Neural network Hilbert transform based filtered backprojection for fast inline x-ray inspection

To cite this article: Eline Janssens *et al* 2018 *Meas. Sci. Technol.* **29** 034012

View the article online for updates and enhancements.

Related content

- Low-dose CT reconstruction via edge-preserving total variation regularization
Zhen Tian, Xun Jia, Kehong Yuan et al.

- A two-step Hilbert transform method for 2D image reconstruction
Frédéric Noo, Rolf Clackdoyle and Jed D Pack

- FBP algorithms for fan-beam data
Jiangsheng You, Gengsheng L Zeng and Zhengrong Liang

Neural network Hilbert transform based filtered backprojection for fast inline x-ray inspection

Eline Janssens[1], **Jan De Beenhouwer**[1], **Mattias Van Dael**[2], **Thomas De Schryver**[3], **Luc Van Hoorebeke**[3], **Pieter Verboven**[2], **Bart Nicolai**[2] **and Jan Sijbers**[1]

[1] imec-Visionlab, Department of Physics, University of Antwerp, Antwerp, Belgium
[2] UGCT, Department of Physics and Astronomy, University of Ghent, Ghent, Belgium
[3] Mebios, Biosystems Department, University of Leuven, Leuven, Belgium

E-mail: Eline.Janssens@uantwerpen.be

Abstract

X-ray imaging is an important tool for quality control since it allows to inspect the interior of products in a non-destructive way. Conventional x-ray imaging, however, is slow and expensive. Inline x-ray inspection, on the other hand, can pave the way towards fast and individual quality control, provided that a sufficiently high throughput can be achieved at a minimal cost. To meet these criteria, an inline inspection acquisition geometry is proposed where the object moves and rotates on a conveyor belt while it passes a fixed source and detector. Moreover, for this acquisition geometry, a new neural-network-based reconstruction algorithm is introduced: the neural network Hilbert transform based filtered backprojection. The proposed algorithm is evaluated both on simulated and real inline x-ray data and has shown to generate high quality reconstructions of 400×400 reconstruction pixels within 200 ms, thereby meeting the high throughput criteria.

Keywords: inline inspection, x-ray tomography, filtered backprojection

(Some figures may appear in colour only in the online journal)

1. Introduction

In industry, there is a great demand for fast x-ray inspection and quality control. To make x-ray inspection time efficient, an inline x-ray system is often preferred. Ideally, it should allow individual inspection of every single sample, while preserving a sufficiently high throughput. Inline inspection techniques are already used in different industries, such as agriculture [1, 2], powder metallurgy [3], log scanning [4], dynamic processes [5], metrology [6], and baggage inspection [7].

The fastest and easiest way of inspecting objects inline with x-rays is radiography. To acquire a radiograph, a side-view arrangement is employed with a source and detector on

opposite sides of the conveyor belt. Based on the radiograph, interior features of a sample, like dense materials or foreign objects, can be detected [8, 9] or components can be inspected [3]. A high efficiency and a relatively inexpensive infrastructure are the main advantages of x-ray radiography. However, plain radiography comes with a substantial disadvantage: due to the accumulation of the attenuation coefficients along the direction of the projection, depth information is lost and possible defects cannot be spatially resolved in 3D. Moreover, defects may render invisible, hidden behind or in front of materials with a higher attenuation coefficient.

The need for 3D information can be met by using a more advanced x-ray inspection technique: computed tomography (CT). Conventional CT exploits information from a large number of projections to obtain an image of the interior of the sample and is widely used in the field of offline inspection and dimensional metrology [10–13]. CT systems consist of

either a source and detector rotating around the object or an object rotating between a source and detector. However, full rotation of the object between a fixed source and detector is not possible in an inline setup and full rotation of the source and detector around the object is difficult or even impossible to realise in an inline setup when reconstruction speed and geometrical constraints are an issue. Furthermore, these conventional CT systems come with a high infrastructure cost (>500 000 euro [14]).

A more cost-friendly x-ray setup that still allows fast and spatially resolved imaging is a side-view arrangement consisting of a fixed cone beam source and a detector moving along with the object, while the object traverses through to the x-ray beam (figure 2). From the (limited) angular range from which projections are acquired, image reconstruction is possible. However, these images will typically suffer from smearing artefacts due to the missing wedge. Several attempts have been made to reduce these artefacts with iterative reconstruction. Sidky *et al*, for example, derived a volume image reconstruction technique for a finite straight-line source trajectory [15]. Zhang *et al* [16] performed a feasibility study for x-ray tomography in a straight-line trajectory scan based on a total variation iterative procedure. The same group introduced an image reconstruction technique based on total variation minimization and alternating directions to reconstruct images in a linear scan [17]. Despite the improved image quality that can be obtained with these techniques, their usefulness is limited for fast inspection due to the long computation time of iterative reconstruction methods.

Recently, we introduced an alternative solution to the angular range problem by adding a rotation of the sample around an axis perpendicular to the conveyor belt [18–20], which largely solves missing wedge artefacts. Nevertheless, even in this scanning geometry the number of projections must be kept small to keep the reconstruction time limited, which may lead to undersampling artefacts. Therefore, in this work, we propose a new type of fast fan beam reconstruction algorithm, analogous to the parallel beam neural network approach of [21, 22]. Our algorithm is based on the Hilbert transform FBP (hFBP) [23] for which the filter is trained by a neural network (NN-hFBP). An advantage of the method is that the NN-hFBP reconstructions can be computed directly from fan-beam data, without the need for rebinning. The algorithm is validated using both simulated and experimental inline scans of agricultural products. It will be shown that the NN-hFBP allows for fast and high quality reconstructions of images in an inline environment from a limited number of projections.

2. Methods

In this section, the proposed NN-hFBP algorithm is introduced. The algorithm is based on two existing algorithms: the NN-FBP and the hFBP. The NN-FBP introduced by Pelt *et al* [21, 22] creates an image by combining multiple FBP reconstructions, each obtained with a different filter. These filters are trained beforehand in a neural network based on an existing training dataset. High quality images can be reconstructed in a
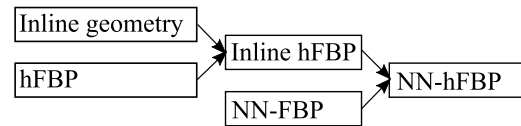

Figure 1. Schematic of the structure describing how the NN-hFBP is formed in this paper.
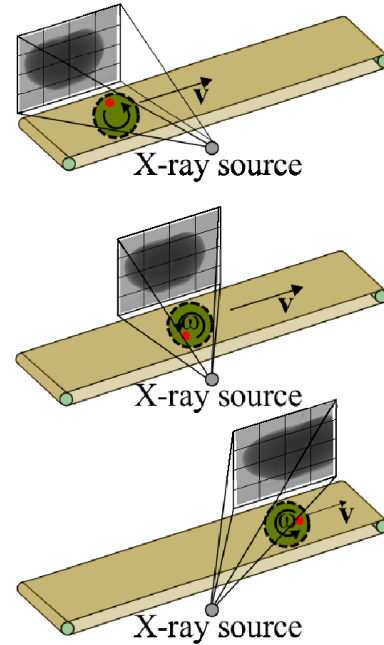

Figure 2. Inline inspection with a flat panel detector that moves along with the object while it transverses and rotates.

very short time with the NN-FBP. However, the method is only applicable to parallel beam data, which restricts its application for x-ray imaging mainly to synchrotron beamlines. For most x-ray sources, the x-rays are emitted in a cone beam. When only considering the central slice of a cone beam dataset, a fan beam dataset can be obtained. Although rebinning from fan to parallel beam would allow direct application of NN-FBP, it slows down the reconstruction and often introduces interpolation artifacts. Therefore, we chose to adapt the NN-FBP algorithm for direct application to fan beam data. To do this, the hFBP [23] was used instead of the conventional FBP. In the hFBP algorithm, the differential of the Hilbert transform of the projection data is backprojected onto the reconstruction plane to create the reconstructed image. In this paper, the hFBP is first adapted for an inline acquisition geometry [18–20] in section 2.1. Next, position- and angle-independent filters are derived in section 2.2, to form the NN-hFBP reconstruction. A schematic of the structure is shown in figure 1.

2.1. Inline Hilbert transform based FBP

The inline acquisition geometry that we will work with consists of an object that rotates and translates on a conveyor belt while passing a fixed source and detector system. The detector can either be steady at a fixed position opposite to the source or it can move along with the object over a certain distance. The disadvantage of a fixed detector is its limited field of view,
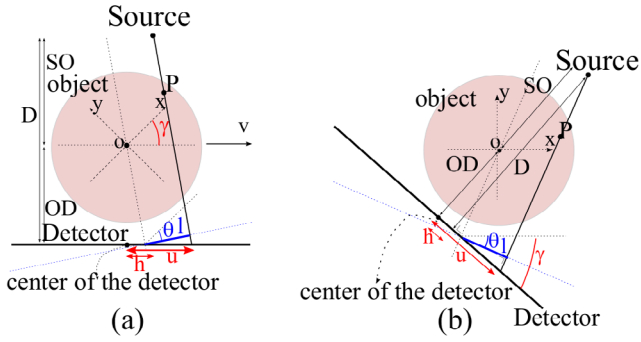
Figure 3. Acquisition geometry for inline inspection. (a) Point of view of the fixed source and a rotating and translating object. (b) Point of view of a fixed object and a rotating source and detector.

forcing objects to rotate faster to obtain projections from a large angular range. Without loss of generality, we chose the acquisition geometry with a moving detector as shown in figure 2 for the remainder of the paper.

To adapt the hFBP algorithm so that it can be used for an inline inspection geometry, we start from the Hilbert transform based reconstruction algorithm for parallel beam data. Here, the Hilbert transform is applied on the detector coordinate. The parallel beam reconstruction formula is given by [23]

$$f(r,\phi) = \frac{1}{8\pi} \int_0^{2\pi} \frac{\partial p_H(l,\theta)}{\partial l} d\theta \tag{1}$$

where p_H is the Hilbert transformed projection data, f is the reconstructed image, (r,ϕ) are polar coordinates, and (l,θ) represent the parameters of a parallel beam geometry: the detector pixel and the projection angle, respectively. Our inline acquisition geometry is characterized by the detector pixel u, the translation distance h between the source and the center of the object and the rotation angle γ of the object (see figure 3). Figure 3(a) shows a projection in the geometry from the point of view of a fixed source and rotating and translating object, while figure 3(b) shows the same projection from the point of view of a fixed object where the source and detector are rotating around the object. In figure 3, D is the distance between the source and the plane of the detector, OD is the distance from the detector to the origin, SO the distance between the origin and a plane through the source parallel to the detector and P a pixel that we want to reconstruct. It is important to notice that the translation distance h is positive when the object is in front of the central position and negative behind the central position.

If the rotation speed ω of the object, expressed in rad/m, is constant, the object's rotation angle γ can be written in terms of this rotation speed and the translation distance h so that only two independent parameters remain:

$$\gamma = -h \cdot \omega. \tag{2}$$

Note that the rotation angle γ is zero when the object is at the central position.

Our new reconstruction formula for inline data will be derived from (1). Therefore, we must express (1) in terms of

parameters h and u instead of l and θ. To do this, l and θ are first written in terms of u and h:

$$l = \frac{uSO + hOD}{\sqrt{D^2 + (u-h)^2}} \tag{3}$$

$$\theta = -h\omega + \arctan\frac{u-h}{D}. \tag{4}$$

To simplify the notation of the upcoming equations, we now introduce a variable $t = u - h$. The Hilbert transform for inline data can now be defined similarly to the Hilbert transform of fan-beam data with a flat panel detector described in [24]. Only in the inline setup, the object is not positioned in the center of the beam. Therefore, we replace the detector pixel u with t, which results in the following Hilbert transform for inline data:

$$p_H^{inl}(u_i, h_j) = \frac{\sqrt{t_{i,j}^2 + D^2}}{\pi} \int \frac{p^{inl}(u_i - \tau, h_j)d\tau}{\tau\sqrt{(t_{i,j} - \tau)^2 + D^2}} \tag{5}$$

where p^{inl} is the inline projection data, p_H^{inl} is the Hilbert transformed inline projection data, and $t_{i,j} = u_i - h_j$, then (6) holds:

$$p_H^{inl}(u,h) = p_H\left(\frac{uSO + hOD}{\sqrt{D^2 + t^2}}, -h\omega + \arctan\frac{t}{D}\right). \tag{6}$$

A proof, similar to appendix A in [24] for (6), can easily be derived.

To create the reconstruction formula for inline data, (6) is used to adapt equation (1). To express the derivative of the parallel Hilbert transform in terms of the derivatives of the inline Hilbert transform, first the partial derivatives of p_H^{inl} with respect to u and h are calculated based on (6)

$$\frac{\partial p_H^{inl}}{\partial u}(u,h) = \frac{D^2 SO - Dht}{\sqrt{(D^2 + t^2)^3}} \frac{\partial p_H(l,\theta)}{\partial l}$$
$$+ \frac{D}{D^2 + t^2} \frac{\partial p_H(l,\theta)}{\partial \theta} \tag{7}$$

$$\frac{\partial p_H^{inl}}{\partial h}(u,h) = \frac{D^2 OD + Dut}{\sqrt{(D^2 + t^2)^3}} \frac{\partial p_H(l,\theta)}{\partial l}$$
$$- \left(\omega + \frac{D}{D^2 + t^2}\right) \frac{\partial p_H(l,\theta)}{\partial \theta}. \tag{8}$$

This system of equations can now be solved for $\partial p_H(l,\theta)/\partial l$:

$$\frac{\partial p_H(l,\theta)}{\partial l} = \frac{\sqrt{D^2 + t^2}}{D + \omega(D\,SO - ht)}\left[\frac{\partial p_H^{inl}}{\partial h}(u,h)\right.$$
$$\left. + \frac{\omega(D^2 + t^2) + D}{D} \frac{\partial p_H^{inl}}{\partial u}(u,h)\right] \tag{9}$$

$$= p_{F_0}^{inl}(u,h). \tag{10}$$

The reconstruction formula for inline data can now be derived from (1) by inserting (9) in (1). This results in the hFBP reconstruction algorithm for fan-beam data in an inline environment where the object rotates with a constant speed ω:

$$f(r,\phi) = \frac{1}{8\pi} \int_0^{2\pi} \frac{\partial p_H}{\partial l} d\theta$$

$$= \frac{1}{8\pi} \int_0^{2\pi} \frac{\sqrt{D^2+t^2}}{D+\omega(DSO-ht)} \left[\frac{\partial p_H^{inl}}{\partial h}(u,h) \right.$$

$$\left. + \frac{\omega(D^2+t^2)+D}{D} \frac{\partial p_H^{inl}}{\partial u}(u,h) \right] d\theta. \quad (11)$$

Here, f is the reconstructed image and u corresponds to the detector pixel where the ray through (r,ϕ) hits the detector at displacement h and $t = u - h$. To compute the integral, the projection data is interpolated in u and h to obtain data corresponding to the desired parallel beam θ values. Therefore, (3) and (4) should be converted to expressions for u and h. To do this, the $\arctan(\frac{u-h}{D})$ was approximated by $\frac{u-h}{D}$ since $\frac{u-h}{D}$ was small. After discretization of (11), the discrete reconstruction algorithm for inline inspection with a constant rotation speed consists of 4 steps and is described in appendix A.

In the case of equiangular data acquisition, the rotation speed of the object is dependent on its position on the conveyor belt. The rotation angle γ can then be written as

$$\gamma = \Gamma \frac{\arctan\left(\frac{-h}{SO}\right) - \arctan\left(\frac{-h_{st}}{SO}\right)}{\arctan\left(\frac{-h_e}{SO}\right) - \arctan\left(\frac{-h_{st}}{SO}\right)} - \gamma_{min} \quad (12)$$

$$= \omega'\left(\arctan\left(\frac{-h}{SO}\right) - a\right) - \gamma_{min}. \quad (13)$$

Here, Γ is the total angular range over which the object will rotate, γ_{min} is the angle over which the object rotates from the first projection until the central position and ω' and a are constants defined as $\omega' = \frac{\Gamma}{\arctan\left(\frac{-h_e}{SO}\right) - \arctan\left(\frac{-h_{st}}{SO}\right)}$ and $a = \arctan\left(\frac{-h_{st}}{SO}\right)$ with h_e the last h-value and h_{st} the first h-value. The positions h_i on the conveyor belt at which projections are taken are distributed in such a way that, without rotation of the object, the projections would be acquired equiangularly:

$$h_i = SO \cdot \tan\left(\alpha_{min} + (i-1) \cdot \Delta\alpha\right). \quad (14)$$

In (14), $\alpha_{min} = \arctan(\frac{-h_{st}}{SO})$ and $\Delta\alpha$ is the angle between two successive projections. The partial derivative with respect to l of the Hilbert transformed parallel projection data, similar to (9) or step 2 of the reconstruction algorithm then becomes

$$p_{F_0}^{inl}(u_i, h_j) = \left[(SO^2 + h_j^2) \frac{\partial p_H^{inl}(u_i, h_j)}{\partial h} \right.$$

$$\left. + \frac{(D(SO^2 + h_j^2) + \omega SO(D^2 + t_{i,j}^2))}{D} \frac{\partial p_H^{inl}(u_i, h_j)}{\partial u} \right]$$

$$\cdot \frac{\sqrt{D^2 + t_{i,j}^2}}{D(SO^2 + h_j^2) + \omega SO(DSO - h_j t_{i,j})}. \quad (15)$$

2.2. NN-hFBP

The inline hFBP reconstruction of the previous section can now be combined with the NN-FBP of Pelt *et al* [21, 22] to
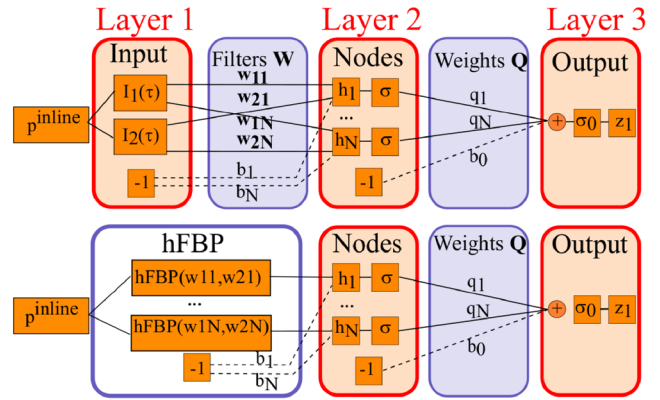

Figure 4. Scheme of the NN-hFBP reconstruction. For training, inputs I_1 and I_2 are created from the projection data with which the filters $\mathbf{w_{1i}}$ and $\mathbf{w_{2i}}$ are trained in the top row. For reconstruction, the filters $\mathbf{w_{1i}}$ and $\mathbf{w_{2i}}$ are combined and used as the new filters to perform several hFBP reconstructions in the second row.

provide fast, high-quality reconstructions for fan-beam data in an inline environment. This is, however, only possible when the hFBP is written as the product of a certain input with a position and angle independent filter. In appendix B, it is shown that the inline hFBP can be written as the sum of two terms with each term the convolution of a datavector (I_1 and I_2) of size n (the number of detector pixels) and a filter (f_1 and f_2):

$$f(r,\phi) = \sum_\tau f_1(\tau)I_1(u-\tau) + f_2(\tau)I_2(u-\tau) \quad (16)$$

where u represents the detector pixel where the ray through (r,ϕ) hits the detector. Since the neural network performs a convolution of an input with a filter, this means that the hFBP can be implemented in a neural network to create the NN-hFBP. To do this, first the correct datavectors I_1 and I_2 should be generated based on the acquired projection data to train the filters (see appendix B). For every detector pixel, two datavectors are generated which are stored in one input vector of size $2n$ so that the total length of the input of the neural network is twice the size of the detector. Once these datavectors are obtained, instead of using the normal filters of the hFBP (described in (B.3) and (B.4)), the network is trained so that the weight matrices $W_1 \in \mathbb{R}^{n \times N}$ and $W_2 \in \mathbb{R}^{n \times N}$ between the $2n$ input nodes and the N hidden nodes of the multilayer perceptron define new filters $\mathbf{w_{1i}} \in \mathbb{R}^n$ and $\mathbf{w_{2i}} \in \mathbb{R}^n$ for the hFBP reconstructions, which are the columns of the weight matrices W_1 and W_2 (replacing the filters f_1 and f_2). This means that after training the network, several hFBP reconstructions can be computed with these new filters instead of the filters of (B.3) and (B.4). Finally, the reconstructions are combined using the activation functions σ and σ_0 (in our case sigmoid functions), the trained weights $\mathbf{q} \in \mathbb{R}^N$, and the biases $\mathbf{b} \in \mathbb{R}^N$ and b_0 of the neural network, as shown in figure 4. The final reconstruction formula then becomes

$$f(r,\phi) = \sigma_0 \left(\sum_{k=0}^{N-1} q_k \sigma(\sum_\tau \right.$$

$$\left. (\mathbf{w_{1k}}(\tau)I_1(u-\tau) + \mathbf{w_{2k}}(\tau)I_2(u-\tau)) - b_k) - b_0 \right) \quad (17)$$

Table 1. Specifications scanning geometry.

	Simulations	Real data
Detector pixels	1148	1146
Detector pixel size	127 μm	127 μm
Translation distance (mm)	[−250, 250]	[−250, 250]
Object detector distance	84.500 mm	84.527 mm
Source object distance	900 mm	563 mm
Downsampling	4	2
Image pixels	256 × 256	400 × 400
Image pixel size	400 μm	200 μm

Table 2. Datasets.

	Training set	Validation set	Test set
Braeburn 1	8	4	4
Braeburn 2	20	7	10
Jonagored	10	6	7
Jonagold	15	6	10
Bell pepper	15	5	10

where w_{1k} and w_{2k} are the filters trained by the neural network.

3. Experiments and results

In this section, the performance of the NN-hFBP algorithm is evaluated with both simulated and real inline data. First, using simulation experiments, the acquisition settings and NN-hFBP network parameters are optimized for maximal image quality. In particular, the influence of equiangular versus non-equiangular sampling, the rotation speed of the sample and the number of hidden nodes of the network was studied in terms of image quality. Secondly, the reconstruction quality of the (optimized) NN-hFBP was compared to that of the conventional reconstruction algorithms FBP and SIRT with 500 iterations. Finally, the performance and image quality of NN-hFBP versus conventional reconstruction algorithms were evaluated using real data experiments. The evaluation of the image quality was done by 4 different evaluation methods: the root mean squared error (RMSE) on the whole image (global RMSE) and only on the apple, bell pepper or walnut region (local RMSE), the feature similarity index (FSIM) [25] and the most apparent distortion (MAD) [26]. The RMSE is defined as

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1..M}(rec(i) - GT(i))^2}{M}} \qquad (18)$$

where *rec* is the reconstructed image, *GT* is the ground truth image and *M* is the number of pixels in the image. For all experiments, reconstructions were made using the ASTRA Toolbox [27–29] where all forward and backprojections were calculated on an NVIDIA GeForce GTX 580 GPU.

3.1. Simulation experiment

To evaluate the performance of the NN-hFBP algorithm on simulation data, inline experiments were simulated that mimic the behavior of a real inline scan. X-ray CT scans of apples and bell peppers were used as test samples, for which the detection of small structural changes such as holes or browning are of interest. Inline CT data were created starting from conventional circular CT scans of apples and bell peppers from respectively 470 and 632 equiangular projections of 1024 × 1024 pixels. From these scans, inline scans were simulated by reorganizing corresponding rays. Such simulated

inline projection data naturally accounts for a realistic polychromatic source as well as realistic noise behaviour.

Specifications of the geometry are given in table 1. The translation distance is expressed as the distance in mm compared to the central position on the conveyor belt opposite to the source. For the experiments, four types of apples and one type of bell pepper were used. The number and types of apples and bell peppers used for training, validation and testing in the different experiments is shown in table 2.

For each experiment, 10 instances of every network were trained by randomly selecting different sets of pixels of the same image data. Each ANN was trained based on 100 000 random pixels for training and validated with 10 000 random pixels. For each of the four apple types, 100 slices of the training datasets were used for training and 10 slices of the validation datasets for validation. In case of the bell peppers, only the central slice of each bell pepper was used for training and validation since the scan quality was not good enough to make fan-beam reconstructions of non-central slices. For the bell peppers, the ANN was trained based on 15 training images and 5 validation images. The reconstruction quality was tested on 50 images for the apple datasets and on 10 images for the bell peppers.

Before describing the first experiment, we note that the hFBP and the NN-hFBP both require an interpolation step. This step causes higher reconstruction times and introduces blurring in the reconstruction in few-view acquisitions. Therefore, in this paper, we propose a heuristic approach where we omit the interpolation step and directly backproject the data along the inline fan-beam projection geometry. This means that for every pixel in the reconstruction grid, data from slightly different parallel projection angles θ is summed up. Although this is an approximation, avoiding the interpolation step makes the reconstruction much faster with only a slight loss of reconstruction quality. The effects of interpolation on the hFBP and the NN-hFBP reconstruction quality can be seen in figures 5(b) and 6(b), respectively on two inline scanned apples with ground truth images in figures 5(a) and 6(a). To avoid the blurring and to further reduce the computation time, we propose to omit the interpolation step and directly backproject the data along the inline fan-beam projection geometry. In figure 5, NN-hFBP is compared to hFBP with interpolation and the heuristic hFBP reconstruction for 32 and 128 projection angles. The hFBP with interpolation provides good image quality when a sufficiently high (in this case 128) number of projections are available, but blurring artifacts appear when only a small number of projections are present. With the heuristic approach, the holes are less blurred than with the interpolation approach for 32 projections. For 128
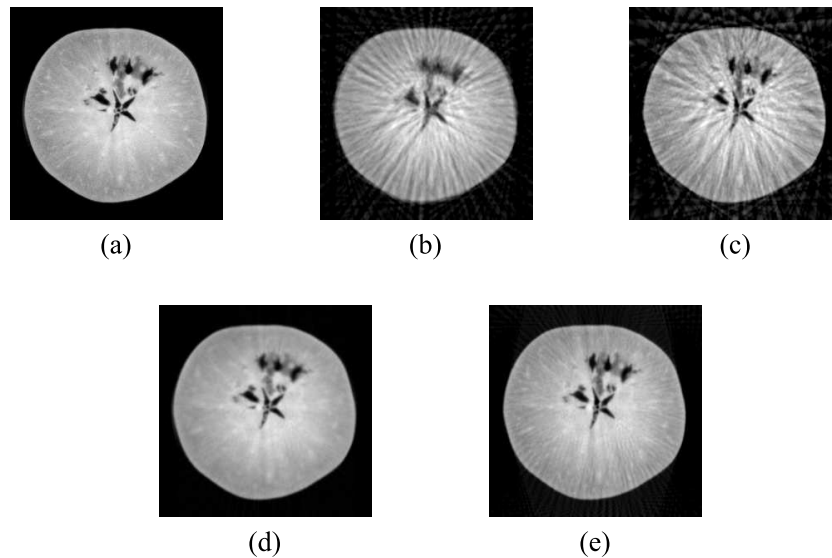
Figure 5. Comparison of the interpolation and the heuristic method. (a) Ground truth image, (b) hFBP reconstruction with interpolation from 32 projections over 2π, (c) heuristic hFBP reconstruction from 32 projections, (d) hFBP reconstruction with interpolation from 128 projections, (e) heuristic hFBP reconstruction from 128 projections.
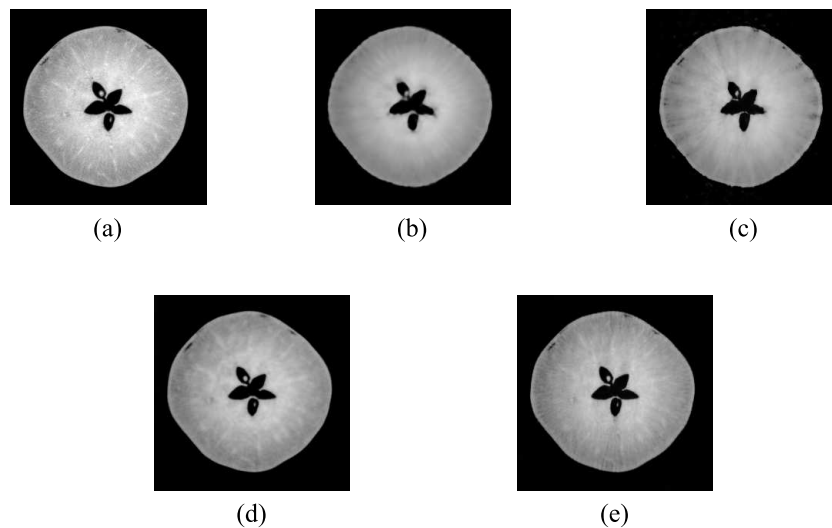

Figure 6. Comparison of the interpolation and the heuristic method in a neural network. (a) Ground truth image, (b) NN-hFBP reconstruction with interpolation from 32 projections over 2π, (c) heuristic NN-hFBP reconstruction from 32 projections, (d) NN-hFBP reconstruction with interpolation from 128 projections, (e) heuristic NN-hFBP reconstruction from 128 projections.

projections, streaking artefacts appear in the heuristic reconstructions, but the reconstruction is less blurred and the holes and brown spots are still clearly visible. In figure 6, a similar comparison is made between an inline NN-hFBP reconstruction with interpolation and a heuristic inline NN-hFBP reconstruction for 32 and 128 projection angles. With the heuristic approach, streak artefacts appear again at the outside of the apple, but the small holes are detected with 32 projections, which is not the case for the conventional NN-hFBP method. Figures 5 and 6 clearly indicate that a choice should be made between blurring or streaks in the reconstructions made with only few projections. Based on the capacity of the heuristic NN-hFBP to better detect the holes with less blurriness for a small number of projections and the faster reconstruction time, we decided to use the heuristic approach in the rest of this paper.

In the first experiment, we optimize the acquisition and network parameters to evaluate the reconstruction quality of the NN-hFBP. Important parameters for data acquisition are the rotation speed and rotation direction of the objects. Therefore, in this experiment, we first evaluate the reconstruction quality (in terms of the global RMSE) of the NN-hFBP of the Braeburn 1 apples as a function of the rotation speed when 128 projections are acquired equiangularly (see: figure 7). The rotation speed is expressed in terms of the angular range Γ over which the apple has rotated from the first projection to the last projection and ranges between $-\pi$ and π. The corresponding reconstructed images are shown in figure 8. From the graph and the images, it is immediately clear that rotation substantially improves the reconstruction quality. Without rotating the object, substantial limited wedge artefacts appear. Furthermore, there is an obvious difference
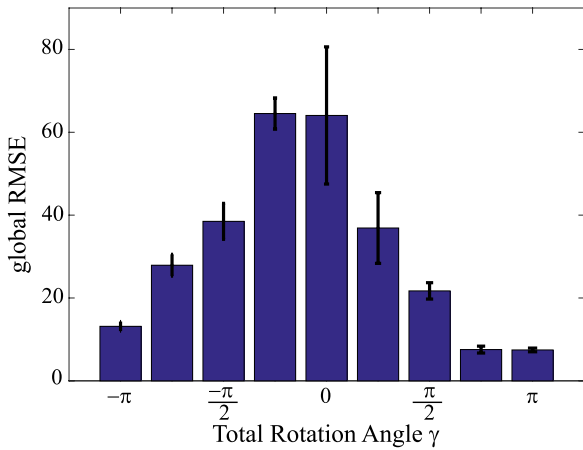
Figure 7. The global RMSE of the NN-hFBP reconstruction for different angular ranges. The *x*-axis gives the total angular range Γ over which the apple has rotated.
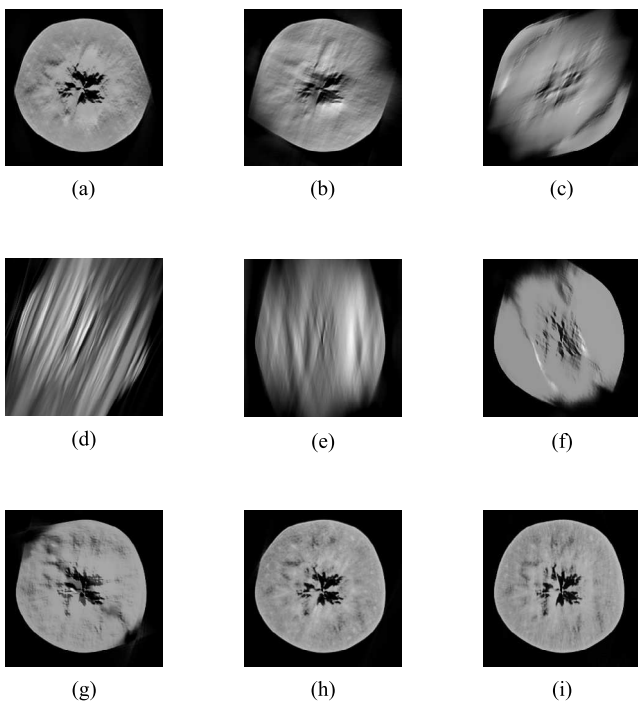

Figure 9. Global RMSE for non-equiangular and equiangular NN-hFBP reconstructions in function of the number of projections.

Figure 10. Evaluation of the reconstruction quality in terms of the global RMSE (a) and reconstruction time (b) of the NN-hFBP evaluated on Jonagored apples based on 1, 2, 4 and 8 hidden nodes for different number of projections.

Figure 8. Reconstruction images of the apples made with the NN-hFBP for angular ranges Γ equal to (a) $-\pi$, (b) $-3\pi/4$, (c) $-\pi/2$, (d) $-\pi/4$, (e) 0, (f) $\pi/4$, (g) $\pi/2$, (h) $3\pi/4$, (i) π.

between a counterclockwise and clockwise rotation. When the object rotates counterclockwise at a slow rotation speed, the different intrinsic projection angles at which the projections are required without rotation are counteracted by the rotation so that the reconstruction is similar to no rotation. A clockwise rotation however allows to substantially increase the angular range, resulting in a higher reconstruction quality.

Secondly, we inspect the influence of two different types of sampling of the projections on the final reconstruction quality. Unless prior knowledge is available about the object to be inspected, equiangular sampling is expected to be optimal. On the other hand, a constant rotation speed and equidistantly acquired projections may have a practical advantage. Hence, we investigate the difference between the reconstruction
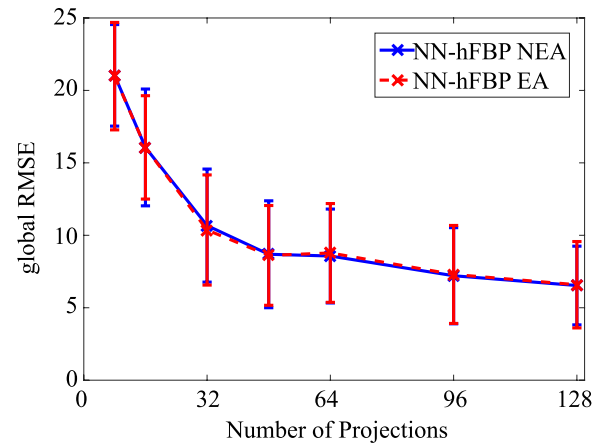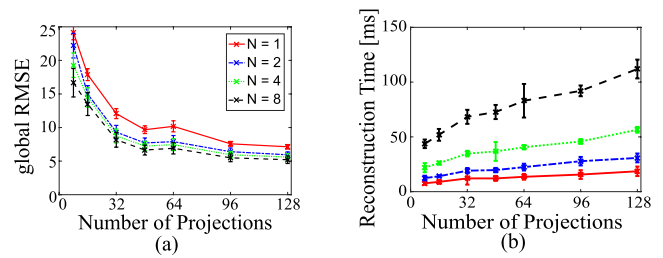
quality of the NN-hFBP for Braeburn 2 apples that rotate with a constant rotation speed (non-equiangular projections) versus apples that rotate with a varying rotation speed so that the acquired projections are equiangular. The results in terms of the global RMSE as a function of the number of projections, both with equiangular (EA) and non-equiangular (NEA) projections, are compared in figure 9. It can be seen that the reconstruction quality is very similar. Hence, in this acquisition setup, there is not a large gain by acquiring the projections equiangularly. This is a desired characteristic since it facilitates the data acquisition. Projections can then be taken at equidistant positions and the rotational speed of the apple can be kept constant.

Finally, we optimize the number of hidden nodes and thus the number of hFBP reconstructions that are combined in the neural network as it represents a trade-off between reconstruction quality and speed. We therefore examine the influence of the number of hidden nodes N of the ANN on the reconstruction quality of the NN-hFBP evaluated on Jonagored apples with a constant rotation speed and $N = 1, 2, 4$ and 8. Figure 10(a) shows the RMSE over the whole image and figure 10(b) the reconstruction time in function of the number of projections. The four graphs represent the cases of 1, 2, 4 and 8 hidden nodes. From the graphs, one can see that increasing the number of hidden nodes improves the reconstruction quality. However, it also increases the reconstruction time. For 32 and 128 projections, the reconstructed images are shown in figure 11. To balance the reconstruction quality and
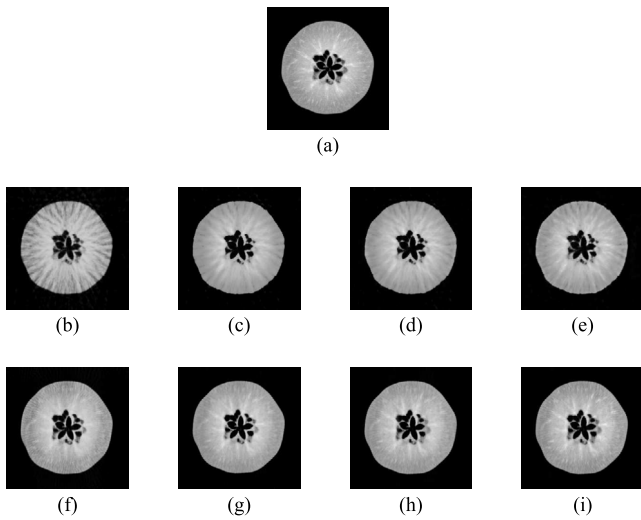
Figure 11. (a) Ground truth image of a Jonagored slice, (b-e) NN-hFBP reconstructions of 32 projections with 1 (b), 2 (c), 4 (d) and 8 (e) hidden nodes, (f-i) NN-hFBP reconstructions of 128 projections with 1 (f), 2 (g), 4 (h) and 8 (i) hidden nodes.
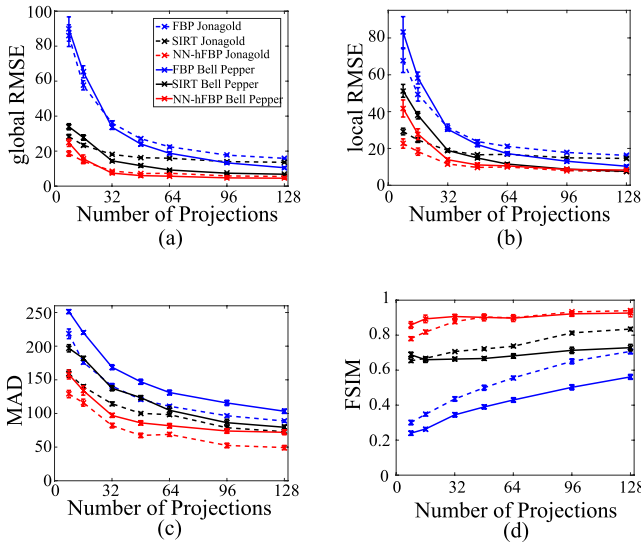

Figure 12. Evaluation of the reconstruction quality for FBP, SIRT and NN-hFBP in terms of the global RMSE (a), local RMSE (b), MAD (c) and FSIM (d), both for Jonagold apples and bell peppers.

reconstruction time, we have chosen to use four hidden nodes, aiming to optimize the reconstruction quality while preserving a reconstruction time of less than 100ms for 128 projections.

From the three experiments, optimal conditions can be derived for the NN-hFBP reconstructions. Now, the reconstruction quality of the NN-hFBP is compared to the quality of the conventional algorithms SIRT and FBP. Results of a comparison with hFBP were omitted, as NN-hFBP clearly outperforms hFBP in terms of reconstruction quality. Figure 12 shows the reconstruction quality of FBP, SIRT and the NN-hFBP as a function of the number of projections in terms of the global RMSE (a), the local RMSE (b), MAD (c) and FSIM (d). As is clear from the plots, the reconstruction quality of NN-hFBP is significantly better than that of FBP and SIRT for all number of projections, both for the Jonagold apples and the bell peppers. In particular, the quality of the NN-hFBP
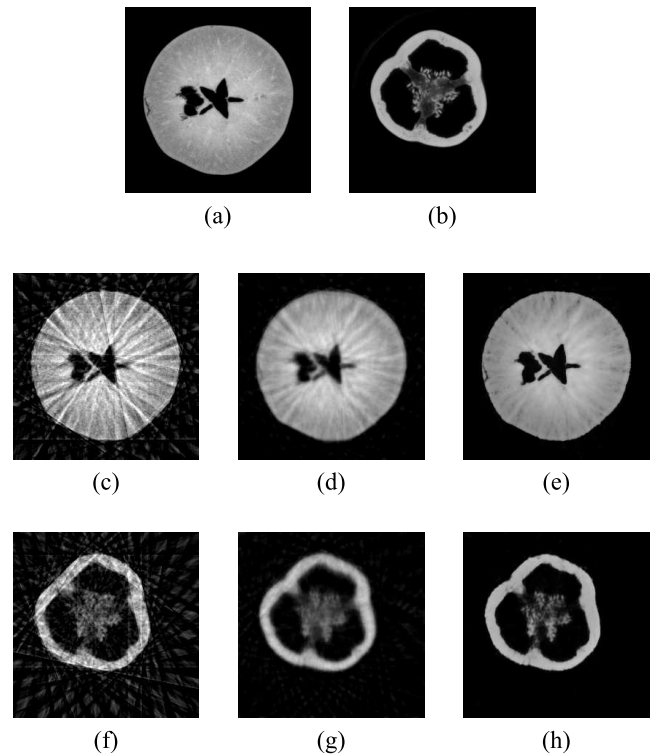

Figure 13. Ground truth images of an apple slice (a) and a bell pepper slice (b) and their reconstructions for 32 projections made with FBP ((c) and (f)), SIRT ((d) and (g)) and NN-hFBP ((e) and (h)).
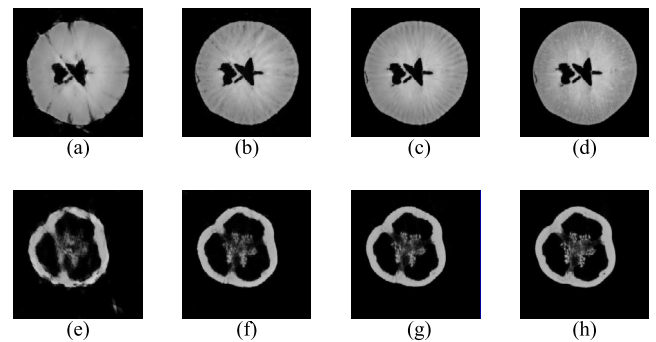

Figure 14. NN-hFBP reconstructions of the apple and bell pepper slices of figure 13 for 16 ((a) and (e)), 32 ((b) and (f)), 64 ((c) and (g)) and 128 ((d) and (h)) projections.

reconstructions in terms of the FSIM is much better than that of FBP and SIRT. This might be because the NN-hFBP is capable of clearly detecting the shape of the object and important features like holes and the core of the apple even with a very small number of projections, in contrast to FBP and SIRT.

A comparison of the reconstructed Jonagold and bell pepper slices from 32 projections with the different methods is shown in figure 13. One can clearly see that the NN-hFBP outperforms the other reconstruction algorithms since much more noise is present in the FBP reconstruction (signal-to-noise ratio of 13.44 versus 4.91 for bell peppers and 9.72 versus 4.80 for apples) and the SIRT reconstruction is slightly blurred. Example images for comparing the reconstruction quality of the NN-hFBP with different numbers of projections
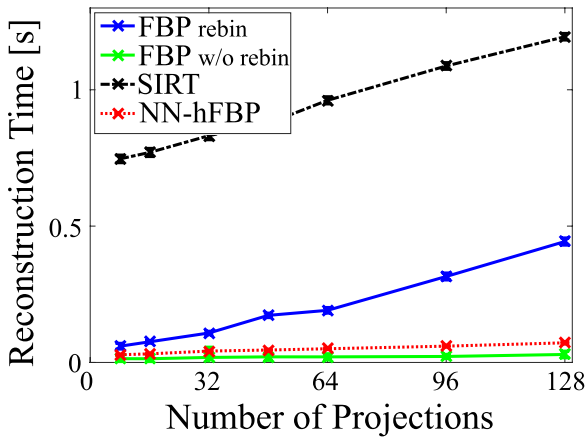
Figure 15. Reconstruction times of FBP (with and without rebinning time included), SIRT and NN-hFBP for apples with 256×256 reconstruction pixels.

Figure 16. Evaluation of the FBP, SIRT and NN-hFBP reconstruction algorithms for walnuts with the (a) global RMSE, (b) local RMSE, (c) MAD and (d) FSIM.

are shown in figure 14 for 16, 32, 64 and 128 projections. On the images made with 16 projections, many artifacts appear which make the detection of undesired regions impossible. The black regions at the outside of the apple might suggest that there are holes as well. Also for the bell pepper, not much information can be obtained with 16 projections. However, from the images obtained with 64 projections, the holes are clearly visible in the reconstruction. Despite the radial lines at the outside of the apple, this image can certainly be used for the detection of holes. Further increasing the number of projections naturally leads to a better reconstruction quality. Figure 15 shows the reconstruction times of FBP with and without the rebinning time, SIRT and NN-hFBP. NN-hFBP is 16 to 28 times faster than SIRT and 3 to 9 times faster than FBP with rebinning time but slower than FBP without the rebinning. Therefore it can be concluded that the long reconstruction time of FBP is caused by rebinning from a fan-beam to a parallel-beam geometry. The overhead time due to the rebinning step scales linearly with the number of projections. Furthermore, the training time of the NN-hFBP has not been
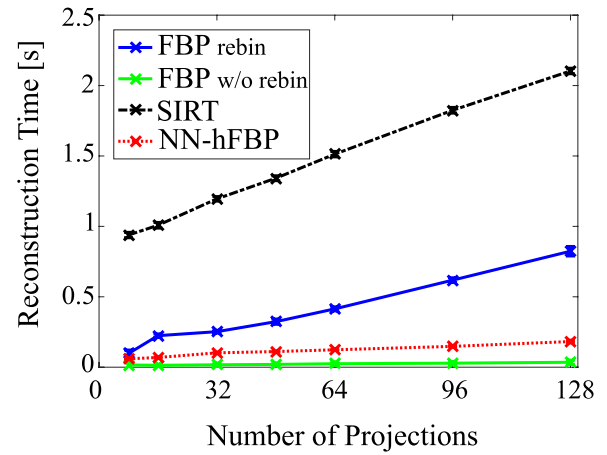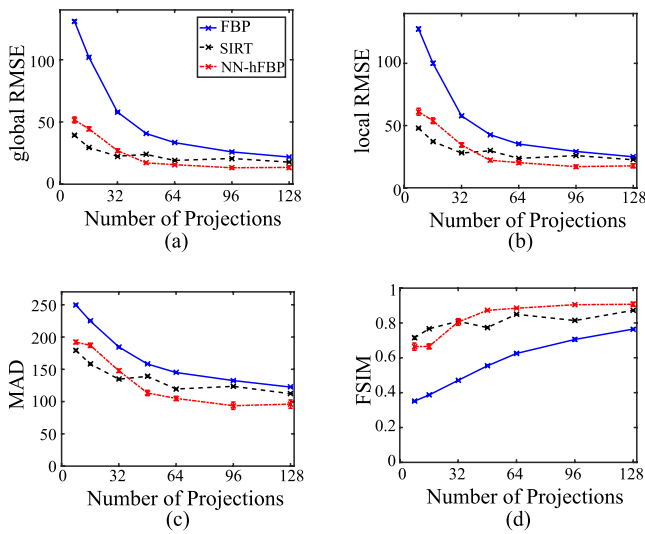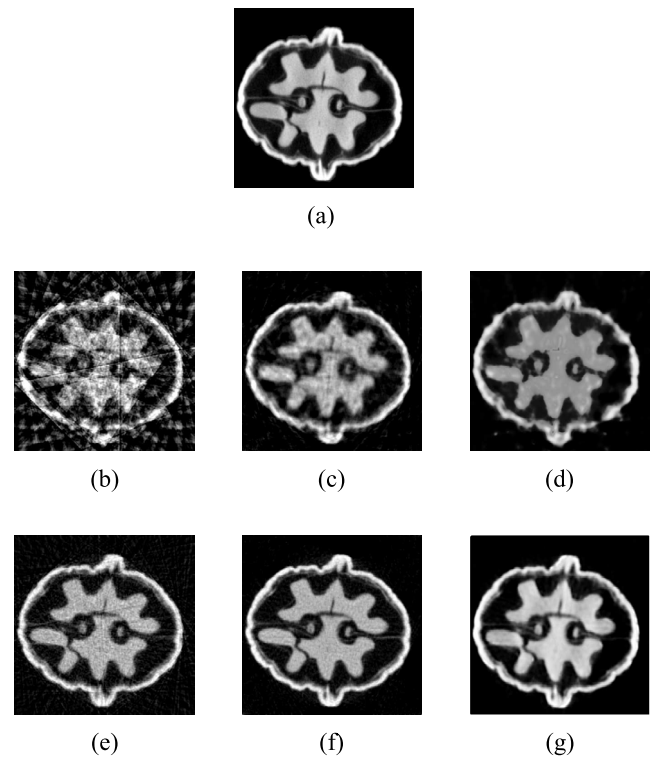

Figure 17. Reconstruction times of FBP (with and without rebinning time included), SIRT and NN-hFBP for walnuts with 400×400 reconstruction pixels.

Figure 18. Ground truth image (a) of a walnut and reconstructions made with FBP (b), SIRT (c) and NN-hFBP (d) from 32 projections and FBP (e), SIRT (f) and NN-hFBP (g) from 128 projections.

taken into account since the NN-hFBP will mainly be used for inspection of a large number of samples, for which the training phase can be done in advance. Once the network is trained, similar samples can be scanned very fast.

3.2. Real data experiment

To test the performance on real x-ray data, a mock-up was built for an inline scanning environment where the sample rotates and translates at the same time, closely mimicking an inline environment with a conveyor belt [20]. Specifications of the scanning geometry are given in table 1. The positions

at which projections were taken on the conveyor belt ranged from −250 mm to 250 mm relative to the central ray and intermittent projections were acquired at positions so that the angles were equiangularly distributed.

With this mock-up, five walnuts were scanned. For each walnut, 512 projections were taken. Based on the 512 projections, 400×400 pixel reconstructions of 30 slices around the central slices were made with the SIRT algorithm. The reconstructed images were used to train the NN-hFBP. The network was trained with 300.000 pixels chosen from 90 images obtained from three training walnuts. Validation was done on 10.000 pixels of 30 images. The results were obtained from 10 images of the last remaining walnut dataset. A subset of the available projections was used and the reconstruction quality of FBP, SIRT and NN-hFBP as a function of the number of projections was compared.

Figure 16 shows the quality of the FBP, SIRT and NN-hFBP reconstructions based on the real data acquired from the walnuts, in terms of the RMSE ((a), (b)), MAD (c) and FSIM (d). As can be observed from these plots, the NN-hFBP outperforms the FBP method, independent of the number of projections. For less than 48 projections, the SIRT algorithms creates however better reconstructions than the NN-hFBP. This might be due to the limited training data available. The reconstruction quality of the NN-hFBP is highly dependent on the amount and quality of the training data. Figure 17 shows the reconstruction time of FBP with and without rebinning time, SIRT and NN-hFBP. Here, we see again that the NN-hFBP algorithm is faster than FBP with rebinning but slower than FBP without rebinning. Despite its lower reconstruction quality than SIRT for a low number of projections, it is still much faster than SIRT. It is therefore better suited for implementation in an inline environment where speed is critical. Figure 18 shows the reconstructed walnut slices from 32 and 128 projections with the different reconstruction algorithms. The NN-hFBP manages to remove the background artifacts and increase the signal-to-noise ratio of the reconstructions (2.65, 4.25 and 4.87 for reconstructions made with 32 projections with FBP, SIRT and NN-hFBP, respectively) while preserving the important features of the walnuts.

4. Conclusion

The NN-hFBP introduced in this work is a fast reconstruction method suitable to inline inspection where only a limited number of projections are available. Our method works directly on the fan-beam data, without the need for rebinning to parallel beam data. Simulation and real data studies showed that NN-hFBP outperforms the conventional FBP with respect to image quality. NN-hFBP is an order of magnitude faster than SIRT and for at least 48 projections, it also outperforms the SIRT algorithm in terms of reconstruction

quality. The reconstruction time is approximately 200 ms for a reconstruction of 400×400 pixels from 128 projections when the forward and backprojection are calculated on an NVIDIA GeForce GTX 580 GPU with the ASTRA Toolbox.

Appendix A. hFBP algorithm for inline inspection with a constant rotation speed

(i) Discrete Hilbert transform:

$$p_H^{inl}(u_i, h_j) = \frac{\Delta_u \sqrt{t_{i,j}^2 + D^2}}{\pi}.$$
$$\sum_\tau \frac{p^{inl}(u_i - \tau, h_j)}{\tau \sqrt{(t_{i,j} - \tau)^2 + D^2}} \quad \text{(A.1)}$$

(ii) Differential by central difference:

$$p_{F_0}^{inl}(u_i, h_j) = \frac{\sqrt{D^2 + t_{i,j}^2}}{D + \omega(D\,SO - h_j t_{i,j})}.$$
$$\left(\frac{p_H^{inl}(u_i, h_{j+1}) - p_H^{inl}(u_i, h_{j-1})}{2\Delta_h} \right)$$
$$+ \frac{\sqrt{D^2 + t_{i,j}^2}(D + \omega(D^2 + t_{i,j}^2))}{D^2 + \omega D(D\,SO - h_j t_{i,j})}.$$
$$\left(\frac{p_H^{inl}(u_{i+1}, h_j) - p_H^{inl}(u_{i-1}, h_j)}{2\Delta_u} \right) \quad \text{(A.2)}$$

where Δ_h is the sampling grid interval with respect to h and Δ_u the sampling grid interval with respect to u.

(iii) Interpolation of u and h

(iv) Backprojection

The values at pixels (x, y) with polar coordinates (r, ϕ) are then reconstructed by

$$f(x, y) = \frac{\Delta_h}{8\pi} \sum_m p_{F_0}^{inl}(u_m, h_m) \quad \text{(A.3)}$$

where u_m is the detector coordinate at projection h_m where the ray through pixel (x, y) hits the detector.

Appendix B. Reformulation of the hFBP

In this appendix, we will reformulate the reconstruction formula for an inline hFBP reconstruction (11) so that it fits the form of (16):

$$f(r, \phi) = \sum_{\tau} f_1(\tau) I_1(u - \tau) + f_2(\tau) I_2(u - \tau). \quad \text{(B.1)}$$

Then it will be possible to use the hFBP in a neural network to create the NN-hFBP. To start with, after discretization, the reconstruction formula of the hFBP for inline inspection with a constant rotation speed (11) can be written in full as

$$f(r, \phi) = \frac{\Delta_h}{16\pi} \sum_{h_j \in H} \sqrt{D^2 + t_j^2}$$

$$\left[\left(\frac{(D + \omega(D^2 + t_j^2))}{D + \omega(D\ SO - h_j t_j)} \right) \frac{1}{D\Delta_u} \right.$$

$$\sum_{\tau} \left(\frac{\sqrt{(t_j^{+\Delta_u})^2 + D^2} p^{\text{inl}}(u + \Delta_u - \tau, h_j)}{\tau \sqrt{(t_j^{+\Delta_u} - \tau)^2 + D^2}} \right.$$

$$\left. - \frac{\sqrt{(t_j^{-\Delta_u})^2 + D^2} p^{\text{inl}}(u - \Delta_u - \tau, h_j)}{\tau \sqrt{(t_j^{-\Delta_u} - \tau)^2 + D^2}} \right)$$

$$+ \frac{1}{D + \omega(D\ SO - h_j t_j)} \frac{1}{\Delta_h}$$

$$\sum_{\tau} \left(\frac{\sqrt{t_{j+1}^2 + D^2} p^{\text{inl}}(u - \tau, h_{j+1})}{\tau \sqrt{(t_{j+1} - \tau)^2 + D^2}} \right.$$

$$\left. \left. - \frac{\sqrt{t_{j-1}^2 + D^2} p^{\text{inl}}(u - \tau, h_{j-1})}{\tau \sqrt{(t_{j-1} - \tau)^2 + D^2}} \right) \right] \quad \text{(B.2)}$$

where H is the set of all displacement values h, $t_j = u - h_j$, $t_j^{-\Delta_u} = u - \Delta_u - h_j$ and $t_j^{+\Delta_u} = u + \Delta_u - h_j$. To implement this reconstruction method into the NN-FBP, (B.2) is reformulated into (B.1) so that the first matrix multiplication of the ANN, the input nodes I_1 and I_2 times the weights $\boldsymbol{w_{1i}}$ and $\boldsymbol{w_{2i}}$ of figure 4, can be replaced by an hFBP reconstruction. In the NN-FBP, the same filters are used to reconstruct every pixel, independent of the angular position of the projection. Therefore, from the two terms in (B.2), two position and angle independent filters are derived:

$$f_1(\tau) = \frac{\Delta_h}{16\pi\tau D\Delta_u} \quad \text{(B.3)}$$

$$f_2(\tau) = \frac{\Delta_h}{16\pi\tau\Delta_h}. \quad \text{(B.4)}$$

The remaining input of the neural network is then defined by two *datavectors* of size n:

$$I_1(u - \tau) = \sum_{h_j \in H} \left(\frac{a(j)\, p^{\text{inl}}(u + \Delta_u - \tau, h_j)}{\sqrt{(t_j'^{+\Delta_u} - \tau)^2 + D^2}} \right.$$

$$\left. - \frac{b(j)\, p^{\text{inl}}(u - \Delta_u - \tau, h_j)}{\sqrt{(t_j^{-\Delta_u} - \tau)^2 + D^2}} \right) \quad \text{(B.5)}$$

$$I_2(u - \tau) = \sum_{h_j \in H} \left(\frac{c(j)\, p^{\text{inl}}(u - \tau, h_{j+1})}{\sqrt{(t_{j+1} - \tau)^2 + D^2}} \right.$$

$$\left. - \frac{d(j)\, p^{\text{inl}}(u - \tau, h_{j-1})}{\sqrt{(t_{j-1} - \tau)^2 + D^2}} \right) \quad \text{(B.6)}$$

with

$$a(j) = \frac{(D + \omega(D^2 + t_j^2))}{D + \omega(D\ SO - h_j t_j)} \sqrt{\frac{D^2 + t_j^2}{(t_j^{+\Delta_u})^2 + D^2}} \quad \text{(B.7)}$$

$$b(j) = \frac{(D + \omega(D^2 + t_j^2))}{D + \omega(D\ SO - h_j t_j)} \sqrt{\frac{D^2 + t_j^2}{(t_j^{-\Delta_u})^2 + D^2}} \quad \text{(B.8)}$$

$$c(j) = \frac{\sqrt{D^2 + t_j^2}\sqrt{D^2 + t_{j+1}^2}}{D + \omega(D\ SO - h_j t_j)} \quad \text{(B.9)}$$

$$d(j) = \frac{\sqrt{D^2 + t_j^2}\sqrt{D^2 + t_{j-1}^2}}{D + \omega(D\ SO - h_j t_j)}. \quad \text{(B.10)}$$

So that (B.2) can be rewritten as

$$f(r, \phi) = \sum_{\tau} f_1(\tau) I_1(u - \tau) + f_2(\tau) I_2(u - \tau). \quad \text{(B.11)}$$

To generate the data vectors for a certain pixel, first the corresponding detector pixel is defined for every projection. We further on denote this pixel as the central detector pixel. Then, the two datavectors I_1 and I_2 are generated as described by (B.5) and (B.6). For both datavectors, the projection data is shifted so that the central pixel is at the center of the detector as explained in [22] and the data from all projection angles is summed.

ORCID iDs

Eline Janssens ⓘ https://orcid.org/0000-0003-0665-5430

References

[1] Haff R P and Toyofuku N 2008 X-ray detection of defects and contaminants in the food industry *Sens. Instrum. Food Qual. Saf.* **2** 262–73

[2] Van Dael M, Lebotsa S, Herremans E, Verboven P, Sijbers J, Opara U L, Cronje P J and Nicolaï B M 2016 A segmentation and classification algorithm for online detection of internal disorders in citrus using x-ray radiographs *Postharvest Biol. Technol.* **112** 205–14

[3] Iovea M, Neagu M, Stefanescu B, Mateiasi G, Clarke A L, Vieyra M and Nicholson P I 2014 Time delay integration detector based solution for industrial x-ray NDT inspection *11th European Conf. on Non-Destructive Testing* http://www.ndt.net/events/ECNDT2014/app/content/Paper/35_Iovea.pdf

[4] An Y and Schajer G S 2014 Geometry-based CT scanner for measuring logs in sawmills *Comput. Electron. Agric.* **105** 66–73

[5] Thompson W, Lionheart W R B, Morton E, Cunningham M and Luggar R D 2015 High speed imaging of dynamic processes with a switched source x-ray CT system *Meas. Sci. Technol.* **5** 055401

[6] Warnett J M, Titarenko V, Kiraci E, Attridge A, Lionheart W R B, Withers P J and Williams M A 2016 Towards in-process x-ray CT for dimensional metrology *Meas. Sci. Technol.* **27** 035401

[7] Uroukov I S and Speller R 2015 A preliminary approach to intelligent x-ray imaging for baggage inspection at airports *Signal Process. Res.* **4** 1–11

[8] Haff R P, Pearson T C and Jackson E S 2013 One dimensional linescan x-ray detection of pits in fresh cherries *Am. J. Agric. Sci. Technol.* **1** 18–26

[9] Kwon J, Lee J and Kim W 2008 Real-time detection of foreign objects using x-ray imaging for dry food manufacturing line *IEEE Int. Symposium on Consumer Electronics* pp 1–4

[10] Maier J, Leinweber C, Sawall S, Stoschus H, Ballach F, Müller T, Hammer M, Christoph R and Kachelrieß M 2017 Simulation-based artifact correction (SBAC) for metrological computed tomography *Meas. Sci. Technol.* **28** 065011

[11] Jansson A and Pejryd L 2016 A dual-energy approach for improvement of the measurement consistency in computed tomography *Meas. Sci. Technol.* **27** 115013

[12] Bradley R S 2016 Estimation of bias and variance of measurements made from tomography scans *Meas. Sci. Technol.* **27** 095402

[13] Lifton J J, Malcolm A A and McBride J W 2016 An experimental study on the influence of scatter and beam hardening in x-ray CT for dimensional metrology *Meas. Sci. Technol.* **27** 015007

[14] du Plessis A, le Roux S G and Guelpa A 2016 Comparison of medical and industrial x-ray computed tomography for non-destructive testing *CSNDT&E* **6** 17–25

[15] Sidky E Y, Zou Y and Pan X 2005 Volume image reconstruction from a straight-line source trajectory *IEEE Nuclear Science Symp. Conf. Record* vol 5 pp 2441–4

[16] Zhang L, Gao H, Cheng J, Chen Z and Xing Y 2008 Feasibility study of applying x-ray tomographic imaging with a straight-line trajectory scan to security inspection *IEEE Nuclear Science Symp. Conf. Record* pp 1274–7

[17] Zhang H, Wang L, Yan B, Li L, Xi X and Lu L 2013 Image reconstruction based on total-variation minimization and alternating direction method in linear scan computed tomography *Chin. Phys.* B **22** 078701

[18] Janssens E, Alves Pereira L F, De Beenhouwer J, Tsang I R, Van Dael M, Verboven P, Nicolai B and Sijbers J 2016 Fast inline inspection by neural network based filtered backprojection: application to apple inspection *CSNDT&E* **6** 14–20

[19] Alves Pereira L F, Janssens E, Van Dael M, Verboven P, Nicolai B, Cavalcanti G D C, Tsang I R and Sijbers J 2016 Fast x-ray computed tomography via image completion *6th Conf. on Industrial Computed Tomography (Wels, Austria)* (www.ndt.net/?id=18772)

[20] De Schryver T, Dhaene J, Dierick M, Boone M N, Janssens E, Sijbers J, Van Dael M, Verboven P, Nicolai B and Van Hoorebeke L 2016 In-line NDT with x-ray CT combining sample rotation and translation *NDT&E Int.* **84** 89–98

[21] Pelt D M, Sijbers J and Batenburg K J 2013 Fast tomographic reconstruction from highly limited data using artificial neural networks *1st Int. Conf. on Tomography of Materials and Structures* pp 109–12 (https://repository.uantwerpen.be/docman/irua/ff904f/3edb0ea8.pdf)

[22] Pelt D M and Batenburg K J 2013 Fast tomographic reconstruction from limited data using artificial neural networks *IEEE Trans. Image Process.* **22** 5238–51

[23] You J and Zeng G L 2007 Hilbert transform based FBP algorithm for fan-beam CT full and partial scans *IEEE Trans. Med. Imaging* **26** 190–9

[24] You J, Zeng G L and Liang Z 2005 FBP algorithms for attenuated fan-beam projections *Inverse Problems* **21** 1801

[25] Zhang L, Zhang L, Mou X and Zhang D 2011 FSIM: a feature similarity index for image quality assessment *IEEE Trans. Image Process.* **20** 2378–86

[26] Larson E C and Chandler D M 2010 Most apparent distortion: full-reference image quality assessment and the role of strategy *J. Electron. Imaging* **19** 011006

[27] van Aarle W, Palenstijn W J, Cant J, Janssens E, Bleichrodt F, Dabravolski A, De Beenhouwer J, Batenburg K J and Sijbers J 2016 Fast and flexible x-ray tomography using the ASTRA toolbox *Opt. Express* **24** 25129–47

[28] Palenstijn W J, Batenburg K J and Sijbers J 2011 Performance improvements for iterative electron tomography reconstruction using graphics processing units (GPUs) *J. Struct. Biol.* **176** 250–3

[29] van Aarle W, Palenstijn W J, De Beenhouwer J, Altantzis T, Bals S, Batenburg K J and Sijbers J 2015 The ASTRA toolbox: a platform for advanced algorithm development in electron tomography *Ultramicroscopy* **157** 35–47