Degree Project in Technology
First cycle, 15 credits

## Fine-Tuning Parameters in CT

## ANTON ADELÖW, TOMAS NORDSTRÖM


#### Abstract

Computed tomography (CT) is a medical imaging technique that uses X-rays to obtain a reconstruction of an object. The term acquisition geometry refers to the arrangement of imaging sensors and the X-ray source as well as the procedure used for data collection. The quality of the reconstruction is often limited by the acquisition geometry and parameter values. In this thesis, we present a procedure for fine-tuning acquisition geometry parameters in CT by minimizing the difference between the forward projection of a known phantom and measured data, i.e. data discrepancies. We extend the ODL library in Python and create acquisition geometries where different parameters have been distorted. We utilize gradient descent in an attempt recover the true parameters of the acquisition geometries. Our results show that the recovery of the true geometry is successful when one or, in some cases, two parameters are perturbed. The objective function becomes very sensitive when more parameters are perturbed, requiring a low learning rate and making convergence slow. Nevertheless, we are able to minimize the objective function in the forward projection for all perturbations. Although our algorithm performs well in some aspects relating to parameter recovery, there is potential for further research by implementing other optimization methods.


## Contents

1 Introduction ..... 4
2 Materials and Methods ..... 4
2.1 Beer's Law ..... 4
2.2 Forward Projection ..... 5
2.3 Acquisition Geometries in CT ..... 6
2.4 Geometry Discretization ..... 10
2.5 Data Discrepancy Minimization ..... 10
2.6 Differentiation ..... 11
2.7 Convexity ..... 13
2.8 Implementation ..... 13
3 Results ..... 14
4 Discussion ..... 17
5 Acknowledgments ..... 20

## 1 Introduction

Computed tomography (CT) is a technique for medical imaging that uses data obtained by X-rays to produce a recreation of the scanned object. CT scans can be used to determine medical diagnosis for conditions such as cancer, damage to bones or internal organs, or problems with blood flow. Historically, there have existed a variety of setups, or so called acquisition geometries, to obtain the required data. The term acquisition geometry refers to the arrangement of imaging sensors as well as the procedure used for data collection. Meaning, the process of configuring the position angles for the source, as well as the pitch and shift values for both the source and detector. Researchers working on improving reconstruction quality in CT often have limited access to the information about the acquisition geometry and the parameter values are known only approximately. This can happen because the scanner manufacturer either did not disclose or did not measure the exact values. For instance, the source position might be hard to measure due to the speed of the rotation. Therefore, this thesis aims to develop a procedure for fine-tuning parameters of the acquisition geometry by minimizing the difference between the forward projection of a known phantom and measured data, i.e. data discrepancies. Furthermore, successful realization of this work creates an opportunity to include the geometry parameters in the set of learned parameters in deep learning-based reconstruction algorithms.

## 2 Materials and Methods

### 2.1 Beer's Law

We start with defining the volume density of particles at $\mathbf{x}$ as $N(\mathbf{x})$ and let $\mathbf{v}_{\phi}$ be their velocity field. Thus, we denote the flux of photons through a surface $S$ as

$$
\begin{equation*}
\Phi(\mathbf{x}):=\oiint_{S} \mathbf{v}_{\phi} N(x) d S \tag{1}
\end{equation*}
$$

where $S$ is the area of the surface. The flux of photon density represents the number of photons passing through a given surface per unit of time and per unit area. In this case, it specifically refers to the flow of X-ray photons through the surface S. By the definition of the attenuation constant from ISO [1989],

$$
\begin{equation*}
\mu=-\frac{1}{\Phi} \frac{\partial \Phi}{\partial \ell} \tag{2}
\end{equation*}
$$

where the attenuation constant quantifies the absorptive and scattering properties of the medium and relates the change in flux to the spatial variation of the particle density. Here, $\partial \ell$ is a short segment of the line $\ell$ which the photons are propagating along. However, as we are looking at different positions in
our medium with varying attenuation constants we are going to have a spacedependent attenuation constant defined as $f(\mathbf{x})$. With this, we get the following relationship

$$
\begin{equation*}
\partial \Phi(\mathbf{x})=-f(\mathbf{x}) \Phi(\mathbf{x}) \partial \ell \tag{3}
\end{equation*}
$$

This can be interpreted as the change in flux when the X-ray travels over a short segment $\partial \ell$ of the line $\ell$. The change in flux will depend on the attenuation coefficient in medium of $\partial \ell$. Recall that $\Phi$ is the flux of photons through an area $S$, thus by multiplying the flux with the frequency of the radiation $v$ and Planck's constant, denoted $h$, we get the radiation power $P=h v \Phi$. Now by dividing the radiation power by the transversal area of the beam, we get the intensity $I=\frac{P}{A}$. Thus, we can rewrite our expression as

$$
\begin{equation*}
\frac{\partial I(\mathbf{x})}{I(\mathbf{x})}=-f(\mathbf{x}) \partial \ell \tag{4}
\end{equation*}
$$

Since both sides depend on the position $\mathbf{x}$ in space and the X-ray propagates along the line $\ell$ which starts at $\mathbf{x}_{0}$ and ends at $\mathbf{x}_{1}$, we derive the following relationship

$$
\begin{equation*}
\frac{\partial I(\mathbf{x})}{I(\mathbf{x})}=-f(\mathbf{x}) \partial \ell \Longrightarrow \ln \left(I\left(\mathbf{x}_{1}\right)\right)-\ln \left(I\left(\mathbf{x}_{0}\right)\right)=\int_{\ell}-f(\mathbf{x}) d \ell \tag{5}
\end{equation*}
$$

Therefore, we have

$$
\begin{equation*}
\ln \left(\frac{I\left(\mathbf{x}_{1}\right)}{I\left(\mathbf{x}_{0}\right)}\right)=\int_{\ell}-f(\mathbf{x}) d \ell \Longleftrightarrow \ln \left(\frac{I_{1}}{I_{0}}\right)=-\int_{\ell} f(\mathbf{x}) d \ell \tag{6}
\end{equation*}
$$

An explanation for the intensity relationship can be given as follows: since the intensity decreases exponentially as the line passes through the object and the sum of the attenuation along a line can be interpreted as a line integral, the difference between the logarithm of the intensity measured at the detector and the logarithm of the intensity measured at the source will be equal to the integral of the (negative) attenuation at all points $\mathbf{x}$ along the line propagating through the object.

### 2.2 Forward Projection

In the context of CT, the real-valued function $f$ will be the (linear) attenuation coefficient. This relates to the information collected from the scan as described by the equation in (6). The line integral is dependent on the line being integrated over and thus, to understand how the geometry is dependent on the parameters, we must formulate equation (6) with explicit dependence on the parameters of the geometry. The ray transform can be defined as

$$
\begin{equation*}
\mathcal{A} f(\ell)=\int_{-\infty}^{\infty} f(\mathbf{x}) d \ell \tag{7}
\end{equation*}
$$

Since the attenuation constant is zero outside the scanned object, the following relationship holds

$$
\begin{equation*}
\ln \left(\frac{I_{1}}{I_{0}}\right)=-\mathcal{A}(f(\ell)) \tag{8}
\end{equation*}
$$

To apply the ray transform to a set of lines passing through the 3D body we use the forward operator $A f(\mathbf{x})$, where we have that

$$
\begin{equation*}
A f(\mathbf{x})=\mathbf{Y} \tag{9}
\end{equation*}
$$

where $f(\mathbf{x})$ represents the attenuation constant at each position in the 3D body $\mathbf{x}$ and $\mathbf{Y}$ is the data collected by the scanner Gündüzalp et al. [2021]. In summary, the ray transform refers to the integration of the attenuation coefficient function $f(\mathbf{x})$ along a ray, while the forward operator $A f(\mathbf{x})$ represents the application of the ray transform to a set of lines passing through the 3D body.

As we scan from multiple angles we obtain a sinogram, which is a projection of the image from all angles Kumar et al. [2010]. An example of this can be seen in figure 1, where we have the angles on the x-axis, the position on the detector on the $y$-axis and the values produced by the forward projection represented using the color of the sinogram and the color bar.


Figure 1: A 2D slice of a sinogram of the Shepp-Logan phantom scanned using a helical path cone beam geometry.

### 2.3 Acquisition Geometries in CT

There are various geometries used in computed tomography for different purposes. One of them is the parallel scanning geometry, where the object is scanned one position at a time, moving in a straight line until a slice is fully scanned. Then, the angular position is changed and the process is repeated until the entire object is scanned. A visualisation of the parallel scanning geometry


Figure 2: Parallel beam geometry Maier et al. [2018]
can be seen in figure 2. However, this method is highly inefficient Natterer and Wübbeling [2001]. To derive the integrating paths for the different acquisition geometries we parameterize the lines. Note that in ODL, the Python library used to simulate the geometries in the report, the initial position for the source is at the negative $y$-axis and the rotation is counter clock-wise. Thus, when parameterizing, the initial position for the source is assumed to be the at the negative y-axis and the rotation counter clock-wise. For parallel beam geometry each line can be represented by

$$
\begin{equation*}
\ell(t)=\mathbf{u}(u, \theta)+\mathbf{v}(\theta) \cdot t \tag{10}
\end{equation*}
$$

where

$$
\begin{align*}
& \mathbf{u}=(u \cdot \cos (\theta), u \cdot \sin (\theta))  \tag{11}\\
& \mathbf{v}=(-\sin (\theta), \cos (\theta)) \tag{12}
\end{align*}
$$

where $\mathbf{u}$ is the offset at the detector from its middle point and $\mathbf{v}$ is orthogonal to $\mathbf{u}$ and is the vector pointing from the source to the detector. $\theta$ is the angle between the initial position of the source and the current position and $u$ is the distance between the center of the detector and the location where the line hits the detector. The ray transform $\mathcal{A}$ of $f$ for the parallel beam geometry becomes

$$
\begin{equation*}
\mathcal{A} f(u, \theta)=\int_{-\infty}^{\infty} f(\mathbf{u}(u, \theta)+\mathbf{v}(\theta) \cdot t) d t \tag{13}
\end{equation*}
$$



Figure 3: Fan beam geometry Maier et al. [2018]

Another choice of geometry is the fan beam geometry, where the scanner and detector are mounted at opposite sides in a rotating configuration. A visualisation of the fan beam geometry can be seen in figure 3. This geometry enables faster switching between relevant positions while maintaining a similar coverage to that of the parallel beam Kumar and Singh [2013]. For the fan beam geometry, we parameterize the lines using the vectors

$$
\begin{align*}
& \mathbf{a}(u, \theta)=\left(-R_{d} \cdot \sin (\theta), R_{d} \cdot \cos (\theta)\right)+(u \cdot \cos (\theta), u \cdot \sin (\theta))  \tag{14}\\
& \mathbf{b}(u, \theta)=\mathbf{a}(u, \theta)-\left(R_{s} \cdot \sin (\theta),-R_{s} \cdot \cos (\theta)\right) \tag{15}
\end{align*}
$$

Where $R_{d}$ and $R_{s}$ represent the radius to the detector and the radius to the source, respectively. The vector a represents the line from the origin to the detector point where the line ends and $\mathbf{b}$ is a vector pointing in the direction of the source from the point of the detector where a points at. Rotation is assumed to be counterclockwise and the starting position of the source is assumed to be at the negative $y$-axis. The ray transform $\mathcal{A}$ of $f$ for the fan beam geometry becomes

$$
\begin{equation*}
\mathcal{A} f(u, \theta)=\int_{-\infty}^{\infty} f\left(\mathbf{a}(u, \theta)+\mathbf{b}_{N}(u, \theta) \cdot t\right) d t \tag{16}
\end{equation*}
$$

where $\mathbf{b}_{N}(u, \theta)$ is the normalization of $\mathbf{b}(u, \theta)$, i.e.

$$
\begin{equation*}
\mathbf{b}_{N}(u, \theta):=\frac{\mathbf{b}(u, \theta)}{\|\mathbf{b}(u, \theta)\|} \tag{17}
\end{equation*}
$$



Figure 4: Conebeam geometry with helical path Kudo et al. [2004]

Finally, the geometry used in this thesis is helical path cone beam geometry, which captures data points in three dimensions instead of two, like the fan beam. This allows for gathering a greater amount of data at each angular position. The cone beam geometry is frequently used in dental imaging. The cone-beam geometry is able to generate high-resolution images at the risk of scattered radiation but works well for smaller objects Scarfe and Farman [2008].

In 3D, an extension to the scanning geometry is the implementation of a helical path, known as 3D helical scanning. Instead of scanning slice by slice, the scanner moves in a helical path along the body's axis. This is visualized in figure 4 . Helical cone beam geometries are commonly used in medical imaging of internal organs, bones, soft tissue, etc Lechuga and Weidlich [2016].

For the cone beam geometry we parameterize the lines in a similar way as for the fan beam geometry. A line in 3-dimensional space can be parameterized as

$$
\begin{equation*}
\ell(t)=\mathbf{a}(u, v, \theta)+\mathbf{b}_{N}(u, v, \theta) \cdot t \tag{18}
\end{equation*}
$$

where $\theta$ is the angle between the initial position of the source and the current position in the $x y$-plane and $u$ and $v$ are coordinates corresponding to the horizontal and vertical distances from the detector center to the point where the line hits the detector, respectively. For the cone beam geometry, we parameterize
the lines using the vectors

$$
\begin{align*}
& \mathbf{a}(u, v, \theta)=\left(-R_{d} \cdot \sin (\theta), R_{d} \cdot \cos (\theta), z\right)+(u \cdot \cos (\theta), u \cdot \sin (\theta), v)  \tag{19}\\
& \mathbf{b}(u, v, \theta)=\mathbf{a}(u, v, \theta)-\left(R_{s} \cdot \sin (\theta),-R_{s} \cdot \cos (\theta), z\right) \tag{20}
\end{align*}
$$

The ray transform $\mathcal{A}$ of $f$ for the cone beam geometry becomes

$$
\begin{equation*}
\mathcal{A} f(u, v, \theta)=\int_{-\infty}^{\infty} f\left(\mathbf{a}(u, v, \theta)+\mathbf{b}_{N}(u, v, \theta) \cdot t\right) d t \tag{21}
\end{equation*}
$$

where $\mathbf{b}_{N}(u, v, \theta)$ is the normalization of $\mathbf{b}(u, v, \theta)$.
The specific geometry used in the system will decide the sampling of lines. When we scan a body of interest, there will be, in theory, an infinite number of lines we could scan by adjusting the system's geometry. Thus, altering the parameters will imply other lines being scanned and different values of the ray transform will be retrieved. This means that there will be differences in the forward projection for different geometries.

### 2.4 Geometry Discretization

The CT scan will be conducted using a limited number of angles (i.e., source positions) and a finite number of detector cells in both the horizontal and vertical directions. Consequently, the number of lines formed during the scan is also finite. Thus, given indices $i, j, k$, we partition the detector and positions in the $z$-direction as follows

$$
(u, v)=\left(\frac{(i+0.5) \cdot w}{n_{w}}-\frac{w}{2}, \frac{(j+0.5) \cdot h}{n_{h}}-\frac{h}{2}\right) \quad \text { and } \quad z_{k}=z_{0}+\frac{N_{\mathrm{rot}} \cdot \eta}{N_{\mathrm{ang}}} \cdot k
$$

where $w$ and $h$ are the width and height of the detector respectively. The variables $n_{w}$ and $n_{h}$ are the numbers of cells along the width and height of the detector, $z_{0}$ is the starting position of the source in the $z$-direction, $\eta$ is the total traversed distance in the $z$-direction per rotation (pitch), which is visualized in figure 4. $N_{\text {rot }}$ is the number of rotations performed and $N_{\text {ang }}$ is the total number of angles (source positions) in the model. Since the set of angles may not be uniform, the angle partition is a finite set of angles, where $\theta_{k}$ simply is the $k$ :th element in the partition. The partition in the $z$-direction is assumed to be uniform.

### 2.5 Data Discrepancy Minimization

In a realistic scenario, the parameters of the acquisition geometry will only be known approximately. This can be because the scanner manufacturer did not measure or disclose the exact parameter values. We assume, therefore, that
we know the values approximately. We define the forward operator for the distorted geometry as $A_{p} f(\boldsymbol{x})$. The data observed from CT scan with this geometry, $A_{p} f(\boldsymbol{x})$, will be a distorted version of the data generated with the true parameters, Y. Thus, to recover the parameters of the true geometry, we wish to solve the following problem

$$
\begin{equation*}
\underset{p}{\operatorname{argmin}}\left\|A_{p} f(\boldsymbol{x})-\mathbf{Y}\right\|^{2} \tag{22}
\end{equation*}
$$

To find the parameters that minimize the objective function (22) we use the method of gradient descent. Let $L(p)=\left\|A_{p} f(\boldsymbol{x})-\mathbf{Y}\right\|^{2}$. Then, minimizing $L(p)$ is done by, at each iteration, finding the $\Delta p$ minimizing

$$
\begin{equation*}
L(p+\lambda \Delta p) \approx L(p)+\lambda \Delta p^{T} \nabla L(p) \tag{23}
\end{equation*}
$$

for some, sufficiently small, step size $\lambda$. As we know, (23) is minimized by letting $\Delta p$ be equal to the negative of the gradient. Thus, we set

$$
\begin{equation*}
p_{i}^{n+1}=p_{i}^{n}-\lambda \frac{\partial}{\partial p_{i}}\left\|A_{p} f(\boldsymbol{x})-\mathbf{Y}\right\|^{2} \tag{24}
\end{equation*}
$$

where $p_{i}^{n}$ is the i:th parameter at iteration $n$ and $\lambda$ is the step size we take at each iteration.

### 2.6 Differentiation

We define $\mathbf{Y}$ and $A_{p} f(\mathbf{x})$ as tensors in $\mathbb{R}^{n_{\theta} \times n_{d w} \times n_{d h}}$, where we define both tensors as a stack of matrices where each matrix represents a specific angle in the model and will be of the same dimensions as the detector screen. Thus, we get

$$
\begin{equation*}
\frac{\partial}{\partial p_{l}} \frac{1}{2}\left\|A_{p} f(\boldsymbol{x})-\mathbf{Y}\right\|^{2}=\sum_{i, j, k}\left(A_{p}^{i, j, k} f(\boldsymbol{x})-\mathbf{Y}^{i, j, k}\right) \frac{\partial}{\partial p_{l}} A_{p}^{i, j, k} f(\boldsymbol{x}) \tag{25}
\end{equation*}
$$

We have

$$
A_{p}^{i, j, k} f(\mathbf{x})=\int_{-\infty}^{\infty} f\left(\mathbf{a}^{i, j, k}(p)+\mathbf{b}_{N}^{i, j, k}(p) \cdot t\right) \mathrm{d} t
$$

so by the Leibniz rule, we get

$$
\begin{align*}
& \frac{\partial}{\partial p_{l}} \int_{L_{1}}^{L_{2}} f\left(\mathbf{a}^{i, j, k}(p)+\mathbf{b}_{N}^{i, j, k}(p) \cdot t\right) \mathrm{d} t \\
&=f\left(\mathbf{a}^{i, j, k}(p)+\mathbf{b}_{N}^{i, j, k}(p) \cdot L_{2}\right) \frac{\partial L_{2}}{\partial p_{l}}-f\left(\mathbf{a}^{i, j, k}(p)+\mathbf{b}_{N}^{i, j, k}(p) \cdot L_{1}\right) \frac{\partial L_{1}}{\partial p_{l}} \\
&+\int_{L_{1}}^{L_{2}} \frac{\partial}{\partial p_{l}} f\left(\mathbf{a}^{i, j, k}(p)+\mathbf{b}_{N}^{i, j, k}(p) \cdot t\right) \mathrm{d} t \tag{26}
\end{align*}
$$

As the limits $L_{2}$ and $L_{1}$ will be independent of $p,(26)$ gives us

$$
\begin{equation*}
\frac{\partial}{\partial p_{l}} A_{p}^{i, j, k} f(\boldsymbol{x})=\int_{-\infty}^{\infty} \frac{\partial}{\partial p_{l}} f\left(\mathbf{a}^{i, j, k}(p)+\mathbf{b}_{N}^{i, j, k}(p) \cdot t\right) \mathrm{d} t \tag{27}
\end{equation*}
$$

When differentiating the objective function we see that it dependent on the location, which is dependent on the parameters. Meaning that, $\mathbf{a}(p)+\mathbf{b}_{N}(p) \cdot t$ represent, for some $t$, an $\mathbf{x} \in \mathbb{R}^{3}$. Thus, when differentiating with respect to a parameter $p_{i}$ we have that

$$
\begin{equation*}
\frac{\partial f}{\partial p_{i}}=\left\langle\nabla_{\mathbf{x}} f, \frac{\partial \mathbf{x}}{\partial p_{i}}\right\rangle=\left\langle\nabla_{\mathbf{x}} f, \frac{\partial\left(\mathbf{a}(p)+\mathbf{b}_{N}(p) \cdot t\right)}{\partial p_{i}}\right\rangle \tag{28}
\end{equation*}
$$

by the chain rule, where $\nabla_{\mathbf{x}} f$ is the gradient of $f$ with respect to $\mathbf{x}$. Thus, we have

$$
\begin{align*}
& \frac{\partial}{\partial p_{i}} f\left(\mathbf{a}(p)+\mathbf{b}_{N}(p) \cdot t\right)=  \tag{29}\\
& \quad\left\langle\nabla_{\mathbf{x}} f\left(\mathbf{a}(p)+\mathbf{b}_{N}(p) \cdot t\right),\left(\frac{\partial}{\partial p_{i}} \mathbf{a}(p)+\frac{\partial}{\partial p_{i}} \mathbf{b}_{N}(p) \cdot t\right)\right\rangle \tag{30}
\end{align*}
$$

and

$$
\begin{align*}
& \frac{\partial}{\partial p_{l}} A_{p}^{i, j, k} f(\boldsymbol{x})=\frac{\partial \mathbf{a}^{i, j, k}(p)}{\partial p_{l}} \int_{-\infty}^{\infty} \nabla_{x} f\left(\mathbf{a}^{i, j, k}(p)\right. \\
& \left.\quad+\mathbf{b}_{N}^{i, j, k}(p) \cdot t\right) \mathrm{d} t+\frac{\partial \mathbf{b}_{N}^{i, j, k}(p)}{\partial p_{l}} \int_{-\infty}^{\infty} \nabla_{x} f\left(\mathbf{a}^{i, j, k}(p)+\mathbf{b}_{N}^{i, j, k}(p) \cdot t\right) \cdot t \mathrm{~d} t \tag{31}
\end{align*}
$$

We are now ready to start differentiating with respect to the parameters. As $\mathbf{b}_{N}(p)$ is a normalized vector defined by $\mathbf{b}_{N}=\mathbf{b} /\langle\mathbf{b}, \mathbf{b}\rangle^{\frac{1}{2}}$, we have the following derivative for the inverse norm

$$
\begin{align*}
\frac{\partial}{\partial p}\langle\mathbf{b}, \mathbf{b}\rangle^{-\frac{1}{2}} & =-\frac{1}{2}\langle\mathbf{b}, \mathbf{b}\rangle^{-\frac{3}{2}} \cdot\left(\left\langle\frac{\partial \mathbf{b}}{\partial p}, \mathbf{b}\right\rangle+\left\langle\mathbf{b}, \frac{\partial \mathbf{b}}{\partial p}\right\rangle\right)  \tag{32}\\
& =-\langle\mathbf{b}, \mathbf{b}\rangle^{-\frac{3}{2}} \cdot\left\langle\frac{\partial \mathbf{b}}{\partial p}, \mathbf{b}\right\rangle \tag{33}
\end{align*}
$$

This gives us

$$
\begin{equation*}
\frac{\partial}{\partial p} \mathbf{b}_{N}=\frac{\frac{\partial \mathbf{b}}{\partial p}}{\langle\mathbf{b}, \mathbf{b}\rangle^{\frac{1}{2}}}-\frac{\mathbf{b}}{\langle\mathbf{b}, \mathbf{b}\rangle^{\frac{3}{2}}} \sum_{i=1}^{3} b_{i} \cdot \frac{\partial b_{i}}{\partial p} \tag{34}
\end{equation*}
$$

Partial derivatives of $\mathbf{a}$ and $\mathbf{b}$ w.r.t. $\theta_{k}$ are then given as

$$
\begin{align*}
& \frac{\partial \mathbf{a}}{\partial \theta_{k}}=\left(-R_{d} \cdot \cos \left(\theta_{k}\right),-R_{d} \cdot \sin \left(\theta_{k}\right), 0\right) \\
& \quad+\left(-\left(\frac{\left(i+\frac{1}{2}\right) \cdot w}{n_{w}}-\frac{w}{2}\right) \cdot \sin \left(\theta_{k}\right),\left(\frac{\left(i+\frac{1}{2}\right) \cdot w}{n_{w}}-\frac{w}{2}\right) \cdot \cos \left(\theta_{k}\right), 0\right) \\
& \frac{\partial \mathbf{b}}{\partial \theta_{k}}=  \tag{35}\\
& \quad \frac{\partial \mathbf{a}(p)}{\partial \theta_{k}}+\left(-R_{s} \cdot \cos \left(\theta_{k}\right), R_{s} \cdot-\sin \left(\theta_{k}\right), 0\right)
\end{align*}
$$

The corresponding partial derivatives of $\mathbf{a}$ and $\mathbf{b}$ w.r.t. $R_{s}$ and $R_{d}$ read as

$$
\begin{align*}
\frac{\partial \mathbf{a}}{\partial R_{d}} & =\left(-\sin \left(\theta_{k}\right), \cos \left(\theta_{k}\right), 0\right) & \frac{\partial \mathbf{a}}{\partial R_{s}} & =(0,0,0)  \tag{36}\\
\frac{\partial \mathbf{b}}{\partial R_{d}} & =\left(-\sin \left(\theta_{k}\right), \cos \left(\theta_{k}\right), 0\right) & \frac{\partial \mathbf{b}}{\partial R_{s}} & =\left(-\sin \left(\theta_{k}\right), \cos \left(\theta_{k}\right), 0\right) \tag{37}
\end{align*}
$$

and those w.r.t. $w$ are given as

$$
\begin{align*}
& \frac{\partial \mathbf{a}}{\partial w}=\left(\left(\frac{i+\frac{1}{2}}{n_{w}}-\frac{1}{2}\right) \cdot \cos \left(\theta_{k}\right),\left(\frac{i+\frac{1}{2}}{n_{w}}-\frac{1}{2}\right) \cdot \sin \left(\theta_{k}\right), 0\right) \\
& \frac{\partial \mathbf{b}}{\partial w}=\left(\left(\frac{i+\frac{1}{2}}{n_{w}}-\frac{1}{2}\right) \cdot \cos \left(\theta_{k}\right),\left(\frac{i+\frac{1}{2}}{n_{w}}-\frac{1}{2}\right) \cdot \sin \left(\theta_{k}\right), 0\right) \tag{38}
\end{align*}
$$

Finally, the partial derivatives of $\mathbf{a}$ and $\mathbf{b}$ w.r.t. $h$ and $\eta$ are given as

$$
\begin{align*}
\frac{\partial \mathbf{a}}{\partial h}=\left(0,0, \frac{j+\frac{1}{2}}{n_{h}}-\frac{1}{2}\right) & \frac{\partial \mathbf{a}}{\partial \eta}=\left(0,0, \frac{N_{\mathrm{rot}}}{N_{\mathrm{ang}}} \cdot k\right)  \tag{39}\\
\frac{\partial \mathbf{b}}{\partial h}=\left(0,0, \frac{j+\frac{1}{2}}{n_{h}}-\frac{1}{2}\right) & \frac{\partial \mathbf{b}}{\partial \eta}=(0,0,0) \tag{40}
\end{align*}
$$

### 2.7 Convexity

The objective function is a norm composed of two parts: the forward projection $A_{p} f(\boldsymbol{x})$ for the perturbed geometry and $\mathbf{Y}$ the data from the forward operator for the actual geometry. Consequently, it is not immediately apparent that the objective function will be convex. Thus, further analysis is necessary to ascertain any conclusions regarding convexity.

### 2.8 Implementation

The forward operator was simulated using the Operator Discretization Library (ODL) in Python and the derivatives were tested on the Shepp-Logan phantom, using gradient descent.

For each parameter of interest, we conducted a unit test to observe convergence using the following algorithm

```
Create True Geometry
Create True Forward Projection: \(Y=\) RayTransform(Phantom)
Perturb Parameter: \(P_{D}^{0}=P \cdot(1.01)\)
for \(n=[1: N]\) do
    Create Perturbed Geometry
    Perturbed Forward Projection:
        \(A_{p} f(\boldsymbol{x})=\) RayTransform(Phantom)
    Get Gradient: \(\mathrm{Grad}=\frac{\partial A_{p} f(\boldsymbol{x})}{\partial P_{D}^{n}}\)
    \(P_{D}^{n+1}=P_{D}^{n}-\) step \(\cdot\left(A_{p} f(\boldsymbol{x})-Y\right) \cdot \operatorname{Grad}\)
end
```

Algorithm 1: Parameter Unit Test
The number of angles used in the simulations were 500 per rotation, the number of rotations were 2 and the space was partitioned from $[-170,-170,0]$ to $[170$, $170,50]$ with $[512,512,50]$ points. The detector was partitioned using 736 cells for the width and 64 cells on for the height. The parameters used for the true geometry were as follows:

- Source Radius: 600
- Detector Radius: 500
- Pitch: 25
- Detector width: 970
- Detector height: 70.

In the case when a single, or a couple of, parameters were perturbed, a $1 \%$ perturbation was added to the true parameter values, while in the case where all parameters were simultaneously perturbed, the size of the perturbation was $1 \%$ of the true parameter values. The angles were perturbed uniformly by adding a distortion to all angles of $1 \%$ in the case when a single or a couple of parameters were perturbed and adding a perturbation of $1 \%$ in the case where all parameters were simultaneously perturbed.

## 3 Results

Like this, the plots in figure 5 were generated, where the parameters have been perturbed one at a time. Thus one can observe that minimizing the objective function and recovering the true geometry, when perturbing only one parameter, is possible and is done quite smoothly. However, by perturbing several parameters at once, which is a more realistic scenario, the task of minimizing the objective function becomes more complex. This is observed in figure 7. The observed convergence exhibits non-smooth behavior, which could be attributed to the misalignment between the steepest descent direction and the direction toward the global minimum at certain points. This phenomenon is visualized evidence figure 6. It is also possible that a smaller step size would yield better


Figure 5: Plots of unit tests when perturbing each parameter one by one.


Figure 6: Image of multivariate gradient descent with zig-zag descent path.
results in this case. Solving the primary optimization task, which entails simultaneously optimizing the geometry for all relevant parameters, we obtained the the results show in figure 8. Since the plot has a logarithmic scale, our objective function is decreasing exponentially. A more detailed examination of the simulation's initial parameters during the first iteration provides a better understanding of the observed steep descent, revealing notable changes in some of the parameters. These trends are illustrated in figure 9a. During the first 30 iterations, the optimization process had an impact on the pitch and detector height parameters, while showing only minor changes in the source radius and detector width, and a slight increase in error for the detector radius. Upon allowing the algorithm to continue, further changes in error were observed for all parameters, as shown in figure 9b. Upon analyzing the error rates of each parameter throughout the entire simulation, we observe that pitch quickly approaches its actual value, while detector height exhibits a significant deviation after 200 iterations, which stabilizes after 2000 iterations. Source radius and detector width move towards their true values, albeit at a slow rate. In contrast,

(a) Plot of error in percentage of Source- and Detector Radius they are perturbed in the model with $1 \%$ for 30 iterations

(b) Plot of error in percentage of Source- and Detector Radius they are perturbed in the model with $1 \%$ for 1000 iterations

Figure 7: Plots of convergence for two parameters.


Figure 8: Plots of logarithmic error when all parameters are perturbed.
the detector radius continues to deviate throughout the simulation.
As seen in figure 12 , a $1 \%$ offset of a parameter results in varying objective function values depending on the parameter. A perturbation of detector width and pitch seem to have a substantial impact on the system error compared to other parameters. By varying the values of two parameters simultaneously, we can observe how the objective function depends on these parameters. This is shown in figure 10. Included in figure 11 are log-scaled versions of the same plots. These plots indicate local convexity when the objective function only depends on the pairs of parameters shown used in the plots.

## 4 Discussion

Our results indicate that the true geometry can be recovered using gradient descent when a single parameter is perturbed. However, as the number of perturbed parameters increases, the problem becomes more complex, making it challenging to recover all true parameters. Analyzing the example when the detector radius and source radius were perturbed in figure 7, the parameters do not seem to converge initially. However, as the parameters gradually stabilize, they start to converge simultaneously. Using the method of gradient descent, we do not observe convergence for all parameters when every parameter is perturbed, in a reasonable time frame. This suggests that the recovery of multiple parameters may still be possible in certain cases, but could require the use of other optimization techniques beyond gradient descent. Interestingly, minimization of the objective function does not seem to necessarily correspond to convergence in parameters. In the case where multiple parameters are perturbed simultaneously, we observe convergence to true parameter values for some of the parameters while others seem to converge at a considerably slower

(a) Plot of error in per mille of parameters when all parameters are perturbed in the model for 30 iterations

(b) Plots of percentage error for parameters when all parameters are perturbed with one per mille for 1000 iterations.

Figure 9: Plots of convergence for two parameters.
pace or not at all. This indicates, since a monotonic decrease in the value of the objective function is observed, that some parameters have a much greater influence on the forward projection than others for the minimization of the objective function. Furthermore, we observe that, for this combination of geometry and phantom, pitch seems to be an important parameter for reconstruction quality, since its convergence reduces the value of the objective function by a considerable amount. The relationship between the choice of geometry parameters and the reconstructions dependence on those parameters could be investigated further for a more complete understanding of the behavior of the objective func-


Figure 10: Heat map of the objective function with perturbation of $-1 \%$ to $1 \%$.


Figure 11: Logarithmic heat map of the objective function with perturbation of $-1 \%$ to $1 \%$.


Figure 12: Values of the objective function resulting from perturbing each parameter $1 \%$.
tion. Furthermore, the heat maps in figure 10 and 11 provide very little insight about the convexity of the objective function when all parameters are included in the model and so the convexity of the objective function should be analyzed further. Another topic that could be investigated is the uniqueness of the forward projection. As Zhang et al. [2022] shows, for a cone beam and a helical path geometry, there will be overlapping data between adjacent helical scans when a pitch that is less than half the detector width is used. This is the case in our experiments. The selection of an optimization algorithm can significantly impact the convergence properties of the model, and alternative optimization algorithms may be better suited to determine the true parameter values. Incorporating a momentum term into the gradient method represents one potential improvement, although this may also elevate the risk of overshooting. Another viable approach involves including second derivative terms to enhance the accuracy of each step during the optimization process. Furthermore, stochastic gradient descent algorithms may be better suited for parameter recovery due to their resilience against local minima, particularly considering the difficulty in determining the convexity of the objective function. It may also be possible to use a deep learning-based approach for tuning the parameters. This entirely different approach would, however, likely provide difficulties of its own.

## 5 Acknowledgments

We would like to extend our sincere gratitude to Jevgenija Rudzusika for her invaluable contributions to our thesis. Her assistance in providing us with relevant reading materials, helping us set up the programming environment, and guiding us in understanding the libraries and mathematics of medical imaging was critical to the success of our project. We are also grateful to Ozan Öktem
for his excellent supervision and support throughout the entire process. Thank you both for your outstanding support.

## References

Doğa Gündüzalp, Batuhan Cengiz, Mehmet Ozan Ünal, and İsa Yıldırım. 3D U-NetR: Low dose computed tomography reconstruction via deep learning and 3 dimensional convolutions, 2021.

ISO. Iso 12345:2015 occupational health and safety management systems - guidelines for the implementation of iso 45001, 1989. URL https://www.saiglobal.com/PDFTemp/Previews/OSH/ISO/ISO_12345_ 10-01/T016943E.PDF.

Hiroyuki Kudo, T. Rodet, Frederic Noo, and Michel Defrise. Exact and approximate algorithms for helical cone-beam ct. Physics in medicine and biology, 49:2913-31, 08 2004. doi: 10.1088/0031-9155/49/13/011.

Alok Kumar and Gursharan Singh. Fan-beam geometry based inversion algorithm in computed tomography (ct). Computerized Medical Imaging and Graphics, 37(7-8):557-563, 2013. ISSN 0895-6111. doi: https://doi.org/10. 1016/j.compmedimag.2013.06.002. URL https://www.sciencedirect.com/ science/article/pii/S0263822313006090.
M. Kumar, S. Kumar, and A. Kumar. Sinogram concept approach in image reconstruction algorithm of a computed tomography system using MATLAB. In 2010 International Conference on Computer and Communication Technology, pages $1-5$, 2010. doi: 10.1109/ICCCT.2010.5640394.

Lawrence Lechuga and Georg A. Weidlich. Cone beam CT vs. fan beam CT: A comparison of image quality and dose delivered between two differing CT imaging modalities. Cureus, 8(9):e778, 2016. doi: 10.7759/cureus.778. URL http://doi.org/10.7759/cureus. 778.

Andreas Maier, Stefan Steidl, Vincent Christlein, and Joachim Hornegger. Medical Imaging Systems: An Introductory Guide. Springer, 2018. doi: 10.1007/978-3-319-96520-8. URL https://www.ncbi.nlm.nih.gov/books/ NBK546147/.

Frank Natterer and Frank Wübbeling. Mathematical methods in image reconstruction. Monographs on Mathematical Modeling and Computation. Society for Industrial Mathematics, 2001.

William C. Scarfe and Allan G. Farman. What is cone-beam CT and how does it work? Dental Clinics, 52(4):707-730, 2008.
Y. Zhang, Y. Zhang, J. Wang, X. Li, and Y. Li. A geometric calibration approach for an industrial cone-beam CT system based on a low-rank phantom.

Measurement Science and Technology, 33(12):125001, 2022. doi: 10.1088/ 1361-6501/ac38ef. URL https://doi.org/10.1088/1361-6501/ac38ef.

