Sinogram Interpolation Method for Sparse-Angle Tomography

Martti Kalke, Samuli Siltanen
Department of Mathematics and Statistics, University of Helsinki, Helsinki, Finland
Email: martti.kalke@helsinki.fi

Received October 13, 2013; revised November 13, 2013; accepted November 20, 2013

ABSTRACT

In sparse-angle X-ray tomography reconstruction, where only a small number of projection images are taken around the object, appropriate sinogram interpolation has a significant impact on image quality. A novel sinogram interpolation method is introduced for extreme sparse tomographic reconstruction where only nine measured projection images are available. The sinogram is interpolated by solving characteristics of the so-called warps, which can be considered as approximation sine waves in a limited region. The numerical evidence suggests that this approach gives superior results over standard interpolation methods when the tomographic data are extremely sparse and noisy.

KEYWORDS
Interpolation; Sparse Imaging; Tomography; Sinogram

1. Introduction

Three-dimensional (3D) X-ray imaging is typically needed for measuring the inner structures of a patient. For example, in implant dentistry where damaged or missing teeth are replaced by artificial teeth, a screw hole with accurate depth, angle and diameter is needed for the implant. The hole has to be deep enough to ensure firm attachment, but not too deep as nerves might then be damaged. This measurement cannot be based on two-dimensional (2D) images since they represent a distorted projection of the tissue. Another clinical application of 3D X-ray imaging is solving the superposition problem. Since a pixel of a 2D projection image is a sum of attenuation along the path of an X-ray, the overlapping low-contrast tissues are difficult to recognize. This can easily lead to misinterpretation and eventually cause a false diagnosis. However, in 3D imaging, the viewing angle can be chosen so that the boundaries between tissues can be accurately identified.

Lately, there has been an increasing interest in sparse-angle tomographic imaging, where fewer projection images are taken in order to reduce patient dose and scanning time. Despite that the sparse reconstruction does not offer high spatial resolution, it can offer a feasible overview on the imaging target, which extends the use of computed tomography (CT) imaging to less serious cases, like minor trauma studies or cosmetic operation planning. Moreover, sparse-angle imaging can also be used in pre-scans, where a low-resolution reconstruction is generated before the actual scan to verify patient positioning and to define optimal X-ray technique values (tube current, voltage, number of projection images) for the final scan. The pre-scan helps to optimize the patient dose and avoid re-exposures.

The challenge of sparse tomographic imaging is insufficient image quality caused by the inadequate number of projection images. This lack of data is illustrated in Figure 1 in the spatial, sinogram and frequency domains. In the final reconstruction, the missing information can be observed as artifacts in the 2D slice view as well as in the 3D volume view. In the slice view, the missing information generates streak artifacts (see Figures 2-4(b)).
$$t_{\text{max}} = \frac{a(N-1)}{2}$$

The distance $a$ is the pixel size, $\omega$ is the difference of the frequency components, $N$ is the number of pixels, $\Delta \theta_0$ is the angular difference between the projection images and $d$ is the maximum distance between two frequency components in adjacent columns. The relation between the spatial and the frequency domain is based on well-known Fourier Slice Theorem, which states that one projection image defines frequencies in the 2D frequency domain within a single line. This line intersects origin, it has same angle than the imaging angle and it is limited by the Nyquist frequency theorem. Books by Kak (2001) and Gonzalez (2008) provide detailed theoretical background of the Fourier Slice Theorem.

**Figure 1.** Representations of two projection images and their relations in (a) spatial, (b) sinogram and (c) frequency domain. The distance $a$ is the pixel size, $\omega$ is the difference of the frequency components, $N$ is the number of pixels, $\Delta \theta_0$ is the angular difference between the projection images and $d$ is the maximum distance between two frequency components in adjacent columns. The relation between the spatial and the frequency domain is based on well-known Fourier Slice Theorem, which states that one projection image defines frequencies in the 2D frequency domain within a single line. This line intersects origin, it has same angle than the imaging angle and it is limited by the Nyquist frequency theorem. Books by Kak (2001) and Gonzalez (2008) provide detailed theoretical background of the Fourier Slice Theorem.

**Figure 2.** The FBP reconstruction of the Shepp-Logan-phantom with 5% Gaussian noise. Upper row: (a) The original phantom, (b) reconstruction from original data and (c) SINT interpolation. Lower row: FBP reconstruction when reference interpolation methods were used; (d) linear; (e) nearest neighbor and (f) spline interpolation.

especially when sharp edges are present [1]. Numerical implementations of the tomographic reconstruction from sparse data are introduced for example in articles by Hyvönen et al. (2010), Rantala et al. (2006), Varjonen (2006), Siltanen et al. (2003) or Kolehmainen et al. (2003) or Kolehmainen et al. (2008) [2-7]. Moreover, the studies by Webber et al. (1999) or Cederlund et al. (2009) are illuminating examples of the capability of clinical sparse angle X-ray tomography [8,9]. In tomographic imaging, the reconstruction pipeline consists of three stages. The first stage is *pre-processing*, where the unidealities and artifacts are compensated for. In the second *reconstruction* stage, the 3D model is calculated based on these manipulated projection images, the known spatial relation of the gantry...
Figure 3. The FBP reconstruction of the Dental Arc-phantom with 5% Gaussian noise. Upper row: (a) The original phantom, (b) Reconstruction from original data and (c) SINT interpolation. Lower row: FBP reconstruction when reference interpolation methods were used; (d) Linear; (e) Nearest neighbor and (f) Spline interpolation.

Figure 4. The FBP reconstruction of the Boxes-phantom with 5% Gaussian noise. Upper row: (a) The original phantom, (b) Reconstruction from original data and (c) SINT interpolation. Lower row: FBP reconstruction when reference interpolation methods were used; (d) Linear, (e) Nearest neighbor and (f) Spline interpolation.
M. KALKE, S. SILTANEN

(i.e. mechanical combination of the detector and the X-ray source) and the object. Finally, in the post-processing
stage, the reconstruction artifacts are reduced and the clinically relevant information is emphasized. A practical
and clinical overview of the reconstruction process can be found in the book by Hsieh (2009) and more

A sinogram is a 2D representation of an X-ray CT scan, where each column represents a single row in the
projection image arranged in increasing angular order. Typically, the horizontal axis represents the angle of the
X-ray detector and the vertical axis, the distance from the rotation center in the detector row. Each volume
element generates a sine wave in the sinogram when the X-rays are orthogonal to the detector (parallel beam
geometry). Therefore, the sinogram consists of a number of overlapping sine functions, whose amplitude and
phase depend on the location of the voxel and the grayvalue equals to the grayvalue of the volume element while
the wavelength is constant (2π). As a simple example of a two-point sinogram seen in Figure 5. Figure 1
illustrates the relation between spatial and sinogram domains, and the upper row in Figure 6 is an example of a
sinogram arising from sparsely sampled data. See chapter 5.11.3 in the book by Gonzalez and Woods (2008) for
more information about sinogram representation [12].

In this study, we focus on interpolating new sinogram columns without modifying the measured sinogram
columns. This scenario is consistently called in this paper as sinogram interpolation. In the literature, there are
also a number of other interpolation routines related to sinogram manipulation which should not be confused
with the sinogram interpolation scenario. One of them is the projection image interpolation, where new values
are generated between the data values to avoid Moire artifacts. This is typically handled by the standard nearest-
neighbor or linear interpolation [1]. Another example is needed for an interpolation routine in helical CT device
for gaining uniform sampling in the azimuthal direction or metal artifact removal processes [13-18].

Also, the sinogram extrapolation can be used for solving the truncation problem as indicated for example by
Sze and Shum (1996), Gilland et al. (2000) or Baojun and Jiang (2007) to expand the projection data in limited
angle tomography [19-21]. Moreover, sinogram extrapolation can be used for expanding the truncated sinogram
in a local tomography situation, where some of the attenuations take place outside the modeled volume (see
Zamyatin and Nakanishi (2006) and references therein [22]).

Sinogram interpolation itself has been studied since the innovation of the medical CT system in the 1970’s.
For example, Brooks et al. (1978) studied both interpolating new data in projection images and interpolating
new projection images to attenuate Moire effects [1]. Also, Lahart (1981) studied a similar problem and used a
least-squares approximation to interpolate projection images when the external shape of the object was known
[23]. However, in those days, the computational resources were very limited and therefore the spatial resolution
as well as the image quality was not equivalent to today’s de-facto standards. Nevertheless, these studies
illuminate the limitations of sparse and limited angle tomographic imaging and therefore can be considered as
fundamental studies of sinogram estimation (see also [24-27]).

![Figure 5. Example of sinogram presentation. Left is the original object with two non-zero points. Middle is the
sinogram presentation of the same object when nine images are taken from 20 to 180 degree angles. Each point
generates a sine shaped wave so that the amplitude is proportional to the distance between the point and the center of
the object (i.e. rotation axis). Right is the reconstruction done from the projection data by using back projection
algorithm.](image-url)
Lately, there have appeared several studies on dedicated directional sinogram interpolation algorithms. For example, Zhe et al. (2003) successfully combined the Huesman algorithm with Principle Component Analysis in PET imaging, and Köstler et al. (2006) applied PDE-based methods for interpolation and the Neumann boundary condition for defining missing columns in the sinogram [28,29]. In this study, about 60% of the sinogram columns were interpolated. Similarly, in the study by Penbel et al. (2005), only 50% of the sinogram columns were interpolated by using cumulant functions instead of linear interpolation [30]. Moreover, Bertram et al. (2004 and 2009) utilized tensors to gain directional interpolation based on sinogram data and achieved impressive results [31,32].

Also, interesting approaches for sinogram interpolation in ill-posed situations are the usage of Stackgrams by Happonen (2005) and the usage of object geometry estimation for sinogram interpolation by Prince et al. (1993) [33,34]. The advanced general directional interpolation methods are also considered by Gerchberg (1974), Papoulis (1975) as well as La Riviere and Pan (1999) [35-37]. Finally, Dong et al. (2013) introduced very promising results when combining the sinogram interpolation with the wavelet cost function [38]. For more general information about different sinogram related extrapolations and interpolations, see for example Brooks et al. (1978), Lahart (1981) or Kak and Slaney (2001) and references within [1,11,23].

In this paper, we introduce a new approach, called Sinogram Interpolation Technique (SINT) for estimating new sinogram columns between known, measured sinogram columns in the extremely sparse situation of only nine known sinogram columns.

This article is organized as follows: In Section 1.2, the theoretical background of the SINT method is described. Secondly, in the Section 1.3, the capability and limitations of SINT by three different numerical examples with additional noise are demonstrated. Finally, in the Section 1.4, we discuss the computational and clinical aspects of SINT and its possible future applications. Also, we analyze the failure of standard interpolation methods in extremely sparse imaging situations.

Throughout this paper, capital letters are used for matrices. Moreover, for indicating a single element in a matrix, the standard [row][column]-order is applied. For example, $S_{k,h}$ indicates the element located at the $k$th row and $h$th column in the matrix $S$. Notwithstanding the aforesaid, a single subscript is used for referring to columns: $S_h$ denotes the $h$th column of the matrix $S$. Superscripts in brackets are used for
additional information, like iteration round, special restrictions or relations. See Table 1 for the definition of each notation used in this work.

We systematically use the term known sinogram column for the columns that are based on actual measured X-ray projection images. The interpolated sinogram columns are called estimated columns since they are a priori unknown. Furthermore, the known sinogram columns adjacent to an estimated column are called known neighbor columns. See also Figure 7 for an illustration of this terminology.

We use the term volume for the target object and image for the projection data, although in our examples the phantoms are two-dimensional and projection images are one-dimensional. Similarly, we use the terms voxel and pixel to describe the volume and image elements to be consistent with other tomographic literature and references. Finally, in this study, we are focusing on 2D parallel beam imaging situations, where all X-rays are perpendicular to the detector. Also, we are covering only cases where the whole object is covered in each projection image, and therefore the so-called local tomography problem is excluded from this study.

2. The SINT Method

In this Section the theoretical background of the SINT method is discussed. Firstly, in Section 2.1 two new concepts, called warp and weight factor, are introduced. Secondly, in the Section 2.2 a new method for defining the grayvalues in single sinogram column located between two a priori known neighbor columns is proposed. Finally, in Section 2.3 the method is summarized and extended to cover all sinogram columns in the sinogram domain.

<table>
<thead>
<tr>
<th>parameter</th>
<th>definition</th>
<th>explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>$a \in \mathbb{N}$</td>
<td>pixel size, see Figure 1</td>
</tr>
<tr>
<td>$A$</td>
<td>$A \in \mathbb{R}_{+}^{N \times L}$</td>
<td>Relation grayvalue matrix, see Equation (1.14)</td>
</tr>
<tr>
<td>$d$</td>
<td>$d \in \mathbb{N}$</td>
<td>maximum distance between adjacent columns, see Figure 1</td>
</tr>
<tr>
<td>$G$</td>
<td>$G \in \mathbb{R}_{+}^{N \times L}$</td>
<td>Eigenvector matrix, see Equation (1.18)</td>
</tr>
<tr>
<td>$h$</td>
<td>$1 \leq h \leq H$</td>
<td>index of the sinogram matrix column or angle vector</td>
</tr>
<tr>
<td>$H$</td>
<td>$H \in \mathbb{N}$</td>
<td>Number of projection images</td>
</tr>
<tr>
<td>$i, j$</td>
<td>$i, j \in \mathbb{N}$</td>
<td>general purpose indexes</td>
</tr>
<tr>
<td>$k$</td>
<td>$k \in \mathbb{N}$</td>
<td>warp index</td>
</tr>
<tr>
<td>$L$</td>
<td>$L \in \mathbb{N}$</td>
<td>Number of warps</td>
</tr>
<tr>
<td>$N$</td>
<td>$N \in \mathbb{N}$</td>
<td>Number of pixels in projection image</td>
</tr>
<tr>
<td>$\hat{N}$</td>
<td>$0 &lt; \hat{N} \leq N \in \mathbb{N}$</td>
<td>Number of non-zero pixels in projection image</td>
</tr>
<tr>
<td>$R$</td>
<td>$R \in \mathbb{N}<em>{+}^{N \times L \times 2}$, $R</em>{ij} = [0,1]$</td>
<td>Rule matrix, see Equation (1.20)</td>
</tr>
<tr>
<td>$S$</td>
<td>$S \in \mathbb{R}_{+}^{N \times L}$</td>
<td>sinogram, see Equation (1.1)</td>
</tr>
<tr>
<td>$v^{(\lambda)}$</td>
<td>$v^{(\lambda)} \in \mathbb{R}_{+}^{L}$</td>
<td>Eigenvector of $\Lambda$ with eigenvalue of one, see Section 2.2</td>
</tr>
<tr>
<td>$w$</td>
<td>$w \in \mathbb{R}_{+}^{L}$, $0 \leq w \leq 1$</td>
<td>weight vector, see Section 2.2.2</td>
</tr>
<tr>
<td>$W_{i}(\theta)$</td>
<td>$k \in \mathbb{Z}_{+}$</td>
<td>Warp function, see Equation (1.9)</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>$\alpha \in \mathbb{R}_{+}$</td>
<td>Warp factor vector, see Equation (1.18)</td>
</tr>
<tr>
<td>$\beta$</td>
<td>$\beta \in \mathbb{R}_{+}$</td>
<td>regularization parameter, see Equation (1.24)</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>$\kappa &gt; 1$</td>
<td>interpolation factor, see Equation (1.28)</td>
</tr>
<tr>
<td>$\Lambda$</td>
<td>$\Lambda \in \mathbb{R}_{+}^{N \times L \times 2}$</td>
<td>used for generating the matrix $G$, see Equation (1.15)</td>
</tr>
<tr>
<td>$\mu, \nu$</td>
<td>$\mu, \nu \in \mathbb{N}$</td>
<td>arbitrary warp indexes</td>
</tr>
<tr>
<td>$\theta$</td>
<td>$\theta \in \mathbb{R}_{+}, 0 \leq \theta &lt; \pi$</td>
<td>imaging angle vector, see Equation (1.2)</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>$\Phi \in \mathbb{R}_{+}^{L \times L}$</td>
<td>Warp matrix, see Section 2.2</td>
</tr>
<tr>
<td>$\psi$</td>
<td>$\psi \in \mathbb{R}_{+}^{N \times L}$</td>
<td>Relation index matrix, see Section 2.2</td>
</tr>
<tr>
<td>$\omega$</td>
<td>$\omega \in \mathbb{R}_{+}$</td>
<td>distance between frequency components, see Equation (1.26)</td>
</tr>
</tbody>
</table>
2. Definitions of the Key Concepts

2.1. Weight Factor

In this study a sinogram is considered as a matrix \( S \in \mathbb{R}^{N \times H} \) with one unknown column \( S_h \). That is

\[
S = \left[ S_1 S_2 S_3 \cdots S_{(k-1)} S_h S_{(k+1)} \cdots S_H \right],
\]

and known projection angles corresponding to the columns of \( S \) as a vector

\[
\theta = [\theta_1 \theta_2 \theta_3 \cdots \theta_{k-1} \theta_h \theta_{k+1} \cdots \theta_H],
\]

where \( \theta \in \mathbb{R}^H \) and \( 0 \leq \theta_1 < \theta_2 < \theta_3 < \cdots < \theta_H < \pi \).

The purpose of the sinogram integration is to generate a feasible approximation for \( S_h \) when other columns and vector \( \theta \) are known. In this method the grayvalues of each unknown sinogram element \( S_{2,h} \) are approximated as weighted sums of nearby sinogram elements in the adjacent columns \( S_{h-1} \) and \( S_{h+1} \). This can be determined in two ways, each producing the same result for \( S_h \); either based on the column \( S_{h-1} \) as

\[
S_{2,h} = \sum_{i=1}^{N} w_{i}^{(h-1)2} S_{i,(h-1)}
\]

or based on the column \( S_{h+1} \) as

\[
S_{2,h} = \sum_{i=1}^{N} w_{i}^{(h+1)2} S_{i,(h+1)}
\]

where \( w_{i}^{(h \pm 1)} \in \mathbb{R}^N \) are called as the weight factors. The first upper index of weight factors indicates the known column index and the second indicates the sinogram element index in the estimated column \( h \). The Equations (1.3) and (1.4) clearly indicate that if the weight factors are known, then an estimate for the \( S_{2,h} \) can be generated. Moreover, if the weight factors for all elements in \( S_h \) are generated, an estimate for the whole sinogram column \( S_h \) is gained.

The Equations (1.3) and (1.4) are combined by defining \( L_2 \) approximation for the \( S_{2,h} \) as

\[
S_{2,h} = \arg \min_{S_{2,h}} \left\| \hat{S}_{2,h} - \sum_{i=1}^{N} w_{i}^{(h \pm 1)2} S_{i,(h \pm 1)} \right\|_2,
\]

whose solution is given by
We end this chapter by highlighting two important identities related to weight factors. Firstly, consider a single sin wave that intersects the sinogram elements $s_{ih}^{-1}$, $s_{ih}^1$, and $s_{ih}^{+1}$. Then we have

$$w_{ih}^{(k-1)h} = w_{ih}^{(k+1)h}.$$  \hspace{1cm} (1.7)

Secondly, because the total sum of each sinogram column is constant, the sum of all weight factors related to the same known sinogram component in columns $h - 1$ and $h + 1$ is always one, i.e.

$$\sum_{j} w_{j}^{(k+1)h} = 1 \text{ for each } j.$$ \hspace{1cm} (1.8)

2.1.2. Warp

In extremely sparse-angle tomographic situation, the number of sine functions and their characteristic parameters (amplitude, grayvalue and phase) cannot be uniquely defined from the sinogram. For that reason, a new concept called warp is introduced. The warp can be considered as a sum of all the sine waves that intersects two sinogram elements $s_{ih}^{-1}$ and $s_{ih}^{+1}$ (see Figure 7). The most essential features of warps are:

1) Each warp has the shape of a sine wave and a wavelength of $2\pi$. Therefore, the phase and amplitude of the warp are uniquely defined by two points.

2) Each warp connects two sinogram elements in known neighbors columns $s_{ih}^{-1}$ and $s_{ih}^{+1}$ to a single sinogram element in the estimated column $s_{h}$ by two strictly positive weight factors.

3) Each warp intersects only strictly positive sinogram elements in the whole sinogram domain.

4) All non-zero sinogram elements are associated with at least one warp and all weight factors are associated with exactly one warp.

Our interpolation strategy is to define the estimated sinogram column $s_h$ by defining the amplitude, phase and so-called warp factor (will be explained in the Section 2.2) for each warp. We will show that by defining these values we can define the weight factors and thereby obtain an estimate for the sinogram column $s_h$. Although the above warp-based approach does not completely characterize the underlying sine waves and provide an exact solution to the interpolation problem, it does give a feasible and stable approximation in limited-data situations as demonstrated in Section 1.3.

2.2. Definition of the Crucial Matrices

In this Section we will introduce three matrices to determine the connection between the warps and grayvalues of the estimated sinogram $(s_h)$. These matrices are:

1) Matrix $A$ to define the relation between the estimated and neighbor known sinogram elements based on the Equation (1.6)

2) Matrix $G$ to fulfill the requirement in the Equation (1.7)

3) Matrix $R$ to limit the weight factors related to the same known sinogram based on the Equation (1.8)

We also build two temporary matrices $\Psi$ and $\Phi$ to simplify the numerical implementation. All these matrices should be generated for each sinogram column separately as described in Section 2.3.

2.2.1. Warp Matrix $\Phi$

The purpose of the warp matrix $\Phi \in \mathbb{R}^{L \times 3}$, where $L$ is the number of warps, is to simplify the creation of other matrices. The warp matrix $\Phi$ defines all warps related to the estimated sinogram column $s_h$ such that each row represents a single warp as follows: The first element in the row is the row index $i_1$ of the known left point $s_{ih}^{-1}$, the second element is the row index $i_2$ of the estimated sinogram column $s_{ih}^{+1}$ and finally the row index $i_3$ is the right known point $s_{ih}^{+1}$.

To define the value $i_1$ the warp function $W_i(\theta)$ is defined for all angles $\theta$ based on two points in the sinogram $(s_{ih}^{-1}$ and $s_{ih}^{+1})$ and their angular values $(\theta^{-1}$ and $\theta^{+1})$.

Since the warp has the shape of a sine wave with a frequency of one and an offset of $(N+1)/2$, it can be defined as
where the amplitude $\mathcal{A}_k$ and phase $\mathcal{P}_k$ depend on the known angles $\theta_{k_{+1}}$ as follows:

$$\mathcal{A}_k = \frac{q_2}{\sin(\theta_{k_{+1}} - \phi_1)}$$

and

$$\mathcal{P}_k = \arccot\left(\frac{q_2 \cos(\theta_{k_{+1}}) - q_1 \cos(\theta_{h+1})}{q_2 \sin(\theta_{k_{+1}}) - q_1 \sin(\theta_{h+1})}\right),$$

where

$$q_1 = \frac{N+1}{2} - i_3 \quad \text{and} \quad q_2 = \frac{N+1}{2} - i_2.$$

If $S_{W_i(\theta)}j > 0$ is for each $j = \{1, 2, 3, \ldots, (h-1), (h+1), \ldots, H\}$, then the warp is valid (see the requirement item 3 in the Section 2.1) and finally $i_2 = \lceil W_k(\theta) \rceil$ where $\lceil \cdot \rceil$ is the rounding operator. Because of this limitation, the height of the matrix $\Phi$ typically smaller than $NN$. The procedure for generating the warp matrix $\Phi$ is described in the Algorithm 1.

### 2.2.2. Relation Matrices $A$ and $\Psi$

The purpose of this phase is to define two quite similar sparse matrices; Relation Grayvalue Matrix $A \in \mathbb{R}^{N \times 2L}$, where and Relation Index Matrix $\Psi \in \mathbb{R}^{N \times 2L}$. The matrix $A$ includes grayvalues of the known sinogram columns $S_{h_{+1}}$ and $\Psi$ consists of the corresponding row indexes. First we define the Relation Grayvalue Matrix $A$, which specifies the relations between the weight factor vector $w$ and the sinogram elements in the estimated column $S_i$. The Equation (1.6) can be represented in a matrix form as

$$S_i = Aw,$$

where the matrix $A \in \mathbb{R}^{N \times 2L}$ consists of zeros and known neighbor sinogram gray values $S_{h_{+1}}$ multiplied by $\frac{1}{2}$. However, unlike in Equation (1.6), the vector $w$ in Equation (1.13) consists only the weight factors that are included in warps and therefore the size of the vector $w$ is $2L$. The vector $w \in \mathbb{R}^{2L}$ will be called in this study as weight vector.

The relation matrix $A$ can be constructed from the row vectors $\tilde{A}^{(i)} \in \mathbb{R}^{N \times K_i}$ such that each row vector $\tilde{A}^{(i)}$ consists of the grayvalues of known neighbor sinogram columns that are connected to an estimated sinogram element $S_{i_{+1}}$ by the warps as described in Section 2.2 and Figure 7. Since each weight factor is

```plaintext
Data: Matrix $S$ and vector $\theta$
Result: Matrix $\Phi$
begnin
\begin{array}{l}
  k = 1 \\
  \text{for } i_2 \leftarrow 1 \text{ to } N \text{ do} \\
  \quad \text{for } i_3 \leftarrow 1 \text{ to } N \text{ do} \\
  \quad \quad \text{Calculate } W_{i_k}(\theta) \text{ (see Equation (1.9))} \\
  \quad \quad \text{if } S_{W_i(\theta)}j > 0 \text{ for all } j \text{ then} \\
  \quad \quad \quad i_2 \leftarrow [W_k(\theta)] \\
  \quad \quad \quad \mathcal{P}_{i_2} \leftarrow [i_1 i_2 i_3] \\
  \quad \quad \quad k \leftarrow k + 1 \\
  \quad \end{array} \\
\text{end} \\
\text{end} \\
end
```

**Algorithm 1. Generation of the warp matrix $\Phi$.**
related to exactly one estimated sinogram element, only one non-zero value on each column is allowed. Therefore, the matrix $A$ has the form

$$A = \frac{1}{2} \begin{pmatrix} A^{(0)} & 0 & \cdots & \cdots & 0 \\ 0 & A^{(2)} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & A^{(N-1)} & 0 \\ 0 & \cdots & \cdots & 0 & A^{(N)} \end{pmatrix}, \tag{1.14}$$

where $0$ are zero row vectors.

We also need the row indexes of the known points (indicated above as $i_1$ and $i_3$) ordered consistently with the structure of the matrix $A$. Those values will be stored into matrix $\Psi \in \mathbb{R}^{N \times 2L}$. The procedure for generating the matrices $A$ and $\Psi$ is described in Algorithm 2.

At this point the matrices $A$ and $\Psi$ have been generated. The matrix $A$ defines the relation between the estimated sinogram column and the weight factors as described in Equation (1.13). The matrix $\Psi$ plays an important role in defining the eigenvector and rule matrices in the following Sections.

2.2.3. Eigenvector Matrix $A$

Having constructed the matrix $A$ appearing in Equation (1.13), we now turn to specifying the weight vector $w$. From the Equation (1.7) can be seen that

$$\Lambda w = w \tag{1.15}$$

where $\Lambda \in \mathbb{R}^{2L \times 2L}$ such that

$$\Lambda_{i_1, i_3} = \frac{S_{h_i(k-1)}}{S_{h_i(k+1)}} \text{ and } \Lambda_{i_3, i_1} = \frac{S_{h_i(k+1)}}{S_{h_i(k-1)}}.$$

if and only if $S_{h_i(k-1)}$ and $S_{h_i(k+1)}$ are related to the same warp, otherwise $\Lambda_{i_1, i_3} = 0$ and $\Lambda_{i_3, i_1} = 0$.

From the Equation (1.15) can be observed that the weight vector $w$ is an positive eigenvector for $\Lambda$ related to an eigenvalue of one$^1$. The number of feasible eigenvectors is exactly $L$ based on the fact that for each positive eigenvector, say $\lambda^{(p)} \in \mathbb{R}^{2L}$, there is always another eigenvector $\lambda^{(p+1)} \in \mathbb{R}^{2L}$ such that $\langle \lambda^{(p)}, \lambda^{(p+1)} \rangle = 0$. Therefore, vector $\lambda^{(p)}$ has at least one negative component and it is not therefore feasible

$^1$This eigenvalue problem always has a solution based on the following fact: Since in each row and column in the matrix $\Lambda$ there is exactly one non-zero component and for each non-zero element in $\Lambda$ holds $\Lambda_{i_1, i_3} = 1/\Lambda_{i_3, i_1}$, the matrix $\Lambda$ is positively semi-definite and orthogonal and it has (only) eigenvalues of one.
solution. For additional information about matrices, eigenvectors and eigenvalues see Golub and Van Loan (1996) or Arfken and Weber (2001) [39,40].

Next we will declare that a feasible solution for the Equation (1.15) can also be a linear combination of the eigenvectors. The Eigenvector Matrix \( G \in \mathbb{R}^{2L\times L} \) is defined as

\[
G = \begin{bmatrix}
v^{(1)} & v^{(2)} & \cdots & v^{(L)}
\end{bmatrix}
\]  

(1.16)

where \( v^{(1)}, v^{(2)}, \ldots, v^{(L)} \) are all positive eigenvectors of the matrix \( \Lambda \) related to eigenvalue of one as discussed above. Then for every vector \( \alpha \in \mathbb{R}^L \) holds

\[
AG\alpha = \left[ \Lambda v^{(1)} v^{(2)} \cdots v^{(L)} \right] \alpha = Ga
\]  

(1.17)

which shows that the linear combination of feasible eigenvectors fulfills the requirement (1.7).

The vector \( \alpha \) is called as warp vector. Weight vector \( w \) can now be determined from the warp vector \( \alpha \) by utilizing the Eigenvector Matrix \( G \in \mathbb{R}^{2L\times L} \) such that

\[
w = Ga
\]  

(1.18)

The procedure for generating the matrix \( G \) is shown in the Algorithm 3.

By combining the Equations (1.13) and (1.18) we get

\[
S_h = AG\alpha
\]  

(1.19)

which indicates that if we can define the warp factors \( \alpha \), we can also determine the grayvalues of the estimated sinogram column \( hS \). To calculate the warp factors \( \alpha \) instead of weight factors \( w \) is a twofold benefit; the requirement (1.7) is now implicitly implemented and the dimension of the original problem is reduced from \( 2L \) to \( L \).

2.2.4. Rule Matrix \( R \)

So far we have generated matrices to determine the relations between the warp factors and the grayvalues in the estimated column. However, the mission for defining the grayvalues for the column \( S_h \) has not been accomplished since the warp factor vector \( \alpha \) is still unknown. Therefore, in this Section we introduce a Rule Matrix \( R \in \mathbb{R}^N_{\hat{N}_1 \times \hat{N}_2} \) ( \( \hat{N}_1 \) and \( \hat{N}_2 \) are number of non-zero elements in column \( S_{h-1} \) and \( S_{h+1} \) ) to determine the actual values of the warp factors.

As defined in Equation (1.8), the sum of weight factors connected to any known sinogram element \( S_{h \pm 1} \) equals to one. To fulfill this requirement, we generate a Rule Matrix \( R \) which sums all weight factors that are connected to the same known sinogram element. Then

\[
Rw = 1
\]  

(1.20)

where the \( \mathbf{1} \) is a vertical vector of ones with a height of \( \hat{N}_1 \hat{N}_2 \).

The principle of generating the Rule Matrix \( R \) is following; for each known non-zero sinogram element a single row to the matrix \( R \), say \( R_j \), is generated so that the value of each element \( R_{j,h} \) = 1 if weight factor \( w_h \) is connected to the same known non-zero sinogram element, otherwise \( R_{j,h} = 0 \). The order of the rows in the matrix \( R \) is irrelevant. The pseudo-code for generating the matrix \( R \) is described in Algorithm 4.

![Algorithm 3. Generation of the matrix G.](image-url)
2.3. Summary of the Method

We summarize this Section by declaring that the grayvalues of the estimated column can be solved from the Equation

\[ S_h = AG\alpha \]  

such that

\[ RG\alpha = 1 \]  

which can be easily gained by combining Equations (1.18) and (1.20).

Since the \( \alpha \) cannot be uniquely defined from the Equation 1.22, the problem is ill-posed and it calls for regularization. The Tikhonov regularization algorithm was chosen, since it controls simultaneously residual \( RG\alpha - 1 \) and the norm of the solution \( \|\alpha\| \) (see chapter 2.3 in Kaipio and Somersalo (2005) for further information [41]). The Tikhonov solution for the Equation 1.22 has a form of

\[ \alpha = \left( (RG)^T RG + \beta I \right)^{-1} (RG)^T 1 \]  

where \( \beta \in \mathbb{R}_+ \) is the regularization factor and \( I \in \mathbb{R}_0^{K_2 \times K_2} \) is an identity matrix. The regularization factor can be based on the fact that in the non-local tomography the sum of each sinogram column is constant for throughout the sinogram. Therefore, the relaxation value \( \beta \) can be calculated from the Equation

\[ \beta = \arg \min_S \sum S_{\beta} - \sum S_{\beta - 1} \]  

The complete Algorithm 5 for defining \( S_h \) is described in details.

To calculate The Equation 1.23 less memory consuming, the Singular Value Decomposition (SVD) can be implemented for the matrix \( RG \) such that \( RG = U\Sigma V^T \). Then the Equation 1.23 has a matrix-free representation

\[ \alpha = \sum \frac{\sigma_n}{\sigma_n + \beta} (1,u_n) v_n \]  

where \( n \) is the minimum dimension of the matrix \( \Sigma \). See for example Theorem 2.5 in Kaipio and Somersalo (2005) for proof [41].

So far we have described how to interpolate a single sinogram column. To interpolate all sinogram columns the procedure described in Sections from 2.2 to 2.2 are repeated for each unknown column. To numerically compute that, two nested loops are applied; an inner loop where the neighbor columns are fixed and the location of the estimated sinogram column varies and an outer loop where we change the neighbor columns. Before starting the actual interpolation process, we have to define how many new sinogram columns are needed for the isotropic resolution. If we generate too few new sinogram columns, we do not gain isotropic resolution. On the
Algorithm 5. Generation of the Sinogram column $S_h$. Constant $\varepsilon$ is a small true positive number. See also Equation (1.25) for alternative way to calculate $S^{(p)}_h$.

other hand, too many new sinogram columns just increases the computational burden without any improvements to the final reconstruction quality.

From Figure 1(c) we can see that

$$\sin \frac{\Delta \theta}{2} = \frac{d/2}{\omega(N-1)/2}, \quad (1.26)$$

where $\Delta \theta$ is the angular difference of the projection images, $\omega$ is the distance between two frequency components, and $N$ is the number of frequency samples. For simplicity, $N$ is assumed to be odd and $\Delta \theta$ is constant.

Based on the Fourier Slice Theorem, to gain uniform resolution in the frequency domain, the distance between frequency components within measurement $\omega$ should be equal to the maximum distance between adjacent frequency components $d$ as discussed by Kak and Slaney (2001) and more deeply by Natterer (1986) [11,42]. Therefore, by defining $\omega = d$, we can define Equation for optimal angular difference $\Delta \theta^{(opt)}$ based on the Equation (1.26) as

$$\sin \frac{\Delta \theta^{(opt)}}{2} = \frac{1}{N-1} \quad (1.27)$$

We introduce an interpolation factor $0 < \kappa > 1$ such that $\Delta \theta_0 = \kappa \Delta \theta^{(opt)}$ where $\Delta \theta_0$ is the original angular difference in sinogram. Then from the Equation (1.27) we get

$$\kappa = \left[ \frac{\Delta \theta_0}{2 \arcsin \frac{1}{N-1}} \right] \quad (1.28)$$

where $[\cdot]$ is the upward rounding operator.

The angle $\theta^{(z)}_h$ for interpolated column $S_h$ can therefore be calculated from the Equation

$$\theta^{(z)}_h = \theta_{z-1} + \frac{z}{\kappa} \Delta \theta \quad (1.29)$$

where $z = 1, 2, 3, \ldots, \kappa - 1$. The detailed procedure for whole sinogram interpolation is described in the Algorithm 6.

3. Results

We used the GNU Octave program (version 3.2.3) with the imaging toolbox for numerical implementation of
the SINT algorithm. The coding and executing environment was Ubuntu distribution version 10.04 including GNU/Linux operating system 2.6.32 and GNOME interface version 2.30.2. The computer that we employed for this study was a commercial 2.4 GHz dual-processor desktop computer with 8 GB of memory.

For the numerical implementation we adopted three synthetic phantoms with a size of $128 \times 128$ pixels; Shepp-Logan, Dental Arc and Boxes. The standard Shepp-Logan phantom was chosen since it is widely used for evaluating the reconstruction image quality. The Dental Arc was calculated from full CT reconstruction of an artificial head (i.e. real teeth and skull embedded in an acrylic head) and therefore it can be considered as a reference for the clinical capability of SINT. Thirdly, the Boxes phantom consists of two isolated homogeneous rectangles to demonstrate a simplest possible case.

Furthermore, these three phantoms were chosen to evaluate the capability of SINT because these phantoms generate essentially different kinds of sinogram representations, as shown in Figure 6. Since the Shepp-Logan phantom is a relatively isotropic object, it produces a smoothly varying sinogram. Secondly, the Dental Arc is a strongly anisotropic object, which produces a narrowing and widening sinogram with significant dynamic range in grayvalues. Finally, the Boxes phantom consists of two separate objects generating a branching and merging sinogram.

In all cases, we took nine synthetic projection images from 25 to 185 degree angles by using the built-in radon function. This function calculates the sum of pixels along a line with the given angles. The interpolation factor was set to 32 based on Equation (1.28), which gave about 0.6 degrees angular differentiation for the final interpolated sinogram. We also evaluated the robustness of this method against pixel noise by adding Gaussian noise with variation of 5% of the pixel grayvalue to each known sinogram element.

As reference interpolation methods, we used linear, cubic spline and nearest neighbor -interpolation methods, which are the most common interpolation methods. For further information about these interpolation methods, see Section 4.4 in the book by the Gonzalez and Woods (2008) [30].

As an interpolation quality metric, we calculated relative $L^2$ norms against the ground truth. That is

$$\frac{\|S^{(\text{int})} - S^{(\text{true})}\|_2}{\|S^{(\text{true})}\|_2} \times 100\%$$

(1.30)

where $S^{(\text{int})}$ is the interpolated sinogram and $S^{(\text{true})}$ is the noiseless sinogram generated from the original phantoms with similar angular difference than the interpolated sinograms. The results are shown in the Table 2.

We also reconstructed the object from the interpolated data for subjective comparison of reconstruction image quality. The reconstruction was executed by the built-in iradon-function, which executes Filtered Back-Projection (FBP) reconstruction with a Ram-Lak -filter. The FBP was chosen since since it exposes the artifacts caused by interpolation flaws better than iterative methods like Algebraic Reconstruction Technique (ART), which have been considered as more suitable algorithms for the ill-posed situations (see for example Kolehmainen et al. (2003) or Siltanen et al. (2003) [6,5]. For post-processing only linear grayvalue mapping was used to guarantee fair comparison. The authors prefer Kak and Slaney (2001) and Natterer (1986) for further information about the reconstruction algorithms and Gonzalez and Woods (2008) for post-processing [11,12,42]. The reconstruction results can be seen in Figure 2.
Table 2. The relative $L^2$-norm of error in noiseless case and 5% noise added to projection images. The error is calculated as in the Equation (1.30).

<table>
<thead>
<tr>
<th></th>
<th>linear</th>
<th>spline</th>
<th>nearest</th>
<th>SINT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shepp-Logan</td>
<td>9.16</td>
<td>9.78</td>
<td>11.97</td>
<td>6.80</td>
</tr>
<tr>
<td>Dental Arc</td>
<td>18.51</td>
<td>19.94</td>
<td>24.90</td>
<td>14.00</td>
</tr>
<tr>
<td>Boxes</td>
<td>19.60</td>
<td>20.89</td>
<td>28.27</td>
<td>7.30</td>
</tr>
<tr>
<td>Shepp-Logan with noise</td>
<td>10.01</td>
<td>10.89</td>
<td>13.00</td>
<td>7.09</td>
</tr>
<tr>
<td>Dental Arc with noise</td>
<td>19.02</td>
<td>20.68</td>
<td>25.56</td>
<td>14.15</td>
</tr>
<tr>
<td>Boxes with noise</td>
<td>20.41</td>
<td>21.94</td>
<td>29.03</td>
<td>8.06</td>
</tr>
</tbody>
</table>

Figure 8. Profiles of the middle row in Shepp-Logan (top), Dental Arc (mid.) and Boxes (bottom) phantoms. The solid line is the profile of the original phantom, dashed is reconstruction with the SINT interpolation and dotted is the reconstruction with linear interpolation. Nearest neighbor and spline were not plotted because they produced very similar result than the linear interpolation.

4. Discussion

The results in this study are very preliminary. However, both qualitative and quantitative results suggest that SINT is significantly better than general interpolation routines when the angular data sampling is extremely sparse. This can be observed from the metrics as well as comparing the subjective image quality in Figures 2-4. See also the profiles in Figure 8.

In this study, we did not compare the SINT method against any sinogram-dedicated or state-of-the-art interpolation method for a number of reasons. Firstly, none of them are dedicated to as sparse a situation as our proposed method. For example, in the studies by Zhe et al. (2003) and Köstler et al. (2006) about 60% of the sinogram columns were interpolated, while in our study over 92% of the sinogram columns were interpolated [28,29]. Similarly, in the study by Penßel et al. (2005) only 50% of the sinogram columns were interpolated [30]. Secondly, part of the methods published recently are not fully documented or they require the right parameter settings for optimal result. Therefore, they cannot be used as reference since we could not guarantee a fair comparison. Thirdly, some of the new interpolation methods are designed for specific cases like the studies by Constantino (2006 and 2009) and earlier by Tam et al. (1990) [43-45]. These studies were limited to a situation where only the outer boundaries of isolated objects were modeled, while we concentrated on a more generic sinogram.
interpolation. Finally, we considered that a comparison with generally known interpolation algorithms gives a better understanding of the capability of this method than comparing to specific and dedicated interpolations.

There are two reasons why generic interpolation routines fail when they are applied to sparse sinograms. First, the generic interpolation methods do not take advantage of the characteristics of the sinogram. Second, the generic interpolation methods work only when there is a strong correlation between the neighbor sinogram values. However, when the distance from the known column increases the correlation between sinogram elements decreases. This vanishing correlation generates a whirlpool-shaped pattern which can be seen in the lower row of Figures 2-4. To verify these findings, we calculated the absolute error as a function of distance from the known sinogram column, \( i.e. \)

\[
\varepsilon_i = \left\| S_i^{(\text{int})} - S_i^{(\text{true})} \right\|_2 
\]

(1.31)

where \( i \) is the distance from the known column to the interpolated column (with index \( i \) satisfying \( 1 \geq i \geq (\kappa - 1)/2 \), where \( \kappa \) is the interpolation factor as defined in Section 2.3). The influence of distance to the interpolation error can be seen in Figure 9, which shows that when the distance from the known sinogram column increases, the interpolation error of the SINT method increases significantly slower than the interpolation error of reference interpolation methods. The disadvantage of the SINT method is execution time, which in our examples was about 15 s per interpolation. The most time consuming task was to verify that all warps intersect non-zero elements in the known columns as described in Section 2.2. Since every combination of the left and right neighbor column elements has to be considered as a warp candidate, the warp tracking operation has to be repeated \( \kappa (H - 1) \) \( \times \) times during the whole interpolation process, where \( N \) is height of the sinogram, \( H \) is the number of original sinogram columns and \( \kappa \) is the interpolation factor. In this study, this meant over 4 million warp tracking operations. Another time consuming operation was the calculation of the eigenvalues for the matrix \( G \). In this study, we applied the Hessenberg and Schur decomposition based QR

![Figure 9](image-url)

Figure 9. The SINT method (solid line), linear (long dash), spline (short dash) and nearest neighbor (dotted line) interpolation errors as a function of the distance from the known column for all three cases; Shepp-Logan (left), dental-arc (mid) and boxes (right). The error is calculated as described in the Equation (1.31).
algorithm, which is a well-known iterative method for defining the eigenvalues without the sorting procedure as discussed by Golub and Van Loan (1996) [39]. Since the matrix $\Lambda$ was very sparse, positive-definite and orthogonal with only eigenvalues of one, the sorting procedure was irrelevant and only few iterations were needed.

There are number of articles indicating that parallel computing radically decreases the execution time in the image processing [46-48]. Despite the fact that we used a non-optimized high-level interpreted language in this study, the SINT method could easily be coded for a parallel processing environment, such as clustered computers or general purpose graphical processing unit (GP-GPU), by using OpenCL, Cuda or other parallel computing coding techniques. The parallelization of the SINT method is effective since the interpolated sinogram values are determined only from the original known sinogram columns, which are constant during the interpolation process and therefore they can be stored in fast read-only memory before executing the interpolation routine.

In this study, the SINT method has been proved to work in parallel beam imaging geometry cases in 2D, where the X-ray beam is perpendicular to the detector, instead of applying it to fan-beam in 2D or cone-beam in 3D. This limitation has been implemented partly for simplifying the theory and keeping the focus on the interpolation method itself. However, our interest is to expand this study to fan-beam and cone-beam imaging geometries in the near future. Our basic strategy for the parallel to fan beam conversion is based on modifying the warp matrix $\Phi$ (see Section 2.2). As illustrated by Kak and Slaney (Chapters 3.4 and 3.5) [11], the projection angle in fan beam imaging geometry is the sum of the projection angle and the fan angle, i.e.:

$$\tilde{\theta}_h(i) = \theta_h + \gamma_i$$  

(1.32)

where $\gamma_i$ is the fan angle, which depends on the sinogram element index. The fan angle can be calculated from the Equation

$$\gamma_i = \arctan \frac{q}{c}$$  

(1.33)

where $c$ is the shortest distance from X-ray source point to detector and $q$ is defined by Equation (1.12). This means that the angular value of each column $h$ is no longer constant, but depends on the row index of the sinogram element. Despite that this modification makes generation of the warp matrix $\Phi$ more complex, the rest of the algorithm can be utilized per se. Finally, as shown by Kak and Slaney (Chapter 3.6), cone-beam geometry can be considered as a stack of fan beam sinograms and the 3D volume can be built by the interpolation process in the azimuthal direction.

5. Conclusion

Finally, very promising implication of the SINT method is in metal artifact reduction. Since metals are typically very radio-dense, they block a major part of the X-rays causing streak artifacts as indicated by Goldman and Fowlkes (2000) and Veldkamp et al. (2010) [49,50]. Since metallic areas can easily be seen in the sinogram as white regions, they can be identified, and the interpolation process could be used across these regions in the sinogram domain as indicated earlier by Ziying and Sze (1998), Yazdi and Beaulieu (2006), Meyer et al. (2009) or later by Abdoli et al. (2011) and Karimi et al. (2012) [15-18,51].

Acknowledgements

This research project was supported by the Instrumentarium Foundation and the Academy of Finland (project 141094 and the Finnish Centre of Excellence in Inverse Problems Research, decision number 250215). The authors wish to thank PaloDEx Group (Tuusula, Finland) and the Finnish Inverse Problems Society for their support. Finally, a special thanks to those who made this study possible by working on open source projects.

REFERENCES


http://dx.doi.org/10.3934/ipt.2010.4.257


http://dx.doi.org/10.1109/TMI.2005.862206


http://dx.doi.org/10.1088/0031-9155/48/10/314


http://dx.doi.org/10.1088/0031-9155/48/10/315


http://dx.doi.org/10.1137/070682791


http://dx.doi.org/10.1038/sj.dmf.4600201


http://dx.doi.org/10.1259/dmf/16577933


http://dx.doi.org/10.1137/1.9780898719277


http://dx.doi.org/10.1109/23.775592


http://dx.doi.org/10.1109/18.887832


http://dx.doi.org/10.1109/23.873022


http://dx.doi.org/10.1364/JOSA.71.001155


http://dx.doi.org/10.1364/AO.18.003704


