Extreme sparse X-ray computed laminography via convolutional neural networks

Extreme Sparse X-ray Computed Laminography Via Convolutional Neural Networks

Luis F. Alves Pereira
Univ. Federal do Agreste de Pernambuco
Garanhuns, Brazil
luis-filipe.pereira@ufape.edu.br

Jan De Beenhouwer
imec-Visionlab
University of Antwerp
Antwerp, Belgium
jan.debeenhouwer@uantwerpen.be

Johann Kastner
University of Applied Sciences Upper Austria
Wels, Austria
johann.kastner@fh-ooe.at

Jan Sijbers
imec-Visionlab
University of Antwerp
Antwerp, Belgium
jan.sijbers@uantwerpen.be

***Abstract*—X-ray Computed Laminography (CL) is a well-known computed tomography technique to image the internal structure of flat objects. High-quality CL imaging requires, however, a large number of X-ray projections, resulting in long acquisition times. Reducing the number of acquired projections allows to speed up the acquisition process but decreases the quality of the reconstructed images. In this work, we investigate the use of Convolutional Neural Networks for processing volumes reconstructed from only four X-ray projections acquired at an inline CL scanning setup.**

***Index Terms*—Computed Laminography, Convolutional Neural Networks, Deep Learning, 3D imaging**

I. Introduction

The number of industrial applications of Computed Tomography (CT) is increasing rapidly [1]. However, for imaging flat objects such as integrated circuits, laminate, or paintings, conventional CT is not suited because projections cannot be acquired from all directions [2]–[4]. In such cases, Computed Laminography (CL) can be applied [5], [6]. Fig. 1 illustrates the scanning setup of a regular CL imaging process where the object rotation axis is no longer perpendicular to the optical axis as in conventional CT, but slightly inclined ($\theta < \frac{\pi}{2}$).

Conventional CL reconstruction techniques usually require hundreds of projections to generate accurate images [7], [8]. The acquisition of such an amount of projections implies a prohibitively long scanning time for industrial applications. By limiting the number of X-ray projections acquired during a CL scan, not only the X-ray dose applied to the scanned object can be reduced, but also the acquisition and processing time can be substantially shortened. Unfortunately, when the number of X-ray projections becomes low, streaking artefacts will appear in the reconstructed CL image [9]. For instance, Park *et al.* [10] showed CL reconstructions obtained using from 12 to 60 X-ray projections, which revealed clear reconstruction artefacts.

This work was supported by the Comet-Project (grant ID: 871974) "Photonic Sensing for Smarter Processes" financed by FFG and the federal government of Upper Austria and Styria, as well as by the FWO-SBO project MetroFlex (S004217N).

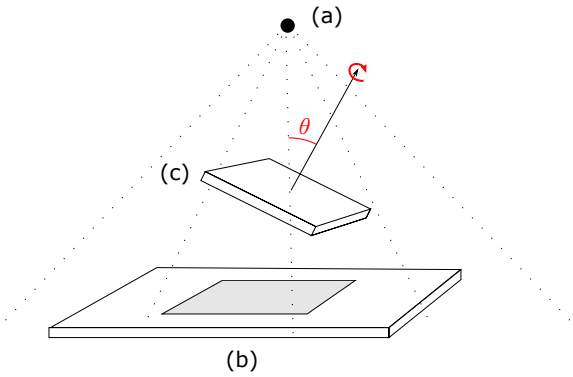


Fig. 1. Scheme of the conventional CL scanning setup: both X-ray source (a) and detector (b) remain static while the object (c) rotates around the axis, which is inclined with respect to the central X-ray path by an angle $0 \leq \theta \leq \frac{\pi}{2}$.

To overcome sparse sampling artefacts, prior knowledge about the sample to be reconstructed can be invoked. Abbas *et al.* showed how Total Variation minimization based reconstruction can successfully generate high quality CL images from 40 X-ray projection views [7], [8]. Batenburg *et al.* [11] showed that, using DART [12], it is possible to obtain high quality reconstructions from only 10 X-ray projections. However, DART strongly relies on the validity of the discreteness of the material.

In recent years, many Convolutional Neural Networks (CNNs) have been proposed to improve the quality of low-dose medical CT images: Han and Ye [13] presented solutions based on 2D Unets; Xie *et al.* [14] applied the 2D GoogLeNet architecture; Chen *et al.* [15] proposed the 2D Residual Encoder-Decoder CNN; Zhao *et al.* [16] introduced the DnCNN model.

In this work, we evaluate three 3D CNNs in improving the

quality of images reconstructed using an extreme sparse-view CL setup. The performance of each CNN is evaluated on a realistically simulated inline CL scanning setup.

This paper is organized as follows: Section II describes the inline scanning setup and the architecture of the CNNs used, Section III presents the experimental setup designed to evaluate the methods, Section IV discusses the results obtained, and Section V concludes the paper.

II. METHODS

A. Inline Scanning Setup

The inline scanning setup [10], illustrated in Fig. 2, is composed of a static X-ray source (a) and detector (b) for imaging flat objects moving parallel to the detector plane with velocity $\vec{v}$. The projections P_1, P_2, and P_n are generated at the positions (1), (2), and (n), respectively.

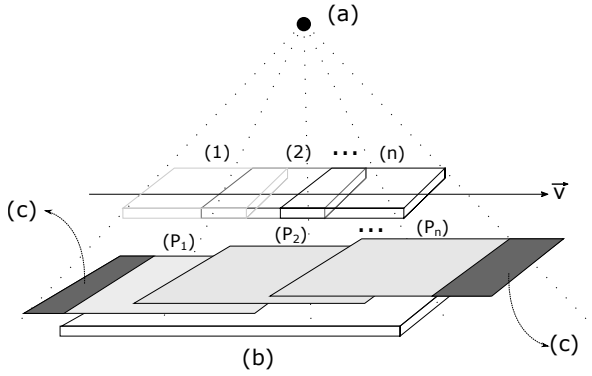


Fig. 2. Scheme of the evaluated inline scanning setup for CL, where a fixed pair of X-ray source (a) and detector (b) is used for imaging flat objects moving parallel to the detector plane. This geometry leads to projection truncation near the detector borders (c).

In this inline scanning setup, besides the limited angular view, the projections acquired near the detector borders are truncated. In fact, Park *et al.* [10] proposed a data truncation correction when working with such inline geometry. In our approach, however, no data preprocessing is required once regular CL reconstructions from truncated data are used to feed our CNNs.

B. The 3D Convolutional Neural Networks

Using a CNN model $f_\theta(\cdot)$, we are looking for the set of parameters θ that minimizes:

$$\arg\min_{\theta} ||f_\theta(\mathbf{r}_c) - \mathbf{r}_s||_2^2 \tag{1}$$

where $\mathbf{r}_c \in \mathbb{R}^{i \times j \times k}$ is a volume reconstructed from a limited set of projections acquired in the inline scanning setup (as shown in Fig. 2), and $\mathbf{r}_s \in \mathbb{R}^{i \times j \times k}$ its respective ground truth that can be generated in the fully sampled conventional CL setup (as shown in Fig. 1).

In this work, we evaluate three 3D CNN models based on 2D models from the state-of-the-art of low-dose medical CT [13]–[15]: *(i)* a 3D Residual Encoder-Decoder (RED-CNN), *(ii)* a 3D GoogLeNet, and *(iii)* a 3D Unet. At each network, the 2D convolutions from the regular architectures were replaced by 3D convolutions in order to incorporate correlations between adjacent slices of a 3D volume [23].

The following subsections present a detailed description of each of those architectures.

1) The 3D Residual Encoder-Decoder CNN: this architecture is composed of 5 convolutional layers in an encoder stage, and 5 deconvolutional layers in a decoder stage. By avoiding data downsampling and upsampling, the image's structural information is kept [15]. The complete network's architecture is described in Table I. Additionally, three residual connections link the blocks 3 to 4, 2 to 5, and 1 to 6.

TABLE I

DESCRIPTION OF THE 3D RED-CNN USED IN THIS WORK. THE PADDING VALUE IS 0, THE STRIDE VALUE IS 1 FOR ALL LAYERS, AND EACH CONVOLUTION IS FOLLOWED BY THE RELU ACTIVATION FUNCTION. THE NETWORK'S INPUT SIZE IS $1 \times 16 \times 64 \times 64$.

Block	Layer	Kernel	Output size
1	Conv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
2	Conv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
3	Conv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
4	Deconv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
5	Deconv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
	Deconv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
6	Deconv 3D	$3 \times 3 \times 3$	$96 \times 16 \times 64 \times 64$
	Deconv 3D	$3 \times 3 \times 3$	$1 \times 16 \times 64 \times 64$

2) The 3D GoogLeNet: this architecture introduces the inception module, which combines multiscale convolutional kernels in a single layer [14]. This network is based on the stacking of 8 inception modules in a row. Its complete description is presented in Table II.

TABLE II

DESCRIPTION OF THE 3D GOOGLENET USED IN THIS WORK. EACH CONVOLUTIONAL AND INCEPTION LAYERS ARE FOLLOWED BY THE RELU ACTIVATION FUNCTION. THE NETWORK'S INPUT SIZE IS $1 \times 16 \times 64 \times 64$.

#	Layer	Kernel	Output size
1	Conv 3D	$3 \times 3 \times 3$	$16 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$64 \times 16 \times 64 \times 64$
2	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
	Inception	$1,3,5 \times 1,3,5 \times 1,3,5$	$192 \times 16 \times 64 \times 64$
3	Conv 3D	$3 \times 3 \times 3$	$64 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$1 \times 16 \times 64 \times 64$

3) The 3D Unet: this architecture is composed of convolutional and deconvolutional layers within stages for data compression and decompression. By introducing downsampling and upsampling layers, the network's receptive field is enlarged [13]. The complete network's architecture is described in Table III. Additionally, four residual connections link the blocks 1 to 9, 2 to 8, 3 to 7, and 4 to 6.

TABLE III
DESCRIPTION OF THE 3D UNET USED IN THIS WORK. THE PADDING VALUE IS 1 FOR ALL LAYERS, STRIDE VALUE IS 2 FOR MAXPOOL AND DECONV LAYERS, AND EACH CONVOLUTION IS FOLLOWED BY THE RELU ACTIVATION FUNCTION. THE NETWORK'S INPUT SIZE IS $1 \times 16 \times 64 \times 64$.

#	Layer	Kernel	Output size
1	Conv 3D	$3 \times 3 \times 3$	$64 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$64 \times 16 \times 64 \times 64$
	MaxPool	$2 \times 2 \times 2$	$64 \times 8 \times 32 \times 32$
2	Conv 3D	$3 \times 3 \times 3$	$64 \times 8 \times 32 \times 32$
	Conv 3D	$3 \times 3 \times 3$	$128 \times 8 \times 32 \times 32$
	Conv 3D	$3 \times 3 \times 3$	$128 \times 8 \times 32 \times 32$
	MaxPool	$2 \times 2 \times 2$	$128 \times 4 \times 16 \times 16$
3	Conv 3D	$3 \times 3 \times 3$	$128 \times 4 \times 16 \times 16$
	Conv 3D	$3 \times 3 \times 3$	$256 \times 4 \times 16 \times 16$
	Conv 3D	$3 \times 3 \times 3$	$256 \times 4 \times 16 \times 16$
	MaxPool	$2 \times 2 \times 2$	$256 \times 2 \times 8 \times 8$
4	Conv 3D	$3 \times 3 \times 3$	$256 \times 2 \times 8 \times 8$
	Conv 3D	$3 \times 3 \times 3$	$512 \times 2 \times 8 \times 8$
	Conv 3D	$3 \times 3 \times 3$	$512 \times 2 \times 8 \times 8$
	MaxPool	$2 \times 2 \times 2$	$512 \times 1 \times 4 \times 4$
5	Conv 3D	$3 \times 3 \times 3$	$512 \times 1 \times 4 \times 4$
	Conv 3D	$3 \times 3 \times 3$	$1024 \times 1 \times 4 \times 4$
	Conv 3D	$3 \times 3 \times 3$	$512 \times 1 \times 4 \times 4$
6	Deconv 3D	$2 \times 2 \times 2$	$512 \times 2 \times 8 \times 8$
	Conv 3D	$3 \times 3 \times 3$	$512 \times 2 \times 8 \times 8$
	Conv 3D	$3 \times 3 \times 3$	$512 \times 2 \times 8 \times 8$
	Conv 3D	$3 \times 3 \times 3$	$256 \times 2 \times 8 \times 8$
7	Deconv 3D	$2 \times 2 \times 2$	$256 \times 4 \times 16 \times 16$
	Conv 3D	$3 \times 3 \times 3$	$256 \times 4 \times 16 \times 16$
	Conv 3D	$3 \times 3 \times 3$	$256 \times 4 \times 16 \times 16$
	Conv 3D	$3 \times 3 \times 3$	$128 \times 4 \times 16 \times 16$
8	Deconv 3D	$2 \times 2 \times 2$	$128 \times 8 \times 32 \times 32$
	Conv 3D	$3 \times 3 \times 3$	$128 \times 8 \times 32 \times 32$
	Conv 3D	$3 \times 3 \times 3$	$128 \times 8 \times 32 \times 32$
	Conv 3D	$3 \times 3 \times 3$	$64 \times 8 \times 32 \times 32$
9	Deconv 3D	$2 \times 2 \times 2$	$64 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$64 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$64 \times 16 \times 64 \times 64$
	Conv 3D	$3 \times 3 \times 3$	$1 \times 16 \times 64 \times 64$

III. EXPERIMENTS

To evaluate the performance of the CNN based CL reconstruction methods, we created a set of a set of 4,000 flat objects of size $10 \times 64 \times 64$ that contain a random number of ellipsoids to represent pores[1], as illustrated in Fig. 3. For each of these phantom objects, four projections were generated using the ASTRA toolbox [27] from an inline acquisition setup as shown in Fig. 2. Our simulations include a cone-beam angle of $30°$, a 64×64 detector, a source-object distance of 120 unit length, and a lateral distance between two consecutive projections of 21 unit length. Furthermore, the first and the fourth projections were truncated by 50% (as shown in Fig. 2).

From the sparse-view CL projection data, 3D volumes were reconstructed using conventional SIRT [24]. SIRT reconstructions were chosen as the input of the 3D CNNs instead of reconstructions using the widely used Filtered Back-Projection (FBP) because SIRT provides better results in situations that include noise, sparse-view, and projection truncation [25].

[1]Our dataset and code are available at: https://github.com/luisfilipeap/Extreme-Sparse-X-ray-Computed-Laminography-Via-Convolutional-Neural-Networks-

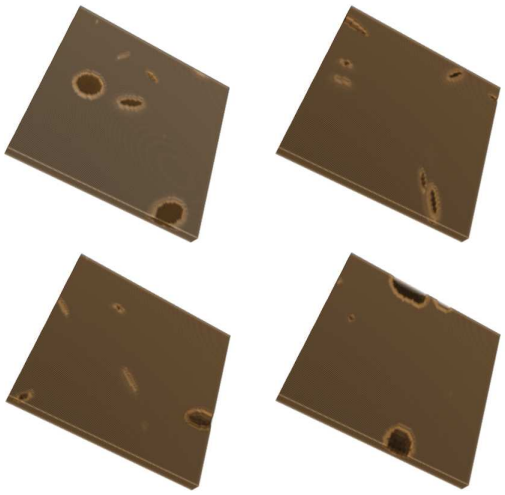


Fig. 3. Four samples of the simulated flat plates used in our experiments.

Next, the performance of the CNN based CL reconstruction methods was tested on sparse-view datasets simulated with various incident flux I^0 at the X-ray tube.

Experiments were run with an Intel Core i7 3.2 GHz, 64 GB RAM, equipped with a GeForce GTX 1070. The 3D CNNs were built using the PyTorch library [28].

IV. RESULTS AND DISCUSSION

Fig. 4 shows axial views of two CL volumes generated by the RED-CNN 3D in the first column, the GoogLeNet 3D in the second column, and the Unet 3D in the third column, when the radiation flux $I^0 \rightarrow \infty$. Ground truth images are presented in the fourth column. Fig. 5 shows coronal views of a reconstructed object. It can be seen that Unet 3D provides the best accuracy and image contrast.

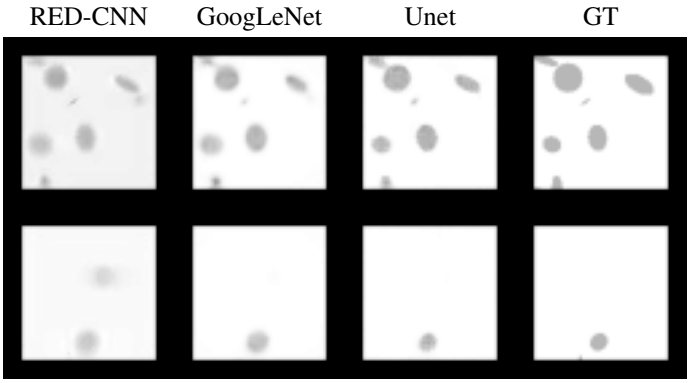


Fig. 4. Axial views of results generated by the RED-CNN 3D (column 1), the GoogLeNet 3D (column 2), and the Unet 3D (column 3), using a radiation flux $I^0 \rightarrow \infty$. Ground truth images are shown in column 4.

Results of a quantitative analysis measuring PSNR and SSIM were conducted with the test set comprising 1,000 unseen samples. For $I^0 \rightarrow \infty$, the average PSNR obtained by the Unet 3D (29.7) was 11% greater than that obtained by the GoogLeNet 3D (26.6), and 20% greater than that obtained by the RED-CNN 3D (24.7). The average SSIM obtained by

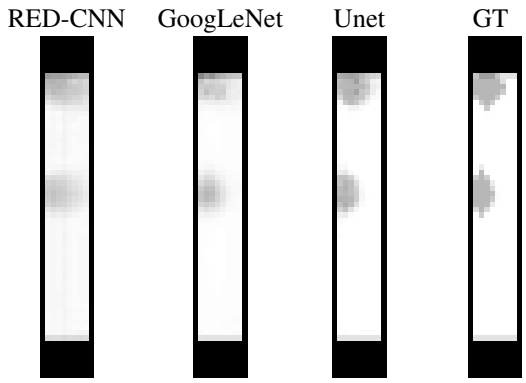


Fig. 5. Coronal views of results generated by the RED-CNN 3D (column 1), the GoogLeNet 3D (column 2), and the Unet 3D CNN (column 3), using a radiation flux $I^0 \rightarrow \infty$. Ground truth images are shown in column 4.

the Unet 3D (0.894) was 16% greater than that obtained by the GoogLeNet 3D (0.768), and 75% greater than that obtained by the RED-CNN 3D (0.504). Those results suggest the adoption of the Unet 3D to enhance SIRT volumes in situations where the projections are not only sparse but also truncated. This is a significant advantage compared to previous methods that require additional preprocessing stages to correct truncated projections [10].

Fig. 6 shows samples of CNN outputs for $10^4 \leq I^0 \leq 10^7$. It can be seen that experiments using $I^0 \leq 10^5$ produce unrecognizable results for every method evaluated. Furthermore, Fig. 7 shows the relation between SSIM and PSNR as I^0 changes. It is shown that, for $I^0 = 10^7$, both Unet and GoogLeNet present PSNR and SSIM values comparable to those obtained at the previous experiment where $I^0 \rightarrow \infty$. For $I^0 \geq 10^6$, only the Unet can provide an average SSIM greater than 0.6, and a PSNR greater than 27.

V. CONCLUSIONS

Conventional tomographic reconstruction of images from sparse-view CL suffer from severe undersampling artefacts. In this work we show, by means of simulation experiments, that the Unet 3D based CL reconstruction method can produce high quality images in terms of PSNR and SSIM. Among the three architectures evaluated, the Unet 3D provided the best results on mapping SIRT reconstructions into an enhanced CL volume. Furthermore, The Unet 3D presented outputs with recognizable features with a photon count as low as 10^6. The good performance of Unet can be explained by the data compression and decompression that enlarge the network's receptive field.

It is expected that the performance of those CNN based solutions behaves similarly with bigger reconstruction volumes. As the CNN input size increases, more computational power is required (especially GPU memory) during the training stage. This issue may be overcome by introducing powerful hardware devices or using of 3D patches in the training stage.

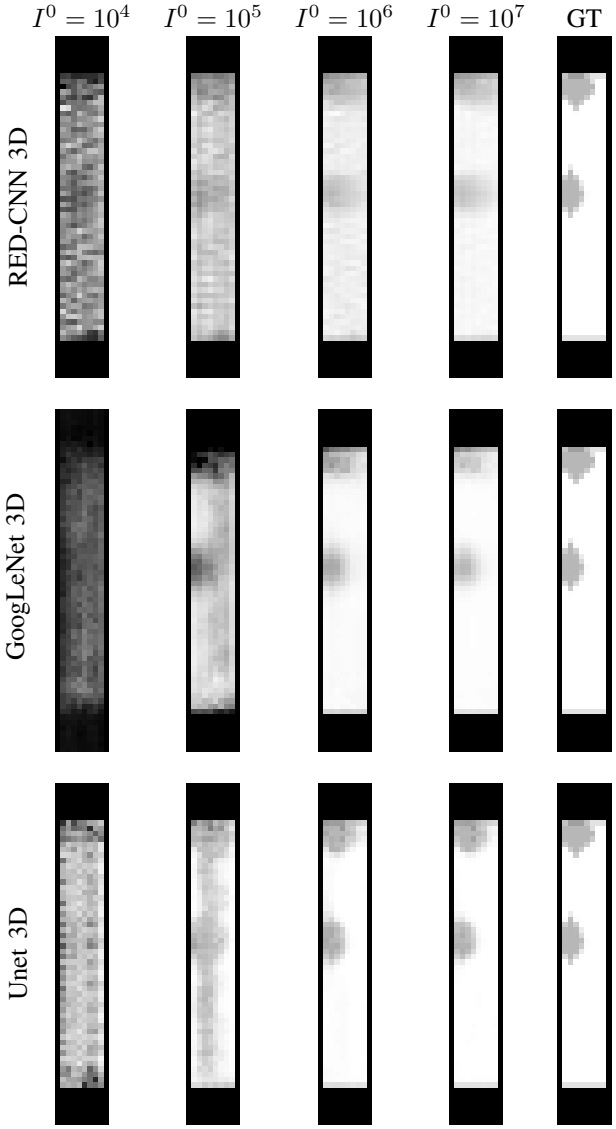
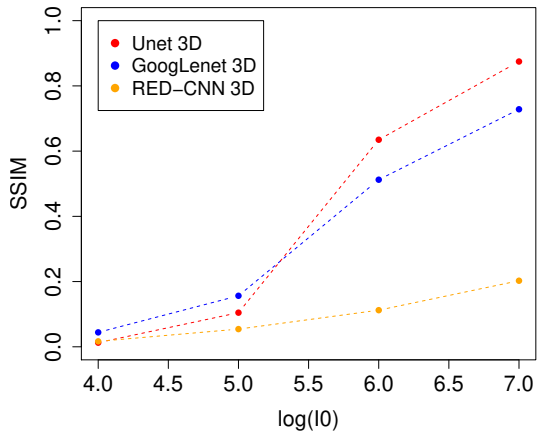
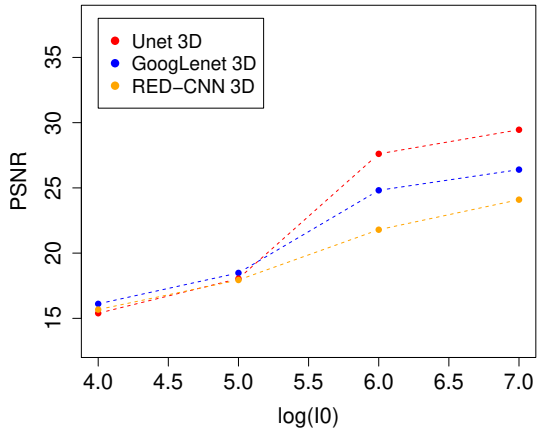


Fig. 6. Coronal views of results generated by the RED-CNN 3D (row 1), the GoogLeNet 3D (row 2), and the Unet 3D (row 3) for different levels of radiation flux I^0 at each column. Ground truth images are shown in the last column.



(a)



(b)

Fig. 7. Relation between the average SSIM (a) and PSNR (b) as I^0 changes from 10^4 to 10^7 while different CNNs are used.

REFERENCES

[1] L. De Chiffre, S. Carmignato, J. P. Kruth, R. Schmitt, and A. Weckenmann, "Industrial applications of computed tomography", CIRP Conference Proceedings, 2014, pp.655-677.

[2] F. Xu, L. Helfen, T. Baumbach, and H. Suhonen, "Comparison of image quality in computed laminography and tomography", Optics Express, 2012, pp. 794–806.

[3] L. Helfen, A. Myagotin, P. Mikulík, P. Pernot, A. Voropaev, M. Elyyan, M. Di Michiel, J. Baruchel, and T. Baumbach, "On the implementation of computed laminography using synchrotron radiation", Review of Scientific Instruments, 2011.

[4] M. Yang, Z. Li, L. Liang, X. Li, W. Liu, and Z. Gui, "Automatic measurement of rotation center for laminography scanning system without dedicated phantoms", Journal of Electronic Imaging, 2014.

[5] R. J. Kline, "Multiscale 3D X-ray imaging", Nature Electronics, 2019, pp. 235–436.

[6] S. Legrand, F. Vanmeert, G. Van der Snickt, M. Alfeld, W. De Nolf, J. Dik, and K. Janssens, "Examination of historical paintings by state-of-the-art hyperspectral imaging methods: from scanning infra-red spectroscopy to computed X-ray laminography", Nature Publishing Group, 2014, pp. 13.

[7] S. Abbas, M. Park, J. Min, H. K. Kim, S. Cho, "Sparse-view computed laminography with a spherical sinusoidal scan for nondestructive testing", Optics Express, 2014, pp. 17745–17755.

[8] S. Abbas, M. Park, and S. Cho, "Fast and low-dose computed laminography using compressive sensing based technique", AIP Conference Proceedings, 2015, pp. 586–590.

[9] M. Park, G. Cho, J. Park, and S. Cho, "Noncircular scanning trajectory for sparse-view computed laminography", 6th Conference on Industrial Computed Tomography, 2016.

[10] M. Park, M. Lee, K. Kim, S. Hwang, S. Y. Jin, H. Kim, S. Cho, and G. Cho, "Scanning Parameter Optimization for Inspection of Wedling Defects in X-ray Linear Laminography", International Conference on Industrial Computed Tomography, 2017.

[11] K. J. Batenburg, W. J. Palenstijn, and J. Sijbers, "3D imaging of semiconductor components by discrete laminography", AIP Conference Proceedings, 2014, pp. 168–179.

[12] K. J. Batenburg, and J. Sijbers, "DART: a practical reconstruction algorithm for discrete tomography", IEEE Transactions on Image Processing, 2011, pp. 2542–2553.

[13] Y. Han, and J. C. Ye, "Framing U-Net via deep convolutional framelets: Application to sparse-view CT", IEEE transactions on medical imaging, 2018, pp. 1418-1429.

[14] S. Xie, X. Zheng, Y. Chen, L. Xie, J. Liu, Y. Zhang, J. Yan, H. Zhu, and Y. Hu, "Artifact removal using improved GoogLeNet for sparse-view CT reconstruction", Scientific reports, 2018, pp.1-9.

[15] H. Chen, Y. Zhang, M. K. Kalra, F. Lin, Y. Chen, P. Liao, J. Zhou, and G. Wang, "Low-dose CT with a residual encoder-decoder convolutional neural network", IEEE transactions on medical imaging, 2017, pp.2524-2535.

[16] T. Zhao, M. McNitt-Gray, and D. Ruan, "A convolutional neural network for ultra-low-dose CT denoising and emphysema screening" Medical Physics, 2019, pp.3941-3950.

[17] O. Ronneberger, P. Fischer, and T. Brox. "U-net: Convolutional networks for biomedical image segmentation", International Conference on Medical image computing and computer-assisted intervention, 2015, pp. 234-241.

[18] J. Wang, H. Lu, Z. Liang, D. Eremina, G. Zhang, S. Wang, J. Chen, J. and Manzione, "An experimental study on the noise properties of x-ray CT sinogram data in Radon space", Physics in Medicine Biology, 2008, p.3327.

[19] N. Bevins, T. Szczykutowicz, M. P. Supanich, "TU-C-103-06: A Simple Method for Simulating Reduced-Dose Images for Evaluation of Clinical CT Protocols", Medical Physics, 2013, pp. 437-437.

[20] D. Zeng, J. Huang, Z. Bian, S. Niu, H. Zhang, Q. Feng, Z. Liang, and J. Ma, "A simple low-dose x-ray CT simulation from high-dose scan. IEEE transactions on nuclear science", 2015, pp.2226-2233.

[21] M. Gholizadeh-Ansari, J. Alirezaie, and P. Babyn, "Deep learning for low-dose CT denoising using perceptual loss and edge detection layer", Journal of digital imaging, 2019, pp.1-12.

[22] J. k. Hwan, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging", IEEE Transactions on Image Processing, 2017, pp. 4509–4522.

[23] H. Shan, Y. Zhang, Q. Yang, U. Kruger, M. K. Kalra, M. K., L. Sun, W. Cong, and G. Wang, "3D convolutional encoder-decoder network for low-dose CT via transfer learning from a 2D trained network", IEEE Transactions on Medical Imaging, 2018, pp. 1522–1534.

[24] R. Gordon, R. Bender, G. T. Herman, "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography", Journal of Theoretical Biology, 1970, pp. 471–481.

[25] C. Mu, and C. Park, C., "Sparse filtered SIRT for electron tomography", Pattern Recognition, 2020, p.107253.

[26] E. Y. Sidky, and X. Pan, "Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization", Physics in Medicine & Biology, 2008, pp. 4777.

[27] W. van Aarle, W. J. Palenstijn, J. Cant, E. Janssens, F. Bleichrodt, A. Dabravolski, J. De Beenhouwer, K. J. Batenburg, J. Sijbers, "Fast and flexible X-ray tomography using the ASTRA toolbox", Optics Express, pp. 25129–25147.

[28] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch", NIPS-W, 2017.