Helgason presents a more detailed explanation of the Radon Transform [8].

$$P_\theta(t) = \int_{line(\theta,t)} f(x,y)ds \tag{1}$$



(a) Projections of an object slice. Source Kak and Slaney [7]

(b) Projections on discretized object. Source: online lecture notes of Prof. Siltanen [9]

Figure 3: Tomographic reconstruction problem

By discretizing $f(x, y)$ as in Figure 3b, the object is transformed into pixels or voxels. The problem can then be defined as a system of linear equations such as in Equation 2, where m is the measurements or projection data, f is the unknown image, and A is our linear equations system matrix. This can be translated as a summation such as in Equation 3, where each pixel x_j is unknown and corresponds to a column in the matrix A, and for each angle there is a line measurement b_i that gives an equation corresponding to an arrow m_k in the matrix A. Object f can then be isolated as in Equation 4, successfully representing the structure of the object as a function of the measurements.

$$m = Af \tag{2}$$

$$m_i = \sum_{j=1}^{n} a_{ij} f_j \tag{3}$$

$$f = (A^T A)^{-1} A^T m \tag{4}$$

The inverse problem can therefore be solved, although solving f from the measured data can be complex, as in general the matrix $A^T A$ is not invertible. Equation 5 shows the matrix A for the example in Figure 3b. A is large, even with this small amount of data of nine pixels and only two projection angles. If there were hundreds of projection angles and images of 2048 pixels, the task of storing A would only be possible with high computational power available in terms of GPU memory.

$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \\ m_6 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & \sqrt{2} & 0 & 0 & 0 & \sqrt{2} & 0 \\ \sqrt{2} & 0 & 0 & 0 & \sqrt{2} & 0 & 0 & 0 & \sqrt{2} \\ 0 & \sqrt{2} & 0 & 0 & 0 & \sqrt{2} & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} f_{1,1} \\ f_{1,2} \\ f_{1,3} \\ f_{2,1} \\ f_{2,2} \\ f_{2,3} \\ f_{3,1} \\ f_{3,2} \\ f_{3,3} \end{bmatrix} \tag{5}$$

Nevertheless, there are many methods to solve large systems of linear equations. It is important to note that the matrix A often cannot be represented and the system of equations can be highly inconsistent due to having multiple solutions (as it is normally undetermined), and thus one must choose between solutions.

2.3 Reconstruction Methods

The reconstruction is the result of computing the inverse problem. Different methods can be used such as an analytical method, which does not solve the system of equations, but instead uses an entirely analytical version of the inverse problem to derive an inversion formula. In order to "solve" (as it is undetermined) the inverse problem, an algebraic method can be used which solves the system of linear equations in Equation 2. These algebraic methods are known as iterative reconstruction methods.

2.3.1 Analytical Reconstruction Methods

These methods have been the standard approach for decades. The aim is to directly invert the measured projection data using mathematical formulas, similarly to Equation 4. These methods shine due to their speed and stability.

- **Simple Backprojection (BP)**: The simple backprojection is the most basic method, as it is the adjoint operator of the Radon Transform. This redistributes each projection value back along its line of integration. For a set of projections $P(\theta, t)$:

$$f(x, y) = \int_0^{\pi} P(\theta, x\cos\theta + y\sin\theta)d\theta \tag{6}$$

  For each point in (x, y) in the image space, the projection value at the detector position $t = x\cos\theta + y\sin\theta$ integrating over all θ is computed for each angle θ. Equation 6 accumulates the contribution of each projection ray that passes through (x, y), effectively computing its value. This method's results are simple, effective, and fast, but it performs a smoothing effect in the reconstructed image (Figure 4) due to the aforementioned accumulation. This effect is spread across a wide area but is especially visible in the center due to the overlapping of the backprojected lines. Furthermore, the edges and fine details can be lost, resulting in a low spatial resolution.

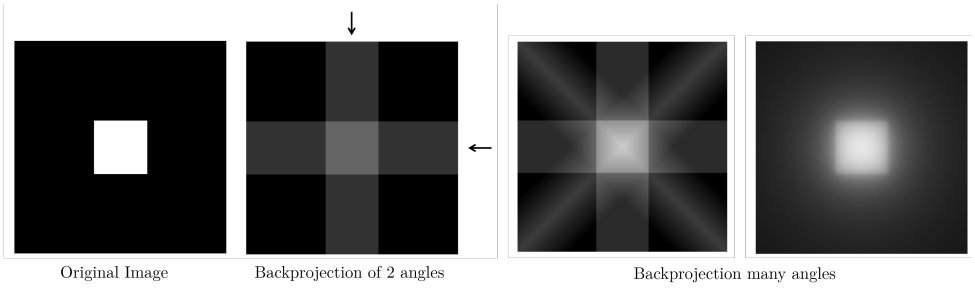


Figure 4: Backprojection with increasing angles. Source online lecture notes of Prof. Batenburg [10]

- **Filtered Backprojection (FBP)**: This method surges as a solution to the smoothing effect in the BP. By applying a high-pass filter in each projection the smoothing effect is countered, resulting in a much more reliable reconstruction as can be observed in Figure 5. Overall, the FBP is a staple reconstruction method, being extremely fast and computationally efficient which allows for real-time medical CT imaging. Nevertheless, the quality of the reconstruction is directly proportional to the number of projections. This is then not ideal for cases such as low radiation CT, where a small amount of projections would intensify the noise, or a high number of projections would lead to unnecessary radiation exposure. Additionally, it does not have a built-in regularization method, therefore the high-frequency noise in the data is amplified by the filter.

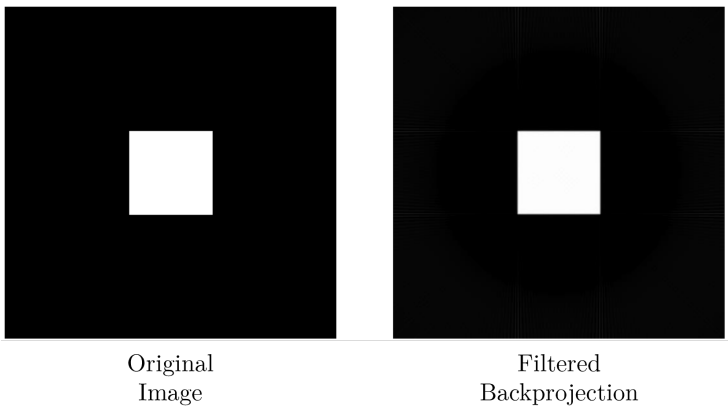


Figure 5: Filtered Backprojection. Source online lecture notes of Prof. Batenburg [10]

2.3.2 Iterative Reconstruction Methods

Iterative Reconstruction (IR) methods attempt to guess the image by successive approximation. Basically, these methods start with an initial image guess, and then they perform iterations refining this guess by comparing the forward projections of the current guess with the measured projections, updating the image at each iteration until convergence.
Iterative algorithms can be categorized by how they model the reconstruction problem. Two important categories are the algebraic methods (which treat reconstruction as solving linear equations) and statistical methods (which derive from optimizing a probabilistic image model, often with regularization).

- **Algebraic Reconstruction Technique (ART)**: Similarly to analytical reconstruction methods like FBP, ART (first introduced by Gordon, Bender & Herman [11]) treats the problem as a linear equation. However, they consider one projection ray at a time instead of the whole image. In practice, they calculate the current line integral (forward problem) through the estimated image and compare it to the measured value (measured projections), correcting the pixels along the ray proportionally to the error. ART converges the estimated image toward an image whose projections match the data, by cycling through all measured projections. ART and its variants handle incomplete or sparse data better than single-pass methods, and they make minimum assumptions of the data, frequently yielding an image where FBP would fail completely (although with some artifacts). The main issue is computational time, as ART is slow to converge and might take hundreds of iterations to yield an acceptable image since every iteration makes use of all the data.

- **Simultaneous Iterative Reconstruction Technique (SIRT)**: Introduced by Peter Gilbert in [12], this method is a variation of the ART. It only differs because it averages the correction after a full set of projection rather than at each ray, ensuring that an artifact or local errors do not heavily impact the whole reconstruction. This method slightly improves upon some of ART's issues, but still shares its computationally taxing nature.

- **Statistical and Model-Based Iterative Methods**: These methods approach the reconstruction problem from a probabilistic perspective. Instead of solving the linear equations, they compute the probability of an image to produce the measured data given a model of the noise and its physics. Therefore, these methods formulate a likelihood function for the image given the measured data and aim to maximize this likelihood. Some of the most used methods are the Maximum-Likelihood Expectation Maximization (MLEM), Ordered-Subsets Expectation Maximization (OSEM), and Model-based iterative reconstruction (MBIR).

  These methods produce images with lower noise and fewer artifacts when compared to FBP by taking into account the physics and noise statistics. Additionally, model-based methods can incorporate prior knowledge or constraints, which can be tuned to preserve edges or suppress noise.

  The downside of these algorithms is their demand on computational resources, which is orders of magnitude more compute-intensive than FBP, especially for full 3D high-resolution images.

2.4 Common Problems

Computational Tomography attempts to compute high quality images of interesting regions of an object of which there is only limited data available and using the least possible computational time.
The limited data can come from two different sides. First, its unavailability; being from a finite number of projections or projection angles of which ideally the more there are the better, but this slows the computational time. Furthermore, there can be a finite angular range where there is no possibility of scanning the object or scene in a particular way, for example, when scanning a patients leg, and there are some angles from which a scan

cannot be done due to the torso or foot being in the way. Secondly, there can be poor quality on the data, which also results in limited data given that the data has less value. This is normally due to noise, object motion, or model mismatch. The later two can also introduce artifacts.

2.4.1 Noise

Noise is present in every real world measurement, and is therefore a part of the final product. The noise affects the overall quality of the reconstruction image. It is difficult to determine the source of the noise, as it can be due to many small factors converged together such as temperature, humidity, and light, which vary depending on the measurement taken. Therefore, when the environment is controlled, the noise can have a particular source, such as in photon counting technology.
Noise can be minimized but not completely removed as it lacks a pattern and is completely random. There are several methods to diminish the amount of noise both in pre-processing and post-processing. These methods will be explained further in Section 4.
In regard to CT, noise impacts the reconstruction generally by affecting the projections. For example, in photon counting technology (used by the nu:view machine) the noise appears due to the error of low photon counts, which results in a statistical Poisson or Shot noise distribution [13]. The noise is normally more visible in homogeneous areas, where it can be seen as a fuzzy bright and dark pattern over the area. Figure 6 showcases an example of the same reconstruction with and without Poisson noise, where these random streaks and fuzziness are visible.

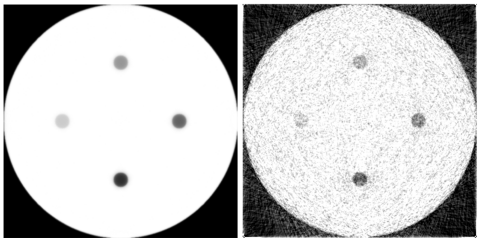


Figure 6: Poisson noise in a reconstruction

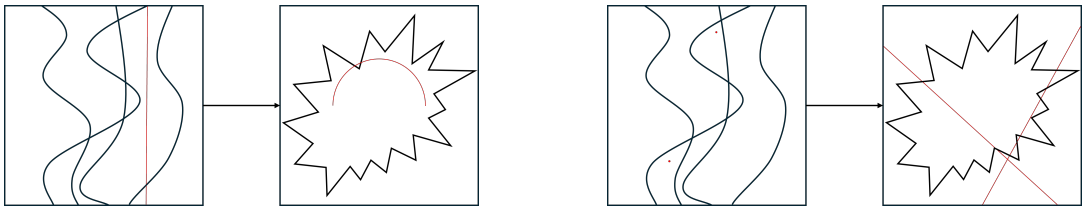
2.4.2 Artifacts

In the reconstruction process, there exists the possibility of interferences or problems which can cause irregularities in the image. These are called artifacts, and they alter the reconstruction locally. These appear due to a specific cause, and the artifacts are therefore defined within different types depending on the resulting disturbance they cause in the reconstruction. Artifacts are usually small issues that can appear in the projections and directly affect the reconstruction. There are many types of artifacts, the most common being zinger and ring artifacts:

- **Ring artifacts**: Arising from errors in the detector, these artifacts are shown as straight lines in the sinogram images, which after reconstructing become ring-like shapes centered in the rotation center that hinder the reconstruction [13]. These detector errors can be miscalibrations or defective parts in it which give irregular

responses. For example, if there is an offset at all angles when compared with the neighboring pixels, this in 180 degree scanning tomography gives birth to a half-ring artifact, as is depicted in Figure 7a

- **Zinger artifacts**: Zinger artifacts are caused by high value spots in the sinogram images, normally due to a detection of a high-energy photon [14]. These high value spots become similar to crossing streaks after the reconstruction. Figure 7b showcases a schematic drawing of such behavior.



(a) Ring artifact

(b) Zinger artifact

Figure 7: Artifact depictions in the sinogram space and reconstruction. Source: by Jiang et al. [15]

3 Dataset Analysis

The data for this project was provided by two different sources, the LUMC radiology department and the AB-CT team.
At the start of the project we visited the LUMC where the team showed us a reconstruction using their visualization software which allowed us to see slices of the breast in every axis. They showed us the small ring artifacts that appeared in the reconstruction, as well as the noise. They also provided a projection dataset.
Figure 8 shows an example of a reconstruction made by the nu:view machine at the LUMC where a ring artifact is visible in the center. This brings the center-point of this thesis: Why did these artifacts appear? Can they be avoided or removed?
The image also shows the aforementioned noise, which is most visible in the gray area around the center part. Where the tissues of the breast should be the same, we observe a gray, distorted, fuzzy pattern that hinders the quality of the overall image, as it can blur fine details crucial to the diagnosis. Another question we planted was whether the noise could be reduced without removing the fine details of the image.

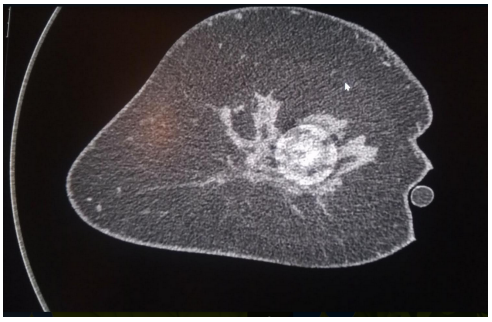


Figure 8: Reconstruction with ring artifact

We had a fruitful meeting with the AB-CT team where they clarified the behavior of the nu:view machine geometry and the pre-processing corrections they implemented in order to reconstruct images. They also kindly provided us with two datasets, with and without the corrections, which contained the projections and their respective reconstructions of a phantom.

3.1 AB-CT's nu:view

The nu:view machine has a complex geometry. It consists of two different concatenated movements. The first movement consists of a full circular movement around the object, and then it transitions to a downwards spiral movement that performs about two full rotations around the object. The full movement is described by a cone beam trajectory, Figure 9 depicts a plot that simulates the trajectory.
The detector is not a flat panel, but a curved panel detector composed of 44 different detectors in a grid-like pattern. Ideally, these detectors are perfectly aligned and centered correctly in such a way that there are no gaps or overlapping between them. The detectors are therefore able to catch all the rays shot by the source, thus obtaining all the information precisely.
This is far from what occurs in reality. During movement, the detectors struggle to stay in the perfect position, therefore they move slightly in any direction (u,v, or d). This results

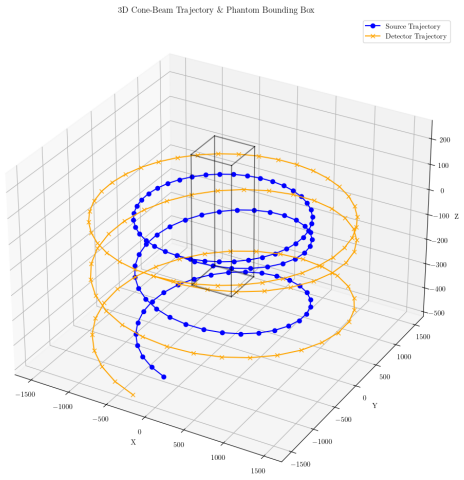


Figure 9: Source and detector movement of the nu:view machine

in parts where the detectors overlap or separate. Figure 10 exemplifies this behavior.
The first thing to consider was what happens when the separation between detectors is altered. This is theoretically simple to answer, as when the vertical separation between detectors creates a line-like structure of empty space, it can be deduced that a ring artifact will appear. However, what would happen when the detectors are overlapped? Or if this movement happens in the horizontal axis instead? Another question that arose was whether artifacts would appear if the detector curvature was tampered with?

(a) Ideal (b) Real

Figure 10: Detector Distribution

3.2 AB-CT's Pre-processing

In order to remove the complications which arose through the virtual flat panel detector, the professionals at AB-CT performed diverse steps.
First, they applied a virtual correction to the detectors, shifting them in place and removing the gaps between them. The absence of gaps ensured that there was no data loss. Figure 11 depicts this correction, where now the detectors only overlap between them.
Despite the overlap bringing about some issues, a reconstruction could still be performed since there was no data loss. Regardless, there were still 44 detectors, and to reconstruct with multiple detectors an iterative method has to be used. High quality 3D CT images tend to have high computational requirements and time. In order to avoid the high computational time, the AB-CT team decided to create a virtual flat panel detector in front

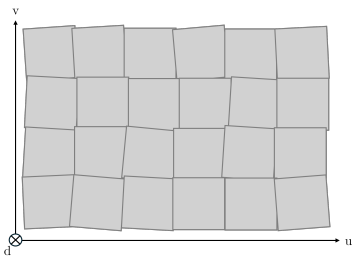


Figure 11: Data loss correction on Detector Distribution

of the detectors, thus simplifying the reconstruction to a single detector. However, doing so brings some challenges, which is why this thesis will later investigate what would happen if reconstruction used iterative methods. Ultimately, the AB-CT team decided to use the single big flat panel detector which allowed the use of methods like FBP for the reconstruction. As shown in the depiction in Figure 12, the new virtual detector sits in front of the curved row of 44 detectors.

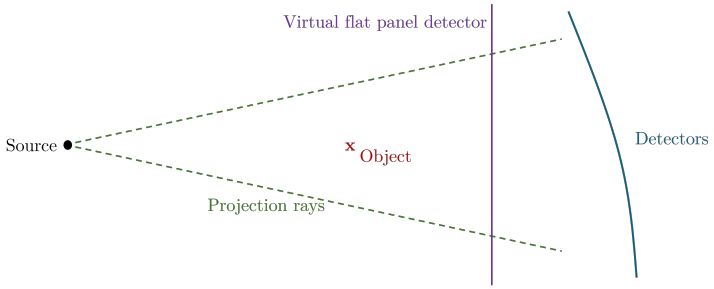


Figure 12: Virtual flat panel detector relative position

The new virtual detector position implies that the rays which impact it have a different angle than when impacting the 44 detectors. This mismatch in the angles requires a perspective correction of every ray detected, as now they theoretically are impacting in another position. Therefore, the team performed the perspective correction of the rays, ensuring that the object was not distorted when reconstructing.
However, performing the perspective correction brings another issue. The corrected projections have lines forming a grid-like pattern, and the lines appear where the different detectors intersected due to the perspective change. Figure 13 showcases the projections of a phantom after the perspective correction.

Figure 13: Phantom projection after perspective correction

As the grid lines are notorious enough to generate ring artifacts and hinder the result, the AB-CT team finally applied a simple interpolation to smooth it out in the aim of reducing the possible ring artifacts that could be generated. Figure 14 shows the same projection of the phantom after the interpolation is performed.

As can be observed, the grid lines are almost completely removed. With this, the projections were ready for the reconstruction step, as they fully implemented the virtual flat panel detector. Having only one big flat panel detector meant that a simple reconstruction algorithm like FBP was now feasible, and the reconstruction could be performed in very little time.

Figure 14: Perspective correction and interpolation

Given that the aim of the machine is to generate a result almost immediately, the interpolation applied was not exhaustive, thus smoothing out the grid pattern but not removing it completely. This trade-off is what gave birth to some small ring artifacts. Some reconstructions made with the nu:view machine at the LUMC hospital which feature the artifacts can be observed in Figure 15.

Figure 15: LUMC Phantom reconstruction

4 Related Work

This thesis is based upon the work of academics and professionals developing methods and solutions through trial and error. This section outlines the primary works which relate to the work undertaken in this thesis. These were particularly important when trying to remove the noise and artifacts of the reconstruction. Their direct impact upon the methods and processes included in the development of the work of this thesis will be established in Section 5.

4.1 Denoising in CT

The main challenge when denoising an image is to conserve the details, especially in CT. Denoising methods are distinguished into two approaches, pre-processing and post-processing. In pre-processing, the noise in the sinogram domain is tackled before performing the reconstruction. On the other hand, in post-processing, the removal of the noise in the image domain is done after the reconstruction.

- **Denoising ct images using median based filters: a review** [16]: This review exemplifies the use of median filters, which are simple denoising approaches in the image domain (post-processing) that might not excel in conserving details, but adequately reduce the noise level. Other spatial filters can be used for image denoising in post-processing such as Gaussian or bilateral [17].
- **Adaptive nonlocal means filtering based on local noise level for CT denoising** [18]: Li et al. proposed an adaptative Non-Local Means (NLM) as a more advanced transform-domain (pre-processing) filter (e.g., wavelet thresholding, Fourier filtering). NLM averages similar patches in the projections to reduce noise.
- **Denoising low-dose CT images using multiframe blind source separation and block matching filter** [19]: Similarly to NLM, Hasan et al. suggested Block-Matching 3D (BM3D) which averages stacks of patches, working equivalently for 3D images.

Such methods have proven effective in CT denoising. Even though they improve the signal-to-noise ratio (SNR), they often involve trade-offs such as excessive smoothing, which may blur the fine details, or heavy computation in the case for high-resolution 3D CT data. Nonetheless, these methods provide a baseline for noise reduction in CT and are typically used alongside or as a pre-processing step for more advanced methods.

4.2 Ring Artifact Removal

As with denoising, pre-processing and post-processing methods are distinguished when removing artifacts.

- **Superior techniques for eliminating ring artifacts in X-ray micro-tomography** [20]: Vo et al. put forward a more modern set of pre-processing filters that target different stripe types (full, partial, fluctuating, ...) depending on the type of the causes of the artifact. These methods are fast and very effective.

- **Reduction of ring artifacts in high resolution micro-CT reconstructions** [21]: Sijbers and Postnov presented a method that converted the reconstructed image into polar coordinates and applied filtering in the angular direction to remove the ring artifacts. As in polar space, a ring is transformed as a horizontal line, and techniques like high-pass filtering or moving-average subtraction eliminate the stripes and therefore the ring artifact.

These methods are effective in removing ring artifacts in CT. Furthermore, they are fast to implement and require very little computational power. However, in cases where the ring artifacts are very intense, it might be challenging to only rely on these methods.

4.3 Deep Learning for Denoising

Deep Learning (DL) has been extensively investigated and is increasingly used with the aim to improve CT image quality, especially in low-dose CT as it often comes with high levels of noise.

- **Low-dose CT with a residual encoder-decoder convolutional neural network** [22]: Chen et al. introduced a supervised network called RED-CNN, a residual encoder-decoder Convolutional Neural Network (CNN). The network trains on pairs of low-dose and normal-dose images, the normal-dose images acting as the denoised counterpart of the low-dose image. Ultimately, the RED-CNN learns to suppress noise while preserving anatomical structures and lesions.

- **Noise2inverse: Self-supervised deep convolutional denoising for tomography** [23]: Put forward by Hendriksen et al., Noise2Inverse is a self-supervised method tailored for tomographic reconstruction that requires no external training dataset. The key idea is to make use of a CNN architecture and use it to reconstruct multiple independent noisy images from noisy subsets of projection data. For example, if the projection angles are split into two groups and these subsets are reconstructed, the result is two images that contain independent noise realizations but the same underlying structure. One reconstruction can then be used as input and the other as the target during training, and the CNN can learn to predict a clean image from a noisy input. Moreover, when applying Noise2Inverse to real data, it significantly reduced noise without blurring fine details.

The main drawback to these methods are the computational requirements, as one has to train a deep neural network. Furthermore, supervised methods rely on high-quality labeled training datasets which are often not available. Nevertheless, these methods achieve significant noise reduction and better structural image fidelity than traditional filtering methods. Self-supervised cases like Noise2Inverse enable training a denoiser on the scanner's own data without the need for ground-truth images, therefore making it possible to denoise and hence improve image quality in situ.

4.4 Deep Learning for Artifact Removal

As stated before, DL as a post-processing tool has seen a lot of use recently, not only for denoising but also artifact removal. In post-processing methods, a CNN is trained to

take in an image with artifacts and output a clean image (for instance, by training on simulated artifact data with known clean targets). This strategy has been explored for metal artifact reduction and ring artifact cleaning.

- **Image denoising and ring artifacts removal for spectral CT via deep Neural Network** [24]: Here Lv et al. proposed a Fully Convolutional Pyramid Residual Network (FCPRN) by combining a Fully Convolutional Network and a Pyramidal Residual Network, resulting in a powerful network that is able to effectively remove ring artifacts and additionally reduce the noise levels on the image.
- **Ring artifact reduction in sinogram space using Deep Learning** [25]: Nauwynck et al. introduced a DL approach for ring artifact removal in the sinogram domain, effectively removing artifacts in pre-processing, which suppresses the lines that cause the artifacts through a line targeting loss function.

In the same way as before, DL methods generally require high levels of computation and a training dataset. Furthermore, CNNs excel at removing noise and/or subtle artifacts, but struggle with highly structured artifacts if the network is not specifically trained for them.

5 Methodology

To begin answering the research questions stated in Section 3, a controlled environment where the impact of every aspect of the reconstruction pipeline could be assessed was necessary. We therefore decided to model the AB-CT nu:view machine. This model would simulate the forward and inverse problem, as there would be a controlled environment which would grant the ability to experiment and test different parameters and configurations settings. This would help potentially determine the origin of the artifacts as well as give the ability to easily apply different algorithms or changes in any step of the process that reduced or removed the artifacts.
In order to create this environment, two main tools were used: Tomosipo and ASTRA Toolbox. These are designed to simulate environments for 2D and 3D tomography image reconstructions using GPU, and can therefore handle different projection geometries and the computations required for the reconstruction.

5.0.1 ASTRA Toolbox

ASTRA Toolbox is a Python and MATLAB based Toolbox presented by Wim van Aarle et al. in [26]. The idea behind this open platform was to create a set of tools for 2D and 3D image reconstruction powerful enough to not be restrained by complex geometries, as were most previous software programs. This presents a set of building blocks that gave flexibility and speed when developing advanced image reconstruction algorithms for tomography.
Thanks to this flexibility, ASTRA is able to accommodate complex geometries, as you can directly input the position of the source and detector and the vector movements for every projection. This makes it possible to simulate any real world scenario. Additionally, it comes with a set of already implemented reconstruction algorithms for 2D and 3D such as FBP, SIRT, and CGLS, among others. These have full GPU compatibility in order to maximize performance.
As a downside, this flexibility can become a problematic when there is a lack experience on the topic, as the lack of visualization tools adds difficulty when trying to understand the geometry's current situation.

5.0.2 Tomosipo

This framework, defined as a *pythonic wrapper for the ASTRA-toolbox* in [27], works as a medium between the user and ASTRA for 3D reconstructions. Being a wrapper, Tomosipo is internally using the ASTRA Toolbox and translating the code inputted to ASTRA functions, thus simplifying the code and making it more user-friendly.
Tomosipo's main advantage are its visualization tools, as at any given point the projection geometry and the movement it describes can be visualized in a 3D image or video/gif. The video format is a Scalable Vector Graphics (SVG) animation and allows saving with a simple function call. Additionally, simple geometries such as a cone beam and parallel beam and transformations such as rotation and translation are quick to set up with a simple line of code. Ultimately, it makes for a more user-friendly experience.
Another important aspect is that Tomosipo accepts tensors, while on the contrary, ASTRA functions only accept NumPy arrays despite working internally with the GPU.

One can construct complex geometries by joining simple movements together. For example, when trying to make a downwards spiral, something that in ASTRA would require a computation for every projection all the positions of source, detector, and movement, in Tomosipo it just takes a simple matrix multiplication of a circular moment by a downwards one, effectively achieving the same result with simple code.
There are, however, some limitations when working with complex geometries that are not product of a combination or when using different detector settings, for instance when working with multiple detectors.

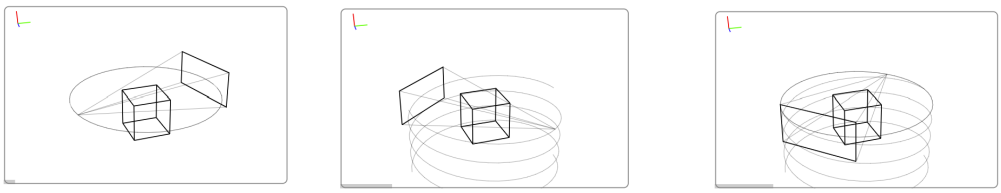
5.1 Pipeline

To mimic the AB-CT nu:view machine, we built the environment by adding a new layer of complexity at every step which allowed the checking of the model's correctness at every key point in the process.
The simulations performed can be divided in two main parts: a one detector simulation, and a multiple detector simulation. For the first part, the reconstruction process of the nu:view machine was simplified. As explained in Section 3, the nu:view machine performs a perspective correction and interpolation on the projections in order to simulate a single flat panel detector with which to reconstruct. To emulate this behavior, one flat panel detector was used and the artifacts and noise that would appear during the interpolation step were manually added. The goal was to test different methods to remove the artifacts and noise, thus developing a post-processing pipeline that would render a clearer reconstruction.
The second part diverged from the nu:view machine reconstruction, mainly because performing the perspective correction and interpolation applied artifacts and added complexity. Instead of simulating the flat panel detector, the reconstruction was made directly with multiple detectors. This made use of a linear reconstruction method to add all projections of every detector together to form the final reconstruction. Using all detectors for the reconstruction had the aim of simplifying the process and potentially avoiding artifacts. As a downside, working with this many projections with an iterative method proved to be quite computationally taxing.
SIRT was used for the reconstruction algorithms. For the case of the simulation with one detector, the reconstruction algorithm did not have an impact on the main goal of ensuring that the post-processing methods worked correctly. A linear reconstruction method was necessary when using multiple detectors, and SIRT offered an easy implementation and good results, and facilitated comparison between the simulations.

5.1.1 One Detector Simulation

As stated previously, the nu:view machine performs several pre-processing steps to reconstruct the images with a virtual flat panel detector. The aim of the first simulation is to model this big flat panel detector as well as the artifacts and noise that appear. In this way, if the artifacts and noise could be accurately removed without affecting the image quality, there would be a solution to the problem which was independent to the reconstruction process. Therefore, the artifact removal algorithms could be applied to the nu:view machine pipeline.
First, the geometry of the machine had to be simulated, which, as explained in Section 3, is a cone-beam geometry composed of a circular movement followed by a downwards

helical movement, performing in total two to three rotations around the object. Given that the geometry is composed of two simple movements, Tomosipo was used due to its simplicity and visualization capabilities. Thus, the two separate geometries were created which can be observed in Figure 16c, and were joined together.



(a) Circular movement (b) Spiral movement (c) Complete movement

Figure 16: Tomosipo projection geometry in SVG format

This model performed the same movement as the nu:view machine, and with that there was a somewhat similar environment that could be controlled. The next step would be to simulate the ring artifacts and the noise seen on the real data. For this, the projections were modified. As explained by Jiang et al. in [15], a line shape offset in the projection space implies a ring artifact when reconstructed. A simple function was implemented which allowed such a line to be positioned at any given point across the projections.
Poisson or Shot noise was used to simulate the noise, modeled by a Poisson process with Equation 7 as the probability mass function (PMF) where the Poisson distribution $f(k;\lambda)$ describes the probability of k events occurring within the observed interval λ.

$$f(k;\lambda) = \frac{\lambda^k \exp^{-\lambda}}{k!} \tag{7}$$

Figure 17 showcases an example of a line added in the projections and the respective reconstruction, as well as the Poisson noise.

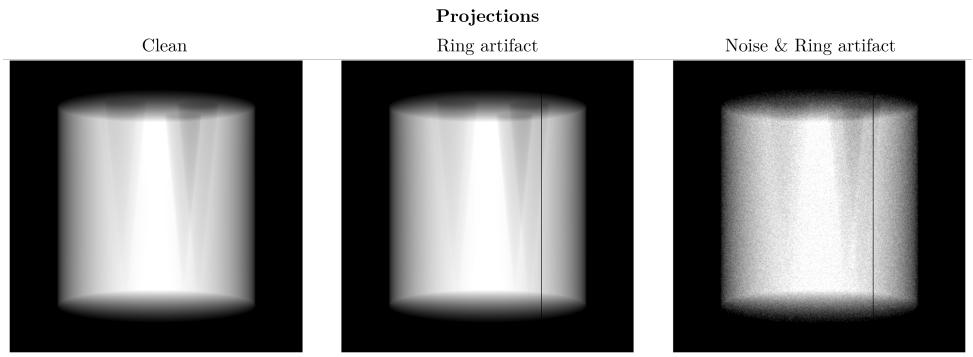


Figure 17: One detector projections with added ring artifact and noise

The next step was to remove the added ring artifacts without affecting the reconstruction quality. The current state-of-the-art has popularized the use of Deep Learning for artifact removal for its great results and flexibility, as it can be used in pre-processing as well as post-processing. However, it comes with the downside of having to train and have a labeled dataset, something that was not available for this project.
Therefore, pre-processing filtering based methods to remove ring artifacts were the best option given the available data. This set of methods, presented by Vo et al. in [28],

are simple to implement, effective, and have minimal impact on the reconstruction. Vo presents different filters to remove the different types of ring artifacts and a unique method that combines the different filters to remove all types of stripes in the projection domain. Furthermore, they are computationally light and require no dataset or training.
With regard to denoising, the noise was added as the filtering techniques require noised data to work correctly. However, if the denoising method effectively removed the noise with multiple detectors, it could also do so with a single detector, therefore the denoising strategies were implemented only in the case of multiple detectors.

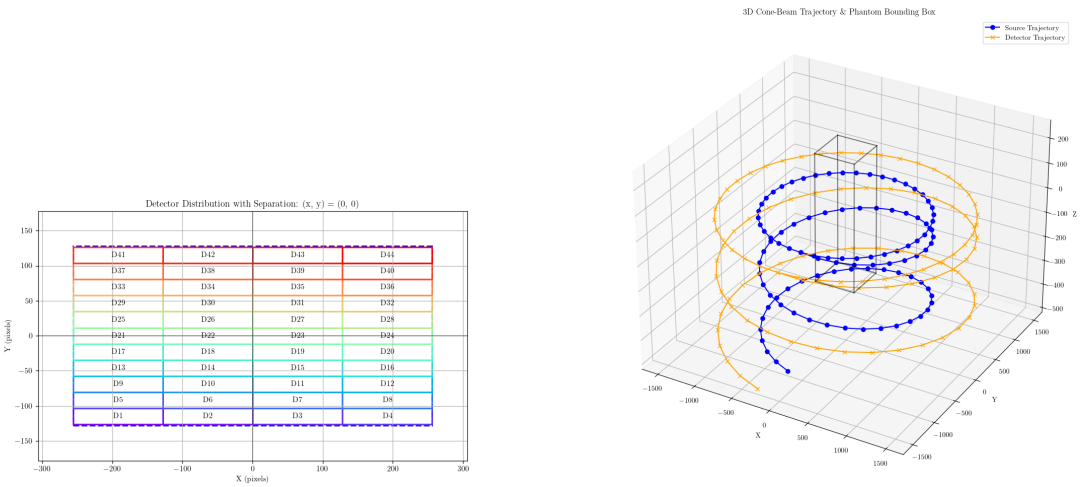
5.1.2 Multiple Detector Simulation

The multiple detector simulation was developed with the aim of mimicking the complexity of the nu:view machine, which is composed of 44 detectors placed in a grid-like pattern with varying distances between them and an overall curvature. The idea was to reproduce such behaviors to understand the cause of the artifacts that appeared in the reconstruction seen in Section 3.
First, two big flat panel detectors were implemented and the behavior of the current simulation was observed. It was at this point that some challenges were encountered while using Tomosipo. Due to how the geometry was constructed in Tomosipo, every projection corresponded to a detector, intercalating them. Therefore, projection n corresponded to the first detector and projection $n+1$ to the second detector. This was not the expected behavior and meant that every detector was skipping one projection, meaning that if there were 100 projections, each detector would have 50, and as they were intercalating, it was not the same 50 projections per detector. This issue only escalated with more detectors, as if the 44 were used, every detector would have once in every 44 projections in total, with none of them matching. This made it impossible to create a proper reconstruction as the same projections were needed for every detector.
In view of this issue, we went a level lower and worked directly with the ASTRA Toolbox which offered a much more flexible implementation of the complex geometry.
In order to use ASTRA, the geometry had to be created from scratch. At any given step, the position and direction of every object in the geometry had to be stated. Following what was done with one detector using Tomosipo, we used the `cone_vec` geometry which required the information of `(srcX, srcY, srcZ, dX, dY, dZ, uX, uY, uZ, vX, vY, vZ)`.
Where:

- `src` : the ray source
- `d` : the center of the detector
- `u` : the vector from detector pixel $(0, 0)$ to $(0, 1)$
- `v` : the vector from detector pixel $(0, 0)$ to $(1, 0)$

This made the multiple detector geometry possible but complex, as all the positions had to be manually geometrically computed for every projection. However, it did mean that any movement could be performed. ASTRA only has the option to create one detector at a time, but every detector can follow the same path one after the other. The flat panel detector was then divided into a matrix of (x, y) columns and rows. If N number of

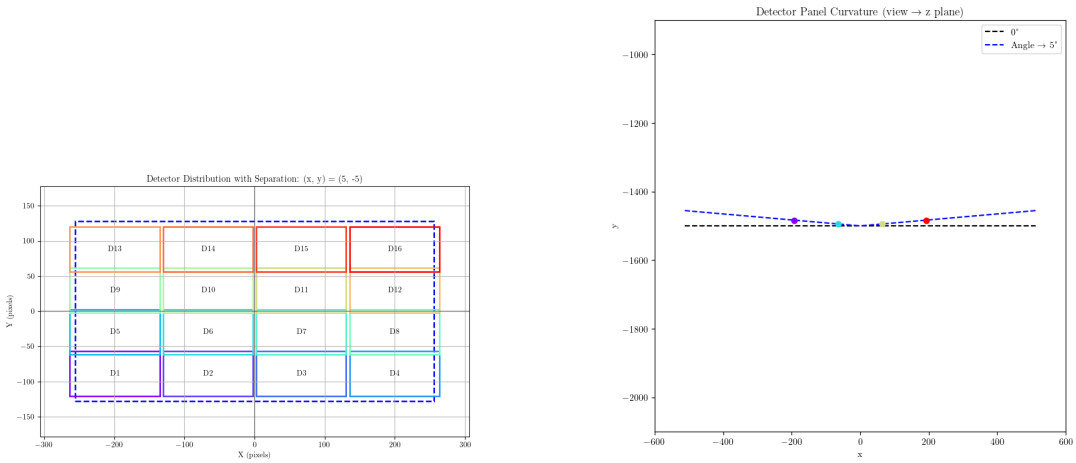
projections were performed, the total amount of projections would be $N \times x \times y$. Despite every detector having a smaller resolution, having that many projections did have an extra computational cost. This linearly increased the reconstruction time as will be seen in the experiments.
Despite ASTRA not disposing of visualization tools, plots were created manually to observe the state of the geometry and correct it if necessary. Figure 18a shows the distribution of the 44 detectors and Figure 18b shows the trajectory described by the detectors and the source, therefore the final projection geometry.



(a) Detector Distribution (b) Astra projection geometry

Figure 18: Astra Projection Geometry and Detector Distribution

As previously described, there were two constraints to reconstructing an image with multiple detectors. Firstly, the number of projections make the reconstruction more computationally taxing. Secondly, a reconstruction algorithm that was linearly independent was needed, as the projections had the shape $N \times x \times y$ and were repeating in a way that only a linear reconstruction method (SIRT) could handle.
To accurately mimic the nu:view geometry, since at this point simulation lacked detector movement variables, these were added to simulate the separation, overlapping, and curvature.
As seen in Figure 10b, the distribution is chaotic in the real case scenario, moving in (u, v) directions and rotating in the d axis. To simulate this behavior, a hyperparameter value for the separation and overlapping was added. The feature was implemented by a constant, which indicated the number of separation pixels there were between the detectors (negative in the case of overlapping). This separation was either horizontal, where the detectors moved along the v axis creating a horizontal separation between the rows of detectors, or vertical, where the detectors moved along the u axis creating a vertical separation between the columns of detector. Figure 19a showcases a vertical separation and a horizontal overlapping between detectors, simulating the lines that appear when the perspective correction is performed (Figure 11). This constant and independent separation allowed the study of the grid-like lines in the projections and their impact in the reconstruction.
In the nu:view machine, the detectors' position describes a curvature similar to an arch, which implies that the x-rays impact at a certain angle in the detectors. This behavior was implemented by adding an angle of curvature with respect to the flat panel detector,



(a) Separation and overlapping

(b) Curvature

Figure 19: Detector movements

making the center of the detectors follow v-like curvature. This ensured that the x-rays impacted at a certain angle, therefore enabling the analysis of whether such behaviors could induce artifacts. Therefore, the curvature was implemented as a hyperparameter which controlled the angle of curvature that the detectors would follow. Figure 19b exemplifies this with a top view of the detectors' centers when they are set at a 5^o angle. Different curvatures were tested to evaluate the impact on the reconstruction images.
Poisson noise was also introduced in this simulation following the same process as in the one detector simulation in Section 5.1.1. The aim was to test the Noise2Inverse as a denoiser for the multiple detector geometry, as if proven effective here it would also work in the one detector scenario. Noise2Inverse uses a self-supervised deep CNN to denoise images for linear reconstruction algorithms and does not require any clean or noisy data. This removed the data limitation and enabled the use of a state-of-the-art method for denoising purposes.

6 Experiments

There were two main objectives to this project, the first to find the root of the artifacts that appeared in the reconstructions performed by the nu:view machine, and the second to remove the artifacts and the noise. In order to achieve these objectives, several experiments were performed which gave insight into and information regarding the artifacts and their nature. Achieving such an understanding helped find and apply measures to remove or avoid the appearance of artifacts and noise.
As mentioned in Section 5, the simulations were divided into two parts; the one detector simulation and the multiple detector simulation. The experiments in this section are all focused on the multiple detector simulation, aside from Experiment 7 in Section 6.10 which corresponds to the one detector simulation.
The hardware used for the experiments was an NVIDIA RTX A5000 with 16 vCPU, $24GB$ of VRAM and $62GB$ RAM. Furthermore, all the plots that showcase reconstructions have been normalized for values [0,1]. This was done to match the phantom values and ensure a good visualization.
Before diving into the experiments, it is important to know more in depth the parameters used for the simulations. The phantom and the metrics used in the experiments will also be explained in more detail.

6.1 Parameters

The geometry of both simulations was the same, a horizontal circular rotation followed by two rotations in a downwards helical movement. Therefore, there were some geometry describing parameters which were set to fit the nu:view geometry that were shared by both simulations:

- **Geometry**: The cone-beam geometry, which simulates a photon tube, was used as the nu:view machine uses a photon tube. This is characterized by source that shoots the photon rays in a cone-like shape.
- **Phantom shape and size**: The phantom shape was cylindrical, and the size was $256 \times 512 \times 512$ (z, y, x) pixels, smaller compared to the $2k$ images of the real case scenario. The smaller resolution was chosen to fit computational resources.
- **Pixel size**: The pixel size was set to 1.0 mm, different to the actual measure of approximately 0.1 mm per pixel. This difference did not affect the results and simplified the geometry computations as there was no conversion.
- **Source to object distance (SOD)**: Set to 1000 mm, this was again different to fit the scale of the simulation which had a higher pixel size and smaller phantom.
- **Object to detector distance (ODD)**: This was set to 1500 mm, same as with SOD.
- **Number of rotations around the object**: There were three in total, one horizontal circular movement and two rotations in a downwards spiral; a similar movement as the nu:view machine.

- **Reconstruction algorithm**: As stated before, the SIRT algorithm was selected as the reconstruction algorithm for the one detector and multiple detector cases.

Furthermore, there were some shared parameters that had to be studied in more depth. Concretely, the number of projections and the number of iterations needed to be tuned.

- **Projection number**: The total number of projection angles used in the geometry. Generally speaking, the more projections there are, the more detailed the data, and therefore the reconstructions would be more accurate. The only limit to this number is the computational cost taken to process that number of projections.

- **Reconstruction algorithm iterations**: This corresponds to the number of iterations the SIRT algorithm performs. Similarly to the number of projections, the limit of this number is also set by hardware, as in broad terms the more iterations used, the better the algorithm will reconstruct. However, it does have a convergence which depends on the data which is directly related to the number of projections.

On the other hand, some parameters affected the specific case of the multiple detector simulation. These parameters were used to assess the behavior of the multiple detectors, such as the number, separation, and curvature as discussed in Section 5.1.2.

- **Number of Detectors**: The total number of detectors the geometry has. Ideally this parameter would be 44, distributed in a matrix of 4 rows and 11 columns as in the nu:view machine. However, due to the high computational cost of reconstruction with the 44 detectors, 5 detectors with a distribution of 1×5 were used for most of the experiments.

- **Detector Separation**: This parameter adds a distance in pixel size between the detectors. The horizontal and vertical separations can be adjusted independently and if the distance is negative, the detectors overlap. This parameter was implemented to study if this separation had a part in the generation of artifacts.

- **Detector Curvature**: This value corresponds to the angle described by the detectors in respect to a flat detector, simulating the curvature described by the detectors. This parameter was also implemented to study if it had a part in the generation of artifacts.

6.2 Phantom

Two different phantoms were used during this project, one mimicking the calibration phantom used in the LUMC and the other one as a simpler version of the foam phantom introduced in [29] to encompass a more realistic tomographic problem.

6.2.1 Calibration Phantom

This phantom was created to mimic the structure of the real calibration phantom used by the hospital to calibrate and test the correctness of the nu:view machine. The phantom used by the hospital is a cylinder, and inside are various smaller cylinders that are made of different materials. To mimic that behavior, a cylinder with four smaller cylinders inside

were created across the whole phantom. The smaller cylinders differed in value, mimicking the difference in materials and composition. Additionally, a new layer of complexity was added to the phantom by transforming the smaller cylinders into cones, adding a vertical variation as can be seen in Figure 20.

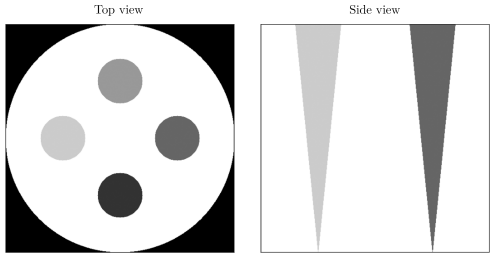
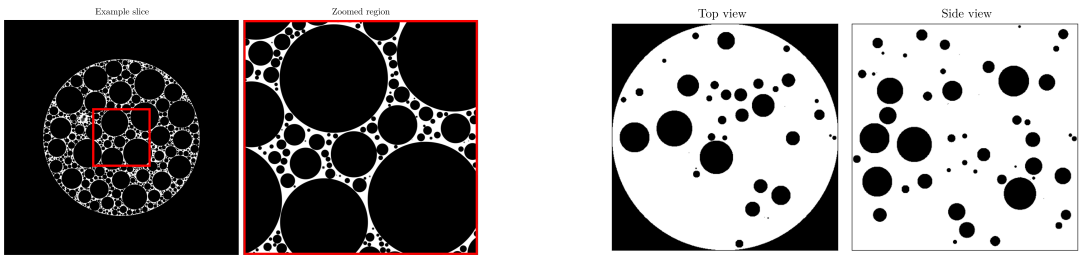


Figure 20: Calibration phantom with vertical cones

6.2.2 Simple Foam Phantom

This phantom, depicted in Figure 21b, was created as a simpler, more computationally friendly version of the original foam phantom introduced in [29] which is shown in Figure 21a. The original foam phantom describes a foam-like structure with thousands of internal features. It is challenging to reconstruct due to the high detail complexity and has a structure that is representative of typical tomographic experiments. Our phantom proposes the same foam-like structure with a simpler design and fewer features. The necessity for a simpler phantom arose since the computational requirements for the original foam phantom were too high when used alongside methods such as Noise2Inverse. Even though complexity was lost in the process, it was still important to retain the foam-like structure due to its intrinsic complexity and similarity to real tomographic problems. Hence, the problem was scaled down while still maintaining the same foam-like approach, rendering a more manageable phantom.



(a) Foam phantom, source from [30]

(b) Simple foam phantom

Figure 21: Foam phantom comparisons

6.3 Metrics

Several metrics can be used to analyze a reconstructed image, the most common being Mean Squared Error (MSE) [31], Peak Signal to Noise Ratio (PSNR) [32], and Structural Similarity Index Measure (SSIM) [33].

In our case, we used the PSNR and SSIM to evaluate the reconstructions made during simulation, as they could be compared to the phantom. The PSNR is useful to measure the noise level and is defined as:

$$PSNR = 10log_{10}(\frac{MAX_I^2}{MSE})[dB] \tag{8}$$

Where MAX_I represents the maximum pixel value of the image I. To define the MSE, a noise-free $m \times n$ monochrome image I and its noisy approximation (reconstruction) K is supposed, therefore:

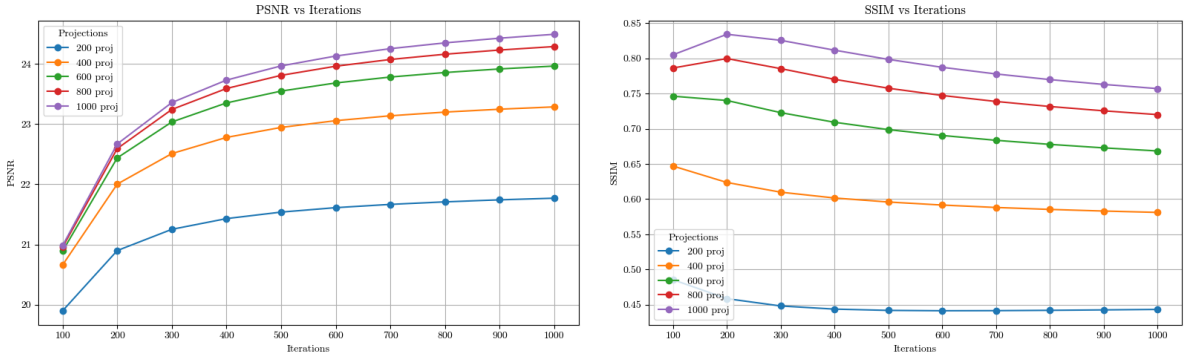
$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \tag{9}$$

Measuring the noise level was useful but had its shortcomings, as a low noise level does not mean that the reconstruction has been successful. For example, there could be a high PSNR indicating a low noise level and, when visualizing the results, it can be observed that the reconstruction was not quite similar to the phantom despite the lack of noise. Thus, we decided to make use of the SSI which measures the similarity between two signals giving a value from 0 to 1 where 1 means the signals are identical. The SSIM is defined as:

$$SSIM(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 = C_2)} \tag{10}$$

6.4 Experiment 1: Optimization

The aim of the first experiment was to optimize the reconstruction algorithm. This was done by finding a set of parameters that yielded the best reconstruction image when evaluating it with PSNR and SSIM metrics.
The simulation had two main parameters to adjust when optimizing for the reconstruction; the number of projections, and, as it used SIRT, the number of iterations of the reconstruction algorithm. Additionally, there were three values that changed the detector geometry: the number of detectors, the separation between detectors, and the angle of the detector curvature. These were tested separately in their respective experiments.
For this experiment, the detector separation and curvature values were set at zero. Only one detector was used given that increasing this number would needlessly escalate computational costs.



(a) PSNR (b) SSIM

Figure 22: Metric results for different projection and iteration values

The values used for the number of projections ranged from 100 to 1000 and 100 to 1000 for the iterations. Figure 22 shows the PSNR and SSIM results per projection and iteration values.
It was observed that the projection number determines in both cases (PSNR and SSIM) the quality of the reconstruction and that the number of iterations has a significant impact in the PSNR in comparison to the SSIM. Furthermore, PSNR steadily increases, peaking at $24.49dB$ for $(projs = 1000, iter = 1000)$, whilst on the other hand, SSIM peaks at 0.834 for $(projs = 1000, iter = 200)$ then slowly decreases with the increase of the iteration number.
The optimal parameters values are those which maximize both metrics, although given the results shown in Figure 22, a trade-off between the two metrics is needed. In order to decide the optimal values, a composite normalization metric of the PSNR and SSIM was performed. This was defined as the sum of the normalized PSNR and SSIM as seen in Equation 12. The normalization of the metrics was performed by following the min-max normalization method [34] defined in Equation 11.

$$m_{norm} = \frac{m - m_{min}}{m_{max} - m_{min}} \tag{11}$$

$$C_{norm} = PSNR_{norm} + SSIM_{norm} \tag{12}$$

Table 1 shows the results of the composite metric. This reveals that the best combined results were obtained with 1000 projections and 800 iterations.

Projs	**Iter**	**PSNR**	**SSIM**	**Composite**
1000	600	24.13	0.787	1.802
1000	700	24.25	0.778	1.804
1000	800	24.35	0.770	**1.805**
1000	900	24.43	0.763	1.804
1000	1000	24.49	0.757	1.803
800	200	22.59	0.800	1.491
800	300	23.24	0.785	**1.507**
800	400	23.59	0.770	1.420
800	500	23.81	0.757	1.327
800	600	23.96	0.747	1.243

Table 1: Reconstruction quality metrics for various projection/iteration settings

There was a complication with these parameters. When introducing multiple detectors, the number of projections is multiplied by the number of detectors. In such a case, there would not be enough GPU memory to store the data when using the available hardware. This would therefore become problematic when using 15 detectors (the number of detectors used in Experiment 3). Due to this limitation, the number of projections was reduced to 800 and the composite metric was recalculated for the new projection number, which is shown in Table 1.
Finally, the optimal parameters chosen for the reconstruction were set to 800 projections and 300 iterations. This set of parameters ensured a high quality reconstruction whilst maintaining manageable computational costs.
Figure 23 showcases the same slice of the phantom for four different configurations of projection and iteration values. This shows that with a projection number of 800, there is still a high quality result despite the lower PSNR compared to 1000 projections.
In summation, the number of projections and iterations are directly proportioned to the image quality when we observe the PSNR. However, with the SSIM, only the number of projections is directly proportioned as the iteration number converges more quickly around 300. 800 projections and 300 iterations were therefore used, as these values give the best combined results in both metrics and are not too computationally taxing.

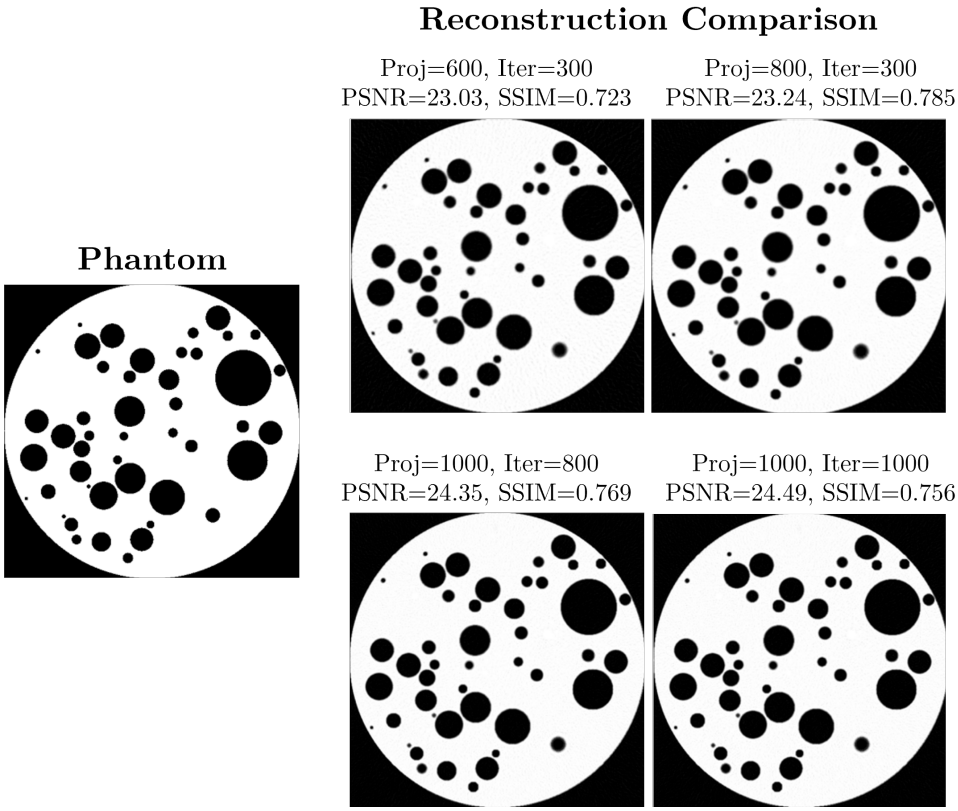


Figure 23: Reconstruction comparison

6.5 Experiment 2: Computational Time

The available computational power has been a limiting factor when performing tests and experiments during this project. It affected decisions such as which algorithms to use as well as the parameters chosen for the experiments. We were therefore brought to evaluate the impact of the proposed algorithm.

In the multiple detector simulation, the main limiting factor was the number of detectors to use because it escalated the number of projections. The resolution of each detector was divided by their number and the number of projections was multiplied. It was necessary to assess the impact this had on our limited computational resources.

For the experiment, a reconstruction using the SIRT algorithm with 100 projections and only one iteration was performed for each detector configuration available. Evaluating the reconstruction time gave insight into the computational requirements of the algorithm. Since the SIRT algorithm performs several iterations, it was only necessary to evaluate the time for one iteration, and using 100 projections enabled the use of the same number of detectors as the nu:view machine. The other two geometry changing parameters, separation and curvature, were set at 0.

Figure 24 showcases the reconstruction time across the different detector distributions. The distribution of the detectors always maintains a rectangular shape, as this is the shape of the detector distribution in the nu:view machine. This accounts for the gaps between some data points in the figure, as, for example, using 43 detectors is not viable in this configuration.

The experiment was performed 10 times. The time was then averaged and the standard

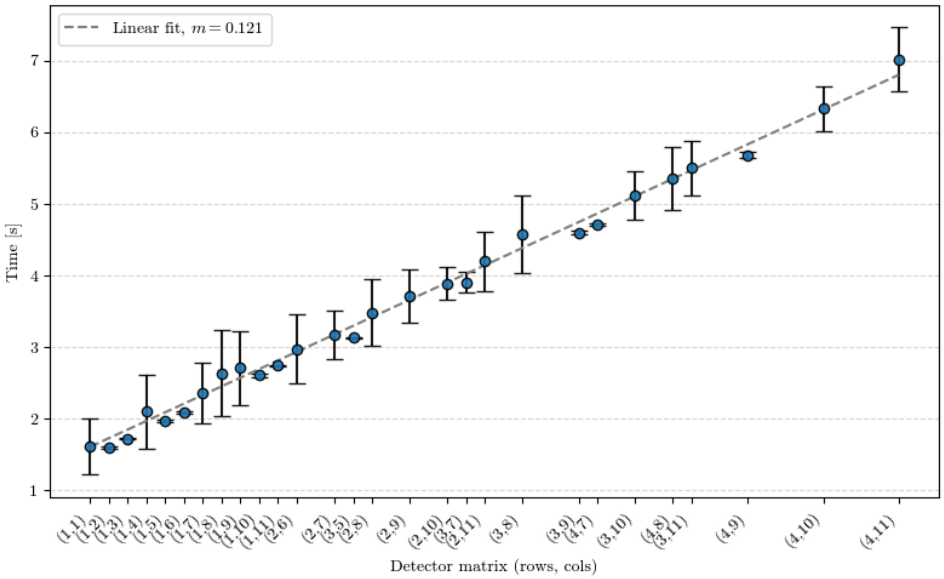


Figure 24: Reconstruction time evolution across detector

variation computed. As can be observed in Figure 24, the reconstruction time linearly increases with the number of detectors. Therefore, a linear fit was performed with the gathered data to better observe the tendency. The gradient of such was $m \approx 0.121$, meaning that with 44 detectors the reconstruction time would approximately increase 5.32 times when compared to only one detector. Note that this increase should be multiplied by the number of iterations used in the SIRT algorithm, as the nu:view machine performs an FBP which is intrinsically only one iteration.
Ultimately, this experiment allowed the quantification of how computationally taxing our method was in comparison to the nu:view machine. Rendering this reconstruction was much more computationally taxing, something that was already theorized when deciding to use SIRT and the different detectors for the task.

6.6 Experiment 3: Detector Separation

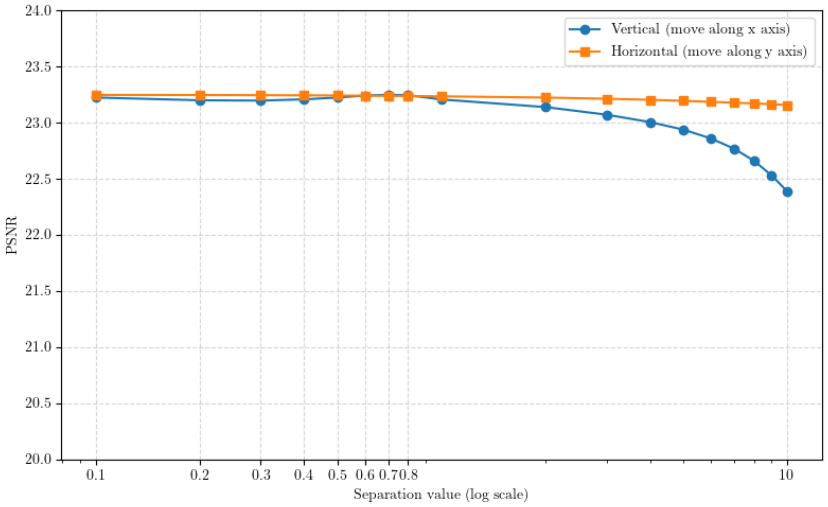
The aim of this experiment was to study the impact the separation between detectors had on the reconstruction. Section 3 explains how the detectors of the nu:view machine had small separations between them. This meant that there was some data lost in the reconstruction as the rays did not impact the detectors. Therefore, it was theorized that some artifacts were bound to appear.

The separation between detectors was modeled as vertical and horizontal independent separations. These cases were studied separately by simulating a constant separation between the detectors.

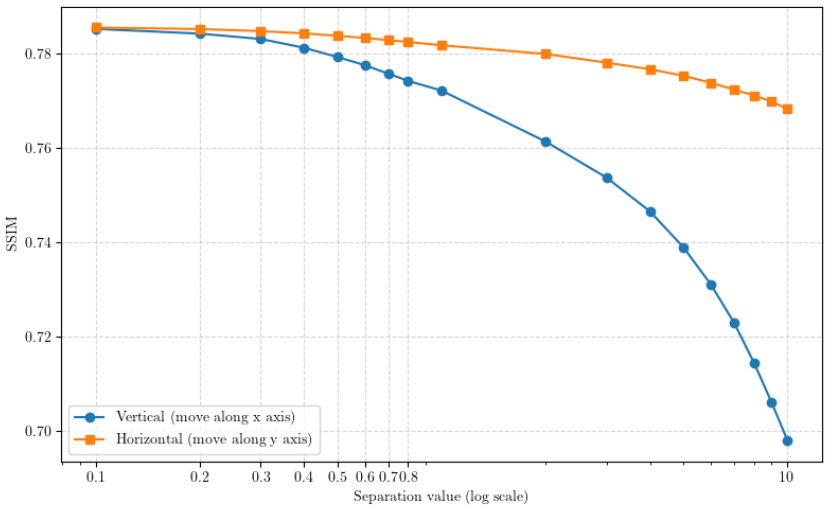
Two experiments were performed, one with horizontal separation and another with vertical. Following the findings in the first experiment in Section 6.4, in both cases the number of projections was set to 800, the number of iterations to 300, and as it was not relevant to the aim of this experiment, the curvature was set to zero. Finally, the number of detectors was set to 15 in a 3×5 distribution to ensure both separations appeared.

In Figure 25, a clear drop in the PSNR and SSIM can be observed when we have a horizontal separation higher than 0.5 pixels. On the other hand, the vertical separation has a decrease in quality, but this is not as pronounced. This is better observed in Figure 26, where the reconstructions can be seen at different horizontal and vertical separation values. It can be seen that the horizontal separation does not induce artifacts due to the nature of the geometry. Some tests were performed only with the circular movement, and there was a clear gap (black line) in the middle of the phantom representing the data lost in that section. However, as the geometry also performs a downwards spiral movement, this data is not lost due to later projections covering the part that was left by the gap between detectors. On the other hand, artifacts appear with the vertical separation, although the artifacts start to be noticeable when the separation between detectors is higher than 0.5 pixels.

The results obtained in this experiment shed light on the nature of the artifacts which appear in the nu:view machine, corroborating the hypothesis made in Section 3. Therefore, the vertical separation or vertical lines that appear in the projections cause artifacts in the reconstruction. However, the horizontal separation does not induce artifacts.



(a) PSNR



(b) SSIM

Figure 25: Metric results for different detector separation values

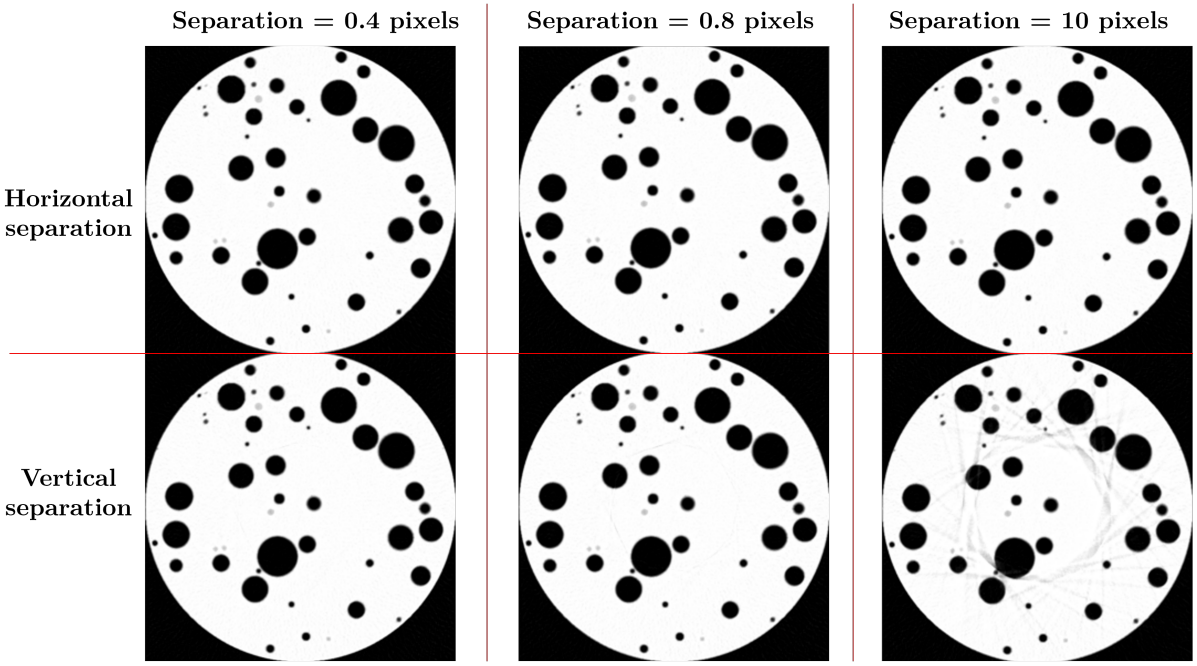
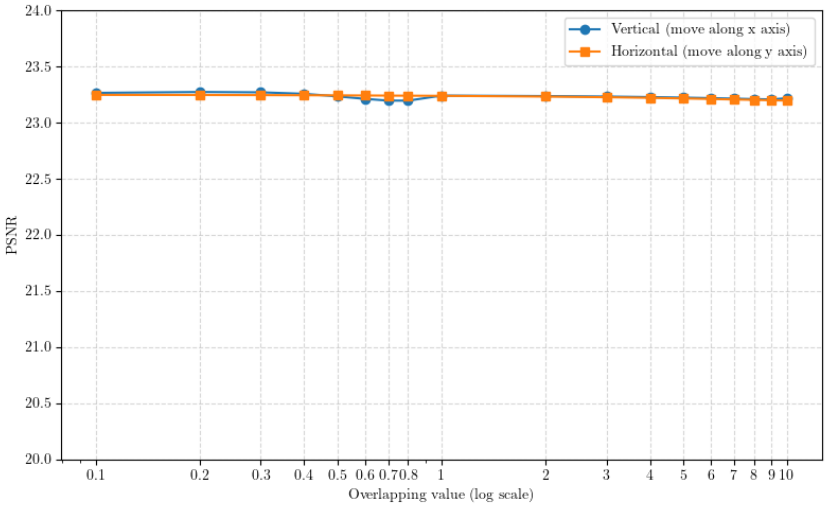


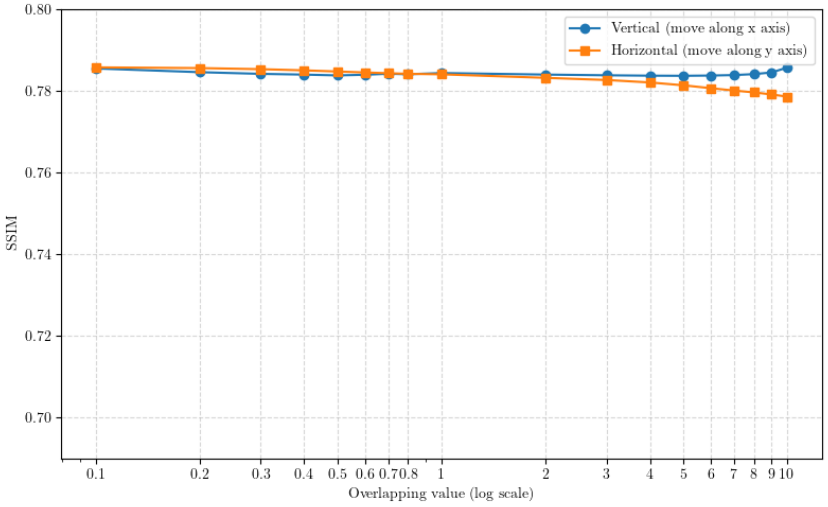
Figure 26: Reconstruction comparison for different separation values

6.7 Experiment 4: Detector Overlapping

Similarly to the separation experiment in Section 6.6, the aim of this experiment was to study the impact the overlapping between detectors had on the reconstruction.
The overlapping observed in Section 3 mainly meant that the rays impacted one or multiple overlapping detectors, depending on the angle of the ray. In this case, therefore, the data was not lost, creating uncertainty regarding whether this behavior would induce artifacts.
The overlapping was modeled in the same way as the separation, but instead of using a positive value to separate the detectors, a negative value was used to overlap them. Both cases were again studied, the horizontal and vertical overlapping constant across the detectors. All the parameters used were the same as in the separation experiment.
Figure 27 shows how the metrics PSNR and SSIM are barely affected by the overlapping of the detectors being horizontal or vertical. More visually, Figure 28 shows the reconstructions for different horizontal and vertical overlapping values, where no artifacts appear in any case. Therefore, the overlapping of the detectors, as it does not have data loss, does not induce artifacts. However, this only holds true if the image is reconstructed with a high enough resolution, as demonstrated in Section 6.7.1
In summation, the overlapping of the detectors does not induce artifacts at a high enough resolution. This is a positive finding, demonstrating that the nature of a linear iterative algorithm like SIRT is able to reconstruct the image without introducing artifacts. Therefore, provided that a perspective correction is performed to only have overlapping between the detectors, this reconstruction method will not induce artifacts in the reconstruction thus solving the artifact problem without the need of post-processing methods or complex interpolation. However, this method, as seen in Section 6.5, does require increased



(a) PSNR



(b) SSIM

Figure 27: Metric results for different detector overlapping values

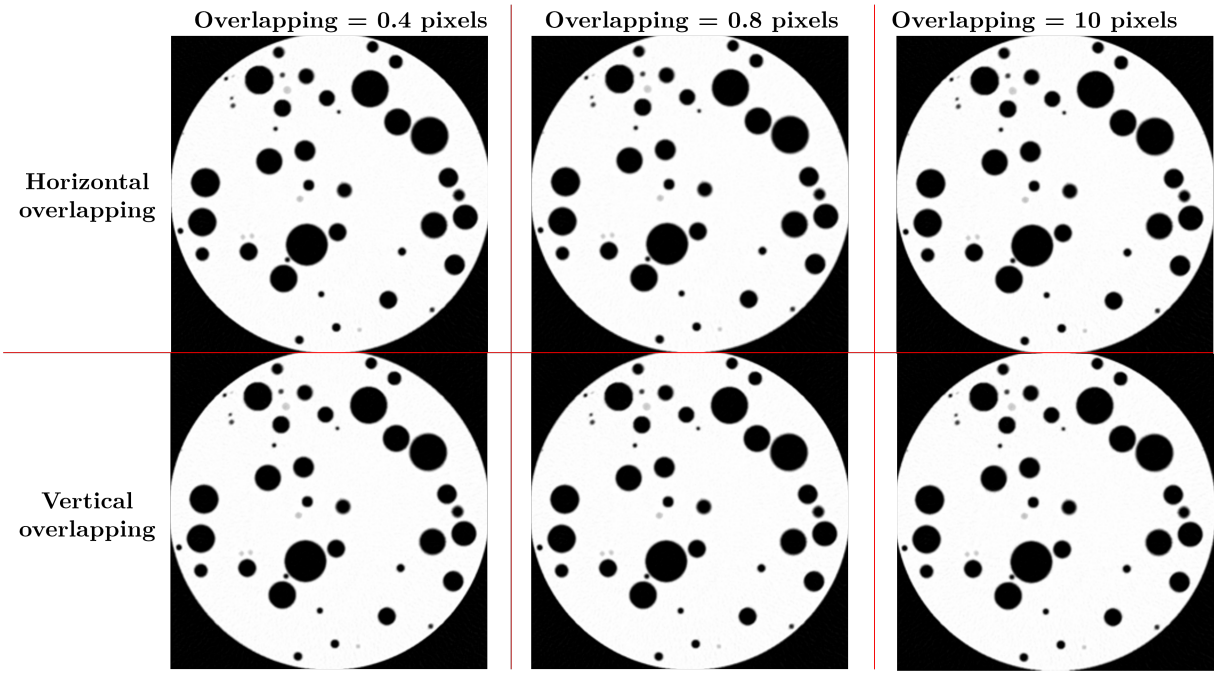


Figure 28: Reconstruction comparison for different overlapping values

computational resources.

6.7.1 Image Resolution when Overlapping Detectors

During the testing part of this experiment, there was a correlation between the absence of artifacts in the reconstruction and its resolution. Basically, at a low resolution, the vertical overlapping of the detectors induced artifacts. The horizontal overlapping never showed artifacts regardless of the resolution. The principal resolution parameter that affected this was the number of iterations of the SIRT algorithm.
Figure 29 showcases the reconstructions with different resolutions with a separation of 0.8 pixels in the vertical direction, where it can be seen how the artifacts are present when the iteration number is under 200.
Artifacts and noise commonly appeared when the projections were below the 200 threshold. The number of projections was therefore set to 200 in order to correctly perform the reconstruction. On the other hand, when the number of iterations was small, there were no artifacts and just a considerable amount of noise visible. However, the moment the detectors overlapped, the artifacts appeared with when using low resolution. A small study was performed to observe this behavior, and finally as seen in Figure 29, it was concluded that the artifacts appeared when the iteration number was under 200.

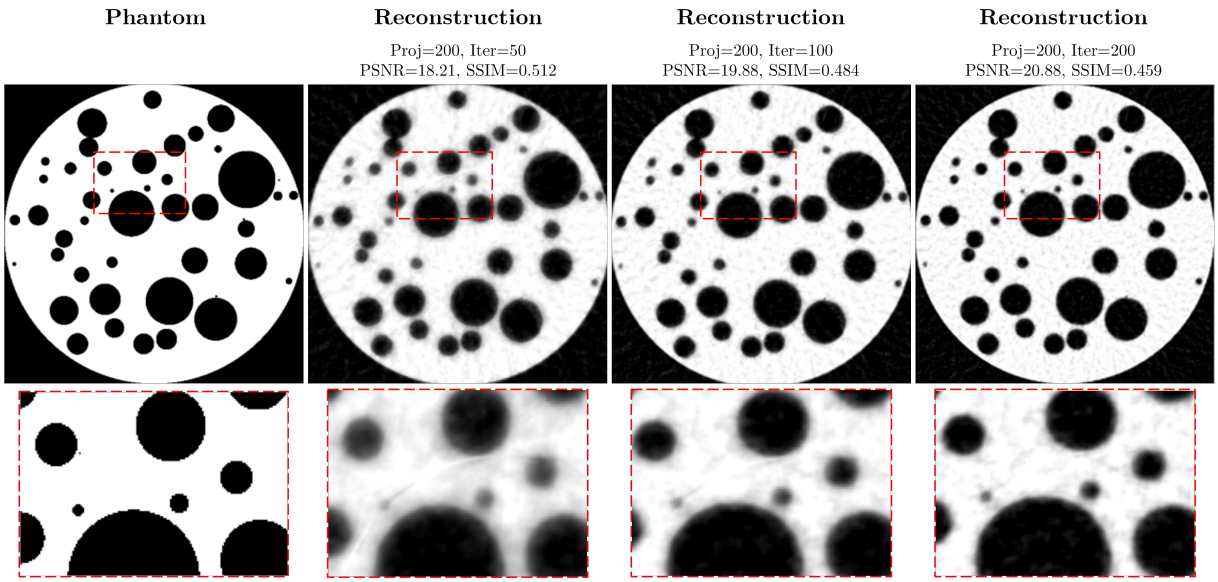


Figure 29: PSNR and SSIM results with an overlapping of 0.8 pixels at different resolutions

6.8 Experiment 5: Detector Curvature

Aligned with the assumptions made in Section 3, the curvature of the detectors could potentially be a cause of observed artifacts.
The detector curvature was modeled by moving the detectors along an angle α around the center of all the detectors as discussed in Section 5.1.2. In order to study the impact of the curvature in the reconstruction, several angles were tested ranging from 0 to 45 degrees while the other parameters remained constant.
Aligned with the results of the first experiment in Section 6.4, the projections were set to 800 and iterations were set to 300. The separation was set to 0 in both directions as it was of no relevance to this experiment. Note that due to the nature of the implemented curvature, an even number of detector columns was necessary as the curvature was applied symmetrically to both sides from the center. Therefore, the number of detectors was set to 12 in a 3×4 matrix in order to avoid folding a detector upon itself, which would generate artifacts in the reconstruction.
Figure 30 shows the results of PSNR and SSIM for all the tested curvature angles and demonstrates how the curvature affects both measurements. With a higher angle there is a lower PSNR and lower SSIM, implying that the curvature adds noise to the reconstruction: the higher the curvature the higher the noise.
In Figure 31, where the reconstructions of the angles $0, 5, 10, 20, 30\&45$ are shown, no artifacts appear in the reconstruction. The is some additional noise which is slightly more visible and adds blurriness to the edges, but it generally still preserves the fine details and does not distort the image.
In conclusion, the curvature of the detectors does not induce artifacts in the reconstruction. However, it does slightly hinder the quality of the reconstruction by adding some noise, as seen in the PSNR and SSIM results.

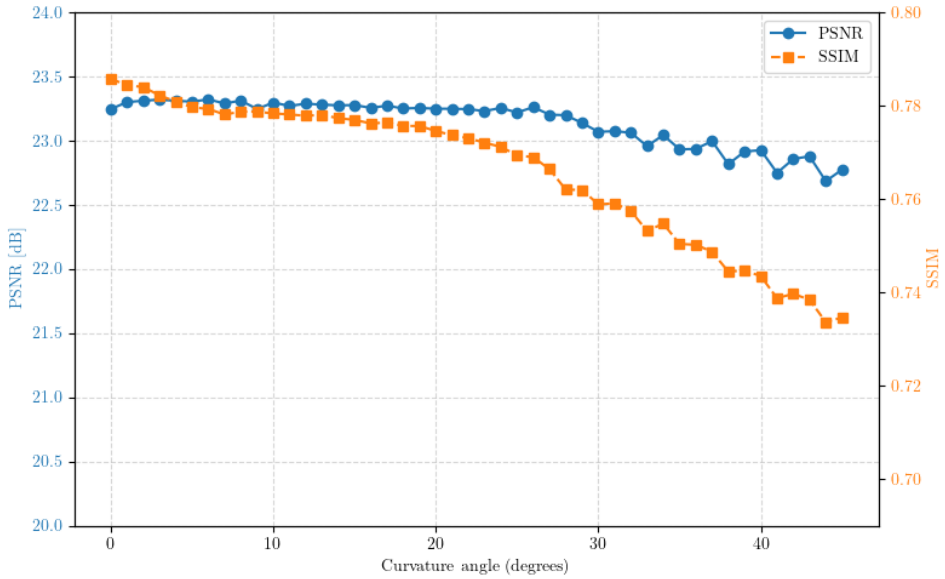


Figure 30: Metric results for different detector curvature angles

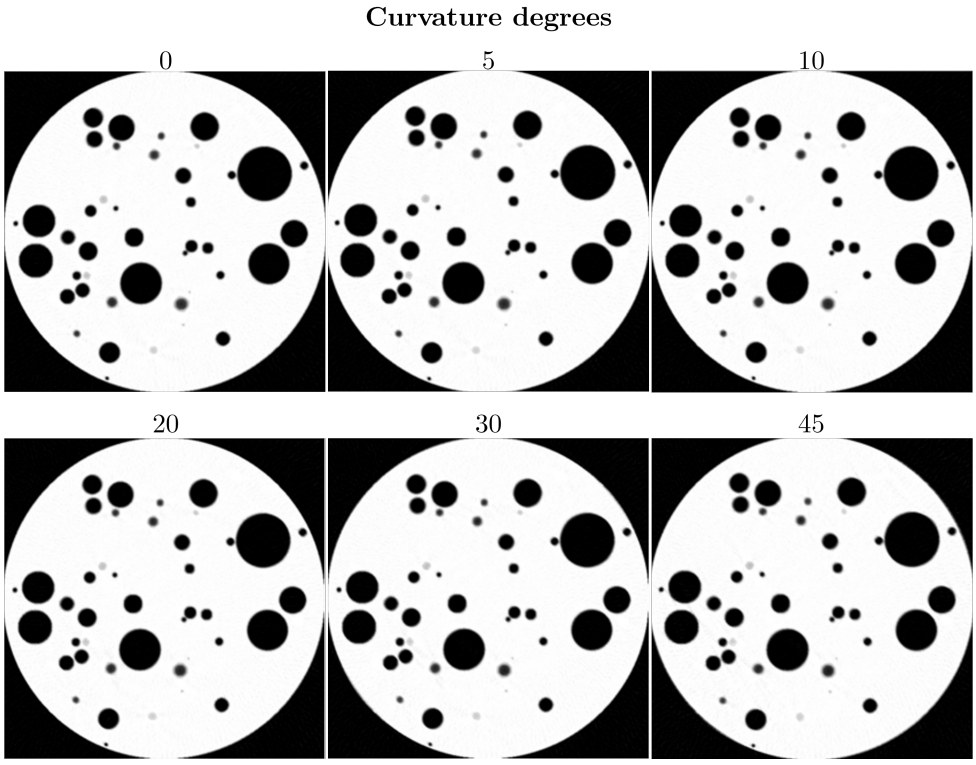


Figure 31: Reconstruction comparison with different detector curvature angles

6.9 Experiment 6: Denoising

The LUMC staff noted that it would be useful to remove the noise visible in the real data reconstructions. As discussed in Section 4, there exist various methods to effectively remove noise in CT imaging. The most reliable methods use DL techniques which normally require a good quality dataset. As we did not dispose of such dataset, a DL technique called Noise2Inverse [23] was used. As explained in Section 4.3, this uses the image itself as a dataset in order to train a CNN.
As explained in Section 5.1.1, Poisson noise was used to model the typical CT imaging noise and was added to the projections, to then make use of the Noise2Inverse algorithm to remove the modeled noise.
The Noise2Inverse algorithm was implemented in the multiple detector simulation, as if there were no issues with that geometry, there would not be any issues with a single detector. The settings used for the experiment were therefore a detector matrix of 1×5 to reduce the projection number. The Noise2Inverse does not remove artifacts, thus the separation between detectors was set to 0 in both directions and the curvature to 0. The number of projections was set to 800 and the number of iterations to 300 as these were the optimal values found in Section 6.4.

6.9.1 Training and Optimization

A first hyperparameter tuning was performed to the Noise2Inverse to determine the best settings for the denoising. The Noise2Inverse algorithm was trained with the U-Net [35] and DnCNN [36] architectures, with both strategies ($1 : X$ and $X : 1$) and different learning rates.
The strategies $1 : X$ meant that the CNN used one subset as input and the rest of the subsets were averaged and acted as target. The contrary happened with $X : 1$ strategy as all subsets except one were averaged and used as input, whilst the remaining subset was used as target.
Among the hyperparameters used for training, the number of subsets, also called K splits, was set to 4 as the Noise2Inverse paper suggested. It was determined after some testing that 100 epochs were enough for convergence. The batch size which was set to 8, the U-Net features were set to 64, and the DnCNN layers to 20. All three of them were set as high as possible, limited only by the hardware used. The remaining hyperparameters were the aforementioned strategy, network architecture, and learning rate, which were used for the hyperparameter tuning.
PSNR and SSIM results with the different settings are shown in Table 2. The metrics were computed using the phantom as the target image. In the table, two main observations can be made: the DnCNN architecture generally outperforms the U-Net architecture with a higher PSNR and SSIM, and the $X : 1$ strategy surpassed the $1 : X$ strategy in most cases.
The best results were obtained with the DnCNN architecture using the $X : 1$ strategy and a learning rate of 0.01, which yielded a PSNR of $21.68dB$ and an SSIM of 0.912.

6.9.2 N2I Results

Finally, Figure 32 shows the comparison between the phantom, the noisy reconstruction, and the denoised reconstruction with the best hyperparameters for Noise2Inverse found in

Network	**Strategy**	**LR**	**PSNR**	**SSIM**
U-Net	1:X	0.0001	2.08	0.213
U-Net	1:X	0.0005	4.00	0.456
U-Net	1:X	0.001	9.91	0.801
U-Net	1:X	0.005	21.36	0.869
U-Net	1:X	0.01	21.23	0.854
U-Net	X:1	0.0001	2.08	0.270
U-Net	X:1	0.0005	4.29	0.492
U-Net	X:1	0.001	10.90	0.784
U-Net	X:1	0.005	21.32	0.846
U-Net	X:1	0.01	19.48	0.727
DnCNN	1:X	0.0001	9.13	0.786
DnCNN	1:X	0.0005	21.48	0.902
DnCNN	1:X	0.001	21.23	0.871
DnCNN	1:X	0.005	21.25	0.895
DnCNN	1:X	0.01	20.45	0.888
DnCNN	X:1	0.0001	8.24	0.745
DnCNN	X:1	0.0005	21.68	0.894
DnCNN	X:1	0.001	21.07	0.891
DnCNN	X:1	0.005	21.37	0.896
DnCNN	X:1	0.01	**21.68**	**0.912**

Table 2: Noise2Inverse PSNR and SSIM values for U-Net and DnCNN

Table 2. It can clearly be seen that the N2I effectively removed the noise whilst maintaining fine details, and the overall structure of the phantom is maintained.

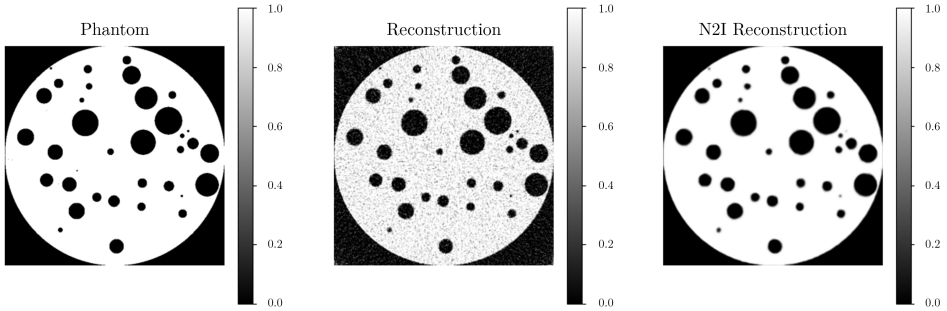
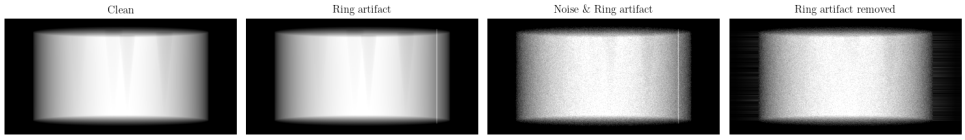


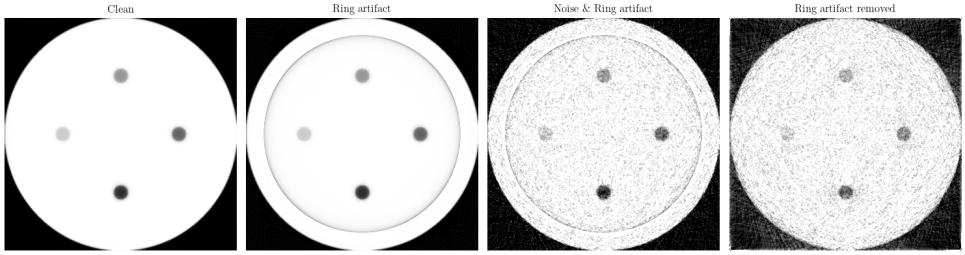
Figure 32: Reconstruction comparison for denoising

6.10 Experiment 7: Artifact removal

As observed in Section 3, there are existing artifacts in the real data reconstructions. One of the objectives of this thesis was to effectively eliminate such artifacts.
Given the results in the previous experiments (Section 6.7) where a reconstruction without artifacts was demonstrated for the multiple detector case, the artifact removal was only performed in the one detector simulation. As the case study was one big flat panel detector, pre-processing filtering methods were used as explained in Section 5.1.1.
The pipeline followed generated the projections, added the ring artifact and Poisson noise, and removed the artifact. As for the noise, experiment 6 in Section 6.9 demonstrates how it can be effectively removed. Therefore, in this experiment, only the artifact was studied. The artifact was added by applying an offset in the projections in the shape of a line, such as the one seen in Figure 33a. It was then removed by using Vo's [28] filtering techniques, which make use of different combinations of filters to remove different types of ring artifacts. Noise was added as the algorithm only works with noised data.



(a) Projections



(b) Reconstructions

Figure 33: Artifact removal

Figure 33b showcases the reconstructions of every step in the process. It shows how the algorithm effectively removes the artifact without adding too much noise.
In short, these artifact removal techniques were effective in removing the ring artifacts within the geometry of the nu:view machine, demonstrating that this pre-processing algorithm could be applied as an extra step in the machine's reconstruction pipeline.

6.11 Real Data

This project always had the intention of applying findings and algorithms to the real data. However, mainly due to time constraints, this was ultimately not possible. Nevertheless, some advances were made.

6.12 Data & Acquisition

There were three different datasets in this project, one provided by the LUMC, and the other two provided by the AB-CT team. The first dataset contained the projections of a full scan of a patient acquired from the nu:view machine at the LUMC. The second and third datasets contained the projections of a testing phantom used by the AB-CT team scanned with their nu:view machine, which also had a reconstruction of a phantom made from the third dataset projections (Figure 34).
The second and third datasets differed in the pre-processing steps (explained in Section 3) applied to the projections. The second dataset had only perspective correction, whilst the third dataset had applied both perspective correction and interpolation.

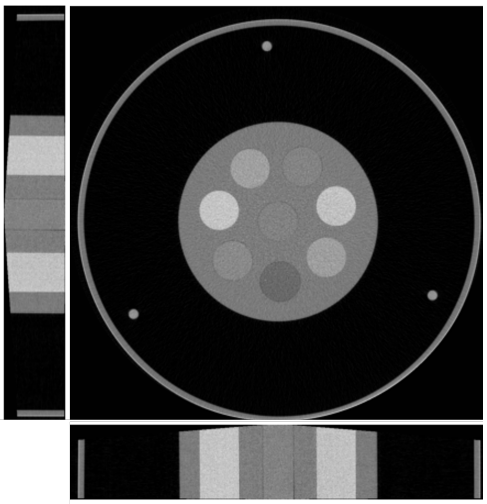


Figure 34: AB-CT's calibration phantom from top and side views

6.13 Reconstruction & Limitations

In order to apply the solutions found in the simulation environments, a clean reconstruction of the data had to first be performed.
For clarity, the scanner pipeline is as follows: Once the projections are acquired, a perspective correction is applied to create a virtual big flat panel detector. Then, an interpolation is performed to remove the grid-like shape resulting from the perspective correction with the multiple detectors, and finally, the FBP algorithm is applied to reconstruct the image.
First, as seen in Figure 14 in Section 3, a projection image of the AB-CT's phantom, the darkest part is in the middle (where the cylinder is located). This means that the data is represented in photon domain, the number of counts on the detector. Thus, the first step is to convert the data into linear projections that can be used for reconstruction. This is done by applying Beer-Lambert's law ([7] Chapter 4) as shown in Equation 13.

$$\mu l = \ln \frac{N_i n}{N_d} \tag{13}$$

Where μ is the linear attenuation coefficient, l is the thickness of the object, $N_i n$ is the number of counts on the detector, and N_d is the number of counts on the detector without the object.

This allowed us to perform a reconstruction with the FBP algorithm using the data from the third dataset. Figure 35 shows the reconstruction results, where a clear shakiness can be seen around the edges of the object in the reconstruction, and the details (smaller cylinders) are not properly defined.

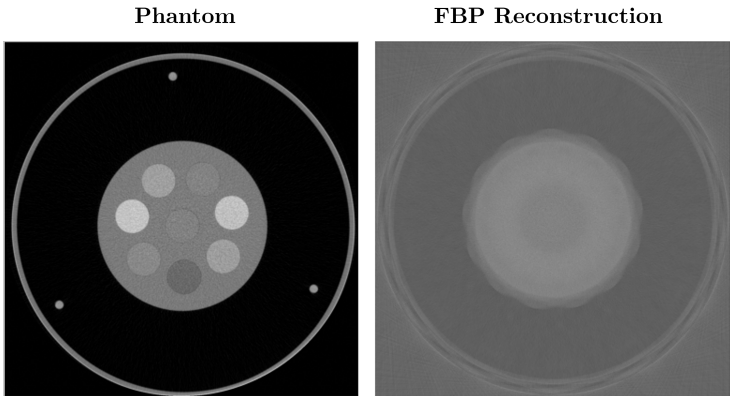


Figure 35: FBP reconstruction of the AB-CT's phantom

After looking more in depth at the projections, we saw that the center of rotation was not the center of the cylinder. This was most likely the cause of the shakiness in the reconstruction. Instead of adjusting the geometry vectors to account for the shakiness, a faster solution slightly modified the projections to center the cylinder in the middle. This ensured that the center of rotation would match with the center of the cylinder, thus removing the shakiness. The results of this correction can be seen in Figure 36, where the shakiness was removed. Nevertheless, even though the outline of the phantom was well-defined, the smaller cylinders inside the phantom were still not visible.

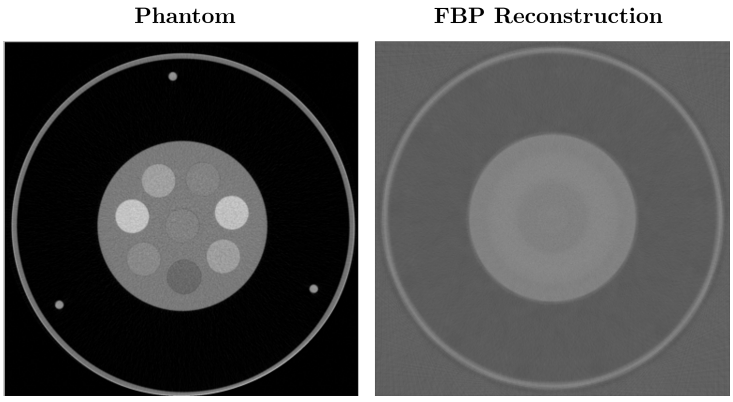


Figure 36: FBP reconstruction of the AB-CT's phantom with center correction

Unfortunately, due to the complexity of the reconstruction and time and knowledge constraints, it was difficult to quickly apply necessary solutions to this reconstruction problem.

Ultimately, the reconstruction of the real data was not fully implemented.

7 Conclusions & Future Work

The main objective of this thesis was to study the artifacts present in the nu:view machine and propose a solution for their elimination. The artifacts were studied by analyzing the LUMC's data and simulating the geometry in two different ways; first with a big flat panel detector to perform post-processing techniques like artifact removal, and then a simulation featuring the multiple detectors of the nu:view machine, where the origin of the artifacts could be studied and denoising could be performed.
The results obtained in the experiments demonstrated that the artifacts were caused by the detector geometry, specifically by the separation of the detectors.
The proposed solution was to use pre-processing filtering techniques to remove the appearing artifacts. This was performed in the flat panel detector simulation where the ring artifact was successfully removed without introducing extra noise.
Additionally, a new reconstruction algorithm was implemented. This algorithm used SIRT, that due to its linear nature, was able to reconstruct the object even with the multiple detectors thus removing the pre-processing steps of perspective correction and interpolation performed by nu:view machine. This method ensured that if the separation of the detectors was solved, it would not induce any artifacts, even if there was overlapping. Ultimately, this solved the artifact problem without the need of post-processing methods or complex interpolation.
Furthermore, Noise2Inverse was used to effectively remove the noise even in the case of reconstruction with multiple detectors. This method does not require any dataset, therefore making it available for any reconstruction.
In summation, the origin of the artifacts was identified and two different solutions were proposed. One solution eliminated the artifacts using pre-processing that could fit in the actual pipeline of the nu:view machine with little computational cost, and the other changed the reconstruction algorithm for a linear iterative method, completely removing the appearance of artifacts in the reconstruction provided that the separation between detectors is solved. The only downside of the latter method is that it is more computationally taxing in comparison to the current FBP algorithm used in the nu:view machine.
Future work could focus on the implementation of the two suggested solutions using real data. This would give insight of how these methods would behave with more complex phantoms and high resolution, studying their feasibility for the nu:view machine.

References

[1] W.J. Palenstijn, K.J. Batenburg, and J. Sijbers. "Performance improvements for iterative electron tomography reconstruction using graphics processing units (GPUs)". In: *Journal of Structural Biology* 176.2 (2011), pp. 250–253. ISSN: 1047-8477. DOI: https://doi.org/10.1016/j.jsb.2011.07.017. URL: https://www.sciencedirect.com/science/article/pii/S1047847711002267.

[2] Marta Licata et al. "Diagnosis of multiple osteomas in an ancient skeleton discovered in the necropolis of Caravate-Northern Italy". In: *Eur J Oncol* 21.4 (2016), pp. 238–42.

[3] Armen R Kherlopian et al. "A review of imaging techniques for systems biology". In: *BMC systems biology* 2 (2008), pp. 1–18.

[4] Leonardo De Chiffre et al. "Industrial applications of computed tomography". In: *CIRP annals* 63.2 (2014), pp. 655–677.

[5] Polad M Shikhaliev, Tong Xu, and Sabee Molloi. "Photon counting computed tomography: concept and initial results". In: *Medical physics* 32.2 (2005), pp. 427–436.

[6] Martin J Willemink et al. "Photon-counting CT: technical principles and clinical prospects". In: *Radiology* 289.2 (2018), pp. 293–312.

[7] Avinash C Kak and Malcolm Slaney. *Principles of computerized tomographic imaging*. SIAM, 2001.

[8] Sigurdur Helgason and S Helgason. *The radon transform*. Vol. 2. Springer, 1999.

[9] Sanni Siltanen. *Siltanen Homepage*. URL: https://siltanen-research.net/.

[10] Kees Joost Batenburg. *Joost Batenburg's Google Scholar profile*. URL: https://siltanen-research.net/.

[11] Richard Gordon, Robert Bender, and Gabor T Herman. "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography". In: *Journal of theoretical Biology* 29.3 (1970), pp. 471–481.

[12] Peter Gilbert. "Iterative methods for the three-dimensional reconstruction of an object from projections". In: *Journal of theoretical biology* 36.1 (1972), pp. 105–117.

[13] F Edward Boas, Dominik Fleischmann, et al. "CT artifacts: causes and reduction techniques". In: *Imaging Med* 4.2 (2012), pp. 229–240.

[14] JCE Mertens, JJ Williams, and Nikhilesh Chawla. "A method for zinger artifact reduction in high-energy x-ray computed tomography". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 800 (2015), pp. 82–92.

[15] Jiayang Shi, Daniël Maria Pelt, and Kees Joost Batenburg. "Multi-stage deep learning artifact reduction for computed tomography". In: *arXiv preprint arXiv:2309.00494* (2023).

[16] Arwa Alhadi Omer et al. "Denoising ct images using median based filters: a review". In: *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*. IEEE. 2018, pp. 1–6.

[17] Anandbabu Gopatoti et al. “Image denoising using spatial filters and image transforms: A review”. In: *International Journal for Research in Applied Science & Engineering Technology (IJRASET)* (2018).

[18] Zhoubo Li et al. “Adaptive nonlocal means filtering based on local noise level for CT denoising”. In: *Medical physics* 41.1 (2014), p. 011908.

[19] Ahmed M Hasan et al. “Denoising low-dose CT images using multiframe blind source separation and block matching filter”. In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 2.4 (2018), pp. 279–287.

[20] Nghia T Vo, Robert C Atwood, and Michael Drakopoulos. “Superior techniques for eliminating ring artifacts in X-ray micro-tomography”. In: *Optics express* 26.22 (2018), pp. 28396–28412.

[21] Jan Sijbers and Andrei Postnov. “Reduction of ring artefacts in high resolution micro-CT reconstructions”. In: *Physics in Medicine & Biology* 49.14 (2004), N247.

[22] Hu Chen et al. “Low-dose CT with a residual encoder-decoder convolutional neural network”. In: *IEEE transactions on medical imaging* 36.12 (2017), pp. 2524–2535.

[23] Allard Adriaan Hendriksen, Daniël Maria Pelt, and K Joost Batenburg. “Noise2inverse: Self-supervised deep convolutional denoising for tomography”. In: *IEEE Transactions on Computational Imaging* 6 (2020), pp. 1320–1335.

[24] Xiaojie Lv et al. “Image denoising and ring artifacts removal for spectral CT via deep neural network”. In: *IEEE access* 8 (2020), pp. 225594–225601.

[25] Maxime Nauwynck et al. “Ring artifact reduction in sinogram space using deep learning”. In: *The 6th International Conference on Image Formation in X-Ray Computed Tomography, 3-7 August, 2020, Regensburg, Germany*. 2020, pp. 486–489.

[26] Wim van Aarle et al. “The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography”. In: *Ultramicroscopy* 157 (2015), pp. 35–47. ISSN: 0304-3991. DOI: https://doi.org/10.1016/j.ultramic.2015.05.002. URL: https://www.sciencedirect.com/science/article/pii/S0304399115001060.

[27] Allard A Hendriksen et al. “Tomosipo: fast, flexible, and convenient 3D tomography for complex scanning geometries in Python”. In: *Optics Express* 29.24 (2021), pp. 40494–40513.

[28] Nghia T Vo, Robert C Atwood, and Michael Drakopoulos. “Superior techniques for eliminating ring artifacts in X-ray micro-tomography”. In: *Optics express* 26.22 (2018), pp. 28396–28412.

[29] Daniël M Pelt, Allard A Hendriksen, and Kees Joost Batenburg. “Foam-like phantoms for comparing tomography algorithms”. In: *Synchrotron Radiation* 29.1 (2022), pp. 254–265.

[30] Daniel M. Pelt. *A Python package for generating foam-like phantoms for CT*. 2019. URL: https://github.com/dmpelt/foam_ct_phantom/.

[31] Zhou Wang and Alan C Bovik. “Mean squared error: Love it or leave it? A new look at signal fidelity measures”. In: *IEEE signal processing magazine* 26.1 (2009), pp. 98–117.

[32] Anil K Jain. *Fundamentals of digital image processing*. Prentice-Hall, Inc., 1989.

[33] Zhou Wang et al. “Image quality assessment: from error visibility to structural similarity”. In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.

[34] Jiawei Han, Micheline Kamber, and Jian Pei. *Data Mining: Concepts and Techniques*. 3rd. Waltham, MA: Morgan Kaufmann, 2011. ISBN: 978-0123814791.

[35] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer. 2015, pp. 234–241.

[36] Kai Zhang et al. “Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising”. In: *IEEE transactions on image processing* 26.7 (2017), pp. 3142–3155.