Deep-Neural-Network-Based Sinogram Synthesis for Sparse-View CT Image Reconstruction

Hoyeon Lee, Jongha Lee, Hyeongseok Kim, Byungchul Cho, and Seungryong Cho, *Senior Member, IEEE*

***Abstract*—Recently, a number of approaches to low-dose computed tomography (CT) have been developed and deployed in commercialized CT scanners. Tube current reduction is perhaps the most actively explored technology with advanced image reconstruction algorithms. Sparse data sampling is another viable option to the low-dose CT, and sparse-view CT has been particularly of interest among the researchers in CT community. Since analytic image reconstruction algorithms would lead to severe image artifacts, various iterative algorithms have been developed for reconstructing images from sparsely view-sampled projection data. However, iterative algorithms take much longer computation time than the analytic algorithms, and images are usually prone to different types of image artifacts that heavily depend on the reconstruction parameters. Interpolation methods have also been utilized to fill the missing data in the sinogram of sparse-view CT thus providing synthetically full data for analytic image reconstruction. In this paper, we introduce a deep-neural-network-enabled sinogram synthesis method for sparse-view CT, and show its outperformance to the existing interpolation methods and also to the iterative image reconstruction approach.**

***Index Terms*—Deep learning, low-dose computed tomography (CT), sparse-view CT, view interpolation.**

I. Introduction

With increased use of X-ray computed tomography (CT) in clinics, potential radiation hazard has been alarmed [1], [2]. There have been developed a host of approaches toward low-dose CT imaging that include reducing or modulating the tube current, optimal selection of the tube voltage [3], [4], and sparse data sampling [5]–[7] to name a few. Sparse data sampling approach is in contrast with the tube current reduction since the former recruits smaller number of ray measurements with a lower noise level of the measured data than the latter. Sparse-view sampling, where the X-ray power is supposed to be turned on-and-off repeatedly, has been actively investigated as a realization of sparse data sampling although its translation to the commercialized diagnostic CT scanners has yet to come. Image reconstruction from sparsely sampled data constitutes a unique, ill-posed inverse problem in CT, and the compressed-sensing-inspired algorithms have been developed to deal with this problem. Minimizing image sparsity such as image total-variation under the constraints of data fidelity and image non-negativity has been searched for in various optimization solver frameworks [8]–[10]. Exploiting image sparsity in such iterative image reconstruction approaches, however, may lead to undesirable image artifacts that heavily depend on the reconstruction algorithm parameters compared to the analytically reconstructed images from fully sampled data. Additionally, the minimal amount of data that guarantees clinically acceptable image quality in various imaging tasks with varying degrees of underlying image sparsity should be carefully determined. The computation time, even though it may not constitute a critical issue with advanced acceleration techniques and parallel computing power, can still be a burden.

Direct application of analytic image reconstruction algorithm such as filtered-backprojection (FBP) to the sparse-view data would lead to images with poor quality and severe streak artifacts. Attempts have been made to synthesize the missing view data so that the full data can be fed into the analytic image reconstruction engine. An interpolation-based data synthesis in the sinogram space is a straightforward example. Various approaches have been developed for synthesizing sinogram data: linear interpolation method [11], a principal component analysis-based method [12], a partial differential equation-based method [13], a frequency consistency condition-based method [14], intensity-based directional interpolation method [15], [16], dictionary learning-based method [17], [18], and some combinatorial methods [19]–[21]. For those interpolation approaches, image reconstruction results would highly depend on the restoring capability of the employed interpolation method. Greatly inspired by the recent progresses of machine learning techniques, we propose

Manuscript received March 1, 2018; revised May 15, 2018 and July 15, 2018; accepted August 21, 2018. Date of publication August 31, 2018; date of current version March 1, 2019. This work was supported in part by the Korean National Research Foundation under Grant NRF-2016M2A2A9A03913610, Grant NRF-2016M3A9E9941837, and Grant NRF-2017M2A2A4A05065897, in part by KEIT under Grant 10051357, in part by KUSTAR-KAIST Institute, KAIST, South Korea, and in part by the NST grant CAP-13-3-KERI. *(Corresponding author: Seungryong Cho.)*

H. Lee and H. Kim are with the Department of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Engineering, Daejeon 34141, South Korea (e-mail: leehoy@kaist.ac.kr; kimhs369@kaist.ac.kr).

J. Lee is with the Department of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Engineering, Daejeon 34141, South Korea, and also with the Medical Imaging Research and Development Group, Health and Medical Equipment Business, Samsung Electronics, Suwon 16677, South Korea (e-mail: jongha.lee@kaist.ac.kr).

B. Cho is with the Department of Radiation Oncology, Asan Medical Center, Seoul 05505, South Korea (e-mail: cho.byungchul@gmail.com).

S. Cho is with the Department of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Engineering, Daejeon 34141, South Korea, and also with KI for Health Science and Technology and ITC, Daejeon 34141, South Korea (e-mail: scho@kaist.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRPMS.2018.2867611

in this paper to use a deep-neural-network for synthesizing the missing data in the sparse-view sinograms.

Machine learning has been actively used for classification tasks such as face recognition [22], tumor classification [23], [24], and image segmentation [25]. Traditional machine learning techniques train the network so that a human-defined "features" can be optimally computed in a given task such as classification and segmentation. Therefore, the performance of the machine learning heavily depends on the features that a user defines. Deep learning, on the other hand, automatically learns its own features during the network training period. The convolution layer and the activation function in the deep neural network can recruit higher dimensional features, of which a human does not have intuitive analogs, and can enhance the performance of the neural network for a variety of tasks. Thanks to such enriched capabilities of the deep learning, its applications have explosively expanded to diverse fields: for example, super-resolution image processing [26]–[28], visual questioning [29], generating artistic features-added photo [30], and generating new images from the random data [31]. Deep learning-based approaches have also been applied to image reconstruction for low-dose CT including low tube-current CT imaging [32]–[34] and sparse-view CT imaging [35], [36]. While those approaches exploit the deep-neural-network in the image domain of CT rather than in the sinogram domain, this paper focuses on restoring the missing data in the sinogram domain so that one can reconstruct images by use of the well-established reconstruction algorithms in practical uses. There is another approach using deep neural network for synthesizing sinogram [37]. This method trains a network to learn residual between input sinogram and sparsely sampled sinograms from different angular directions, and concatenates those multiple sparsely-sampled sinograms to generate a full-view sinogram with an optimization network. Our approach particularly differs from it in that our approach trains the network in a patch-based scheme so that one can adapt the network to a system containing different number of detector bins by decomposing into patches. The advancement and wide applications of the deep learning techniques are partly stimulated by the advances in high computational power of general purpose graphic processing unit and by various libraries that are publically available for individuals to apply to various fields [38]–[40].

In this paper, we implemented a convolutional-neural-network (CNN) using the Caffe library [38] for synthesizing the missing data in the sparse-view sinogram. We would like to note that the focus of this paper is on demonstrating that a deep learning-based sinogram synthesis can provide a useful solution to the low-dose CT imaging. We used real patients' CT data from The Cancer Imaging Archive (TCIA) [41] and reprojected the images to generate sinograms for training. Background of CNN, the structure of the networks used in this paper, data preparation for training, and some other methods for a comparison study are described in Section II. The comparison of the results with the other CNN approach, with the analytic interpolation methods, and also with an iterative reconstruction method will be summarized in Section III. Discussion and conclusions will then follow.

II. Methods

A. Convolutional Neural Network

CNN [42] is the most commonly used structure of deep-neural-network for image applications. It is composed of several layers including convolution layer, pooling layer, and fully connected layer, and of activation functions. The convolutional layer performs convolutions to the input data with its output results forming input signals to the next layer. For each layer, weight (W) and bias (b) together with an input (x) are given to the layer and a convolution operation is performed as follows:

$$W * x + b. \tag{1}$$

The pooling layer down-samples input data with a specific method such as maximum pooling, or average pooling. In general, the pooling layer makes shift-invariant results by maintaining specific values from the input. The shift-invariance is important in the applications such as segmentation and classification where the position of the target can be arbitrarily given in the data. Fully connected layer refers to a layer structure in which each neuron is connected to all the neurons in the previous layer and in the next layer. Activation function is applied after fully connected layer or convolutional layer, and it is in the form of a nonlinear function such as hyperbolic tangent, sigmoid, or rectified linear unit (ReLU) [43]. With the data passing through the convolution layer, pooling layer, and activation function, the network finds features for a given task; therefore, handcrafted features are not required anymore. As training goes on, the features would evolve toward the goal with the cost function minimized. The goal in this paper is to synthesize missing sinogram data, and it constitutes a kind of regression problem in which CNN-based approaches have been very successful.

B. Structure of the Proposed Network

We constructed our network based on a residual U-Net. The U-Net is one of the CNN model proposed for image segmentation [44]. Residual learning is one of the techniques that can make a network converge faster and more efficiently. It trains the network to learn differences between the ground truth and the input data [26], [45]. Adding the residual learning scheme to the U-Net showed enhanced performances in removing streak artifacts in medical imaging [35], [36]. We employed the residual learning scheme in the network, and replaced pooling layers by convolutional layers to make the down-sampling trainable as well. Replacing a nontrainable layer by a trainable one has shown outperformances in other applications in [44] and [46]. In our case, the measured values in the sinogram space are more important than the initial guessed values in the missing sinogram. Therefore, giving higher weights to the measured pixels or highly correlated pixels to them is more appropriate than giving higher weights to the maximum values as is often done in a max-pooling

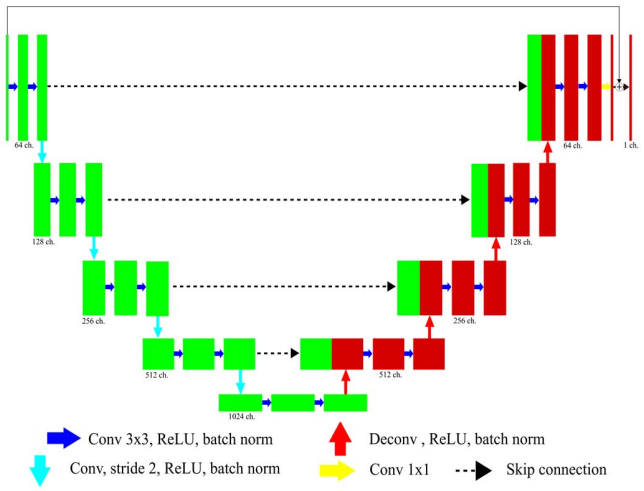


Fig. 1. Network structure for training.

scheme. Using pooling layers is known to yield a faster output than using a stride-based convolution, since pooling does not require convolution operations. However, the restoration accuracy should not be compromised by computation time particularly in medical imaging applications; therefore, additional computation time associated with a stride-based convolution is worth taking.

By replacing pooling layers by convolutional layers, the kernel of the network will be also trained to find optimal down-sampling weights for the task. The structure of our network is shown in Fig. 1. We set the stride of the convolutional layers in association with the down-sampling to be 2, while the other layers to be 1. The input data has a single channel, and the number of output channels of the first convolution layer is 64. The number of channel doubles as the convolution layer with a stride of 2 is applied up to the maximum number of 1024, and it reverses back to a single channel in the U-net structure as shown in Fig. 1. For the convolution layers without down-sampling, we used a zero-padding scheme to maintain the size.

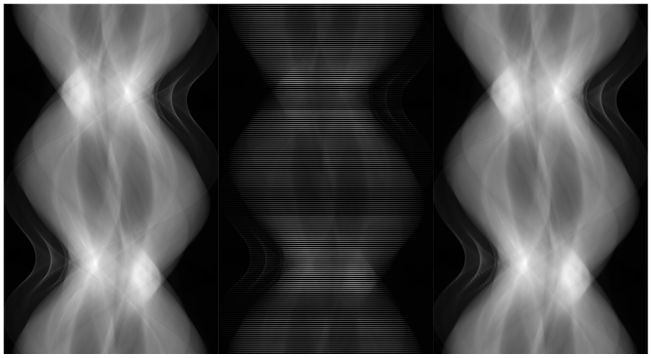
Input data are prepared in patches from the sinograms as will be explained in details later and, at the bottom layer, the original input data and the output data of the last convolution layer are summed in a residual learning framework. To avoid confusion, we would like to use the dedicated terminologies throughout this paper as follows: the input/output/ground truth data are used for representing the patches acquired from the input/output/ground truth onograms, respectively. The cost function shown in the following equation compares the output data from the network with the ground truth data:

$$\frac{1}{2N}\sum_{k=1}^{N}\left\|x^k - y^k\right\|_2^2 \tag{2}$$

where x represents the network output patch, y the ground truth patch in a vectorial format, and N the number of batches used for an iteration. The superscript k refers to a patch in the training set.

TABLE I
SIMULATION CONDITIONS

Conditions	*Values*
Source to detector distance	1500 mm
Source to axis distance	1000 mm
Detector pixel spacing	0.9 mm
Angular interval	0.5 deg
Number of detector channels	750 pixels
Number of projection data	720 views
Field-of-view(FOV) of projection	439.02 mm



(a) (b) (c)

Fig. 2. (a) Ground truth sinogram. (b) Sparsely sampled sinogram from (a) (180 views). (c) Up-sampled sinogram from (b) using linear interpolation.

C. Training the Network

Since CNN allows a supervised machine learning, we need to provide training data and ground truth data to the network. We reprojected seven real patients' images of Lung CT [47] from TCIA using distance-driven projection algorithm [48] in a fan-beam CT imaging geometry in this paper. The number of slice images used for training and validation was 634. The CT scan parameters are summarized in Table I. The field-of-view in Table I means the diameter of a circle covered by the detector with its center located at the isocenter of the system.

We subsampled the original sinograms by a quarter to make them sparsely sampled one. In other words, the sparsely view-sampled sinograms with an equal angular separation between the sampled views has been prepared by selecting every forth views from the original full sinograms. Then, we applied a linear interpolation along the scan angle direction for synthesizing initial full sinograms for training the network. The same size of the input sinogram with that of the original sinogram was thus used in this paper. Example images of an original full sinogram, the sparsely view-sampled sinogram, and the linearly interpolated one are shown in Fig. 2.

We would like to note again that the convolution operations have been applied to patch-based data in the CNN. Patch-based training reduces the memory requirements for input data and increases number of data used for training. From both input and ground truth sinograms, we extracted patches of the same size. We have varied the patch size and found that the patch size around 50 or bigger results in a similar network performance with bigger patch sizes requiring longer computation time due to increasing number of convolution

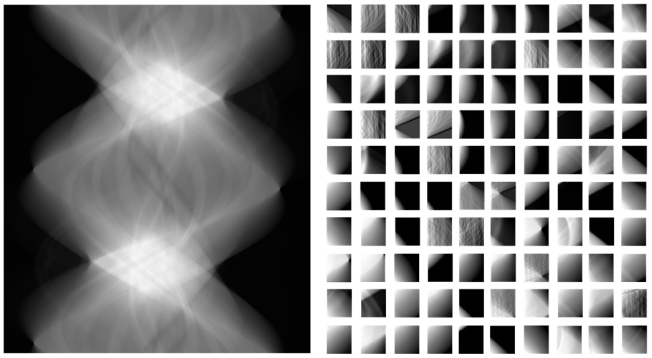


Fig. 3. (a) Up-sampled sinogram for training and (b) patches extracted from (a).

TABLE II
TRAINING PARAMETERS

Parameters	*Values*
Number of data	2,142,660 patches
Batch size	67 patches
Learning rate	0.0001
Solver	Adam optimizer
Number of training epochs	200 epochs

operations. Therefore, we have extracted patches in the size of 50 × 50 with a stride of 10 as shown in Fig. 3. A stride refers to the sampling interval in pixels between the neighboring patches. When the stride is smaller than the size of the patch, each patch has overlapping pixels with its neighboring patches. With these overlapping regions between patches, the tiling artifacts in the synthesized sinogram would be mitigated since the pixel values are averaged in the overlapped regions. The final output data of the network are still in the form of patches, which will be combined in the aforementioned way to form the synthesized sinogram.

Our database contains 2 142 660 patches for training and 918 285 patches for validation. The training parameters of the network are summarized in Table II. We used adaptive momentum estimation (Adam) optimizer [49] to optimize the network. It is one kind of gradient-based optimizers, which has shown outperformance to the stochastic gradient descent methods. The method requires a base-learning rate, first momentum, and second momentum; we set the first and second momentum to be 0.9 and 0.999, respectively, as recommended by the original paper. The computation was done on a PC with Intel i7 2.80 GHz, 16 GB of random access memory, and a GPU of GTX Titan X 12GB memory. Data with a selected batch number pass through the network in an iteration. After all the data prepared for training pass through the network, it completes an epoch. The loss was calculated for every iteration to update the network parameters, and we averaged losses in an epoch to plot them.

D. Other Methods for Comparison

For comparing the performance of the proposed method, we implemented two analytic interpolation methods and another CNN structures. A linear interpolation method and a directional interpolation method were implemented according to [16]. In the linear interpolation method, a linear interpolation was performed along the angular direction to fill the missing data. The directional interpolation algorithm searches for a direction of an imaged object using the gradient of a sinogram. At a given pixel, the interpolation weights are calculated from the eigen value and the vector of the gradient sinogram. A CNN implemented for comparison is composed of 20 successive convolution layers with ReLU activation function, of which the structure is shown in Fig. 4. This network was used in our earlier work by the way [50]. Additionally, we implemented the conventional U-Net with pooling layers for down-sampling as shown in Fig. 5. We also compared our network with the U-Net that is not using a residual learning scheme to see the effects of skip connection.

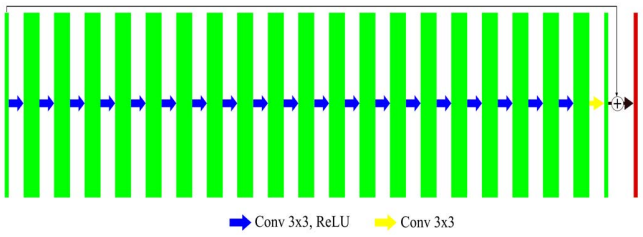


Fig. 4. Twenty successive convolution layers.

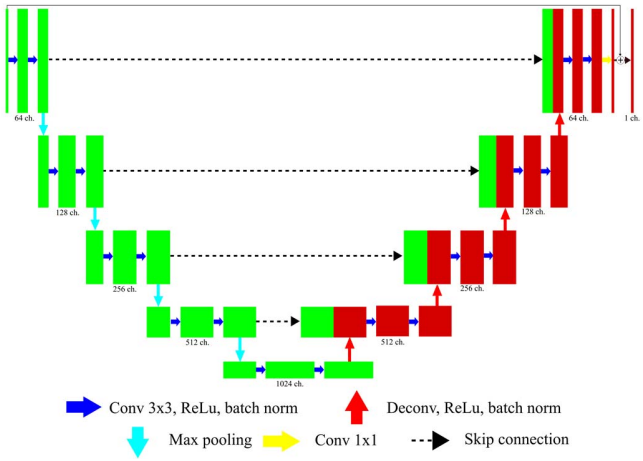


Fig. 5. Network structure for conventional U-Net with pooling layers.

E. Image Reconstruction

FBP algorithm [51] was used for image reconstruction from the ground truth sinogram and also from the synthesized sinogram. The array size of the reconstructed images is 512 × 512. We used the same pixel size of the original patients' CT images, and the pixel size varies among the patients accordingly. For each given imaging task, we would thus have seven FBP-reconstructed images: ground truth image, image from sparsely sampled sinograms, images from the analytically interpolated sinograms (linear and directional interpolation), and images from the sinograms synthesized by three different deep neural networks. In addition, we have implemented an iterative image reconstruction algorithm that can directly

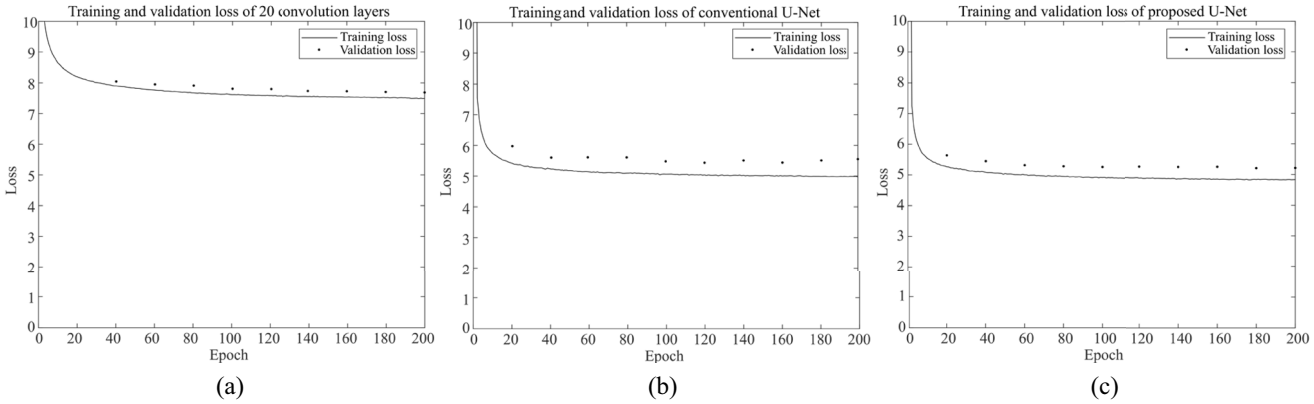


Fig. 6. Training and validation loss plot of (a) 20 successive convolution layers, (b) conventional U-Net, and (c) proposed U-Net.

reconstruct from the sparsely sampled sinogram. We implemented a total variation minimization method with projection on convex sets (POCS-TV) [52].

III. Results

A. Network Training Result

The training loss, or the Euclidean error of the network output of the successive convolution layers, conventional and proposed U-Nets are plotted as a function of epochs in Fig. 6. The solid lines represent the training error, and the scatter points represent the validation error at every 20 epochs. As shown in the plots, the Euclidean loss of validation dataset has similar value to the training error for all networks. Proposed U-Net has resulted in smaller training and validation errors compared to the successive convolutional layers and to the conventional U-Net.

B. Interpolation Results

To evaluate the performance objectively, we recruited eight patients' from the same Lung CT dataset that did not participate neither in training nor validation phases. The number of slices used for evaluation was 662 slices. The sinograms have been prepared in the same way according to the CT scanning geometry and separated into patches with the same size used for training, and fed into the trained networks. Two example sinograms used for evaluation are shown in Figs. 7 and 8 in their absolute differences with the ground truth sinograms. For comparison, sinogram differences of the other methods are also shown. As one can see in the figures, the synthesized sinogram by CNNs have smaller difference from the ground truth sinogram than the sinograms synthesized by other methods. We plotted line profiles of the absolute difference of the sinograms in Fig. 9(a) and (b) along the line segments highlighted by green color in Figs. 7 and 8, respectively.

For a quantitative comparison, we computed normalized root mean-square-error (NRMSE), which is the root mean-square-error divided by the difference between maximum and minimum values of the ground truth images, peak signal-to-noise ratio (PSNR), and structural similarity (SSIM) [53]. Comparison results are summarized in Tables III–V. Although the sinograms synthesized by CNNs have similar values, the proposed network produced more accurate recovery than the other networks.

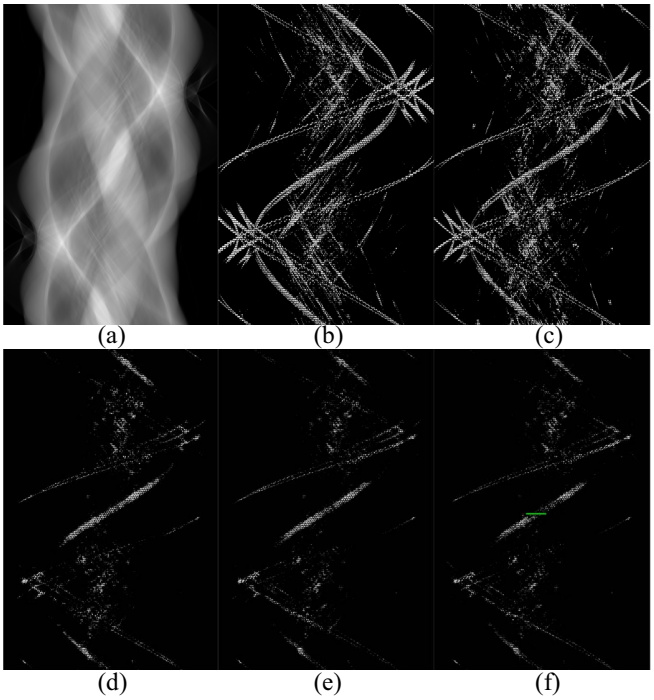


Fig. 7. (a) Ground truth sinogram of patient #1, difference between ground truth sinogram and interpolated sinogram using (b) linear interpolation, (c) directional interpolation, (d) 20 convolution layers, (e) conventional U-Net, and (f) proposed U-Net, display window: [0.25, 0.45].

TABLE III
NRMSE of Interpolated Sinograms

$\times 10^{-3}$	*Linear interpolation*	*Directional interpolation*	*20 Convolution*	*Conventional U-Net*	*Proposed U-Net*
Patient #1	2.14	1.73	0.87	0.81	**0.80**
Patient #2	3.38	2.48	0.97	0.94	**0.93**
Patient #3	1.98	1.69	0.77	0.73	**0.72**
Patient #4	2.34	1.79	0.74	0.69	**0.67**
Patient #5	2.78	2.32	1.07	0.99	**0.97**
Patient #6	1.89	1.61	0.77	0.74	**0.73**
Patient #7	2.56	2.09	0.92	0.87	**0.86**
Patient #8	3.40	2.38	1.19	**1.14**	**1.14**

TABLE IV
PSNR OF INTERPOLATED SINOGRAMS

	Linear interpolation	*Directional interpolation*	*20 Convolution*	*Conventional U-Net*	*Proposed U-Net*
Patient #1	53.38	55.21	61.18	61.83	**61.98**
Patient #2	49.32	52.02	60.17	60.48	**60.50**
Patient #3	53.93	55.35	62.19	62.61	**62.69**
Patient #4	52.60	54.95	62.58	63.20	**63.49**
Patient #5	51.10	52.66	59.40	60.07	**60.28**
Patient #6	54.34	55.73	62.09	62.52	**62.66**
Patient #7	51.82	53.60	60.69	61.15	**61.29**
Patient #8	49.33	52.44	58.46	58.85	**58.86**

TABLE V
AVERAGE SSIM OF INTERPOLATED SINOGRAMS

	Linear interpolation	*Directional interpolation*	*20 Convolution*	*Conventional U-Net*	*Proposed U-Net*
Patient #1	0.9496	0.9541	0.9806	0.9816	**0.9820**
Patient #2	0.9305	0.9409	0.9795	0.9797	**0.9807**
Patient #3	0.9493	0.9552	0.9813	0.9813	**0.9819**
Patient #4	0.9465	0.9561	0.9835	0.9836	**0.9838**
Patient #5	0.9316	0.9393	0.9763	0.9774	**0.9780**
Patient #6	0.9487	0.9557	0.9808	0.9811	**0.9815**
Patient #7	0.9423	0.9491	0.9810	0.9811	**0.9816**
Patient #8	0.8656	0.8853	0.9250	0.9263	**0.9277**

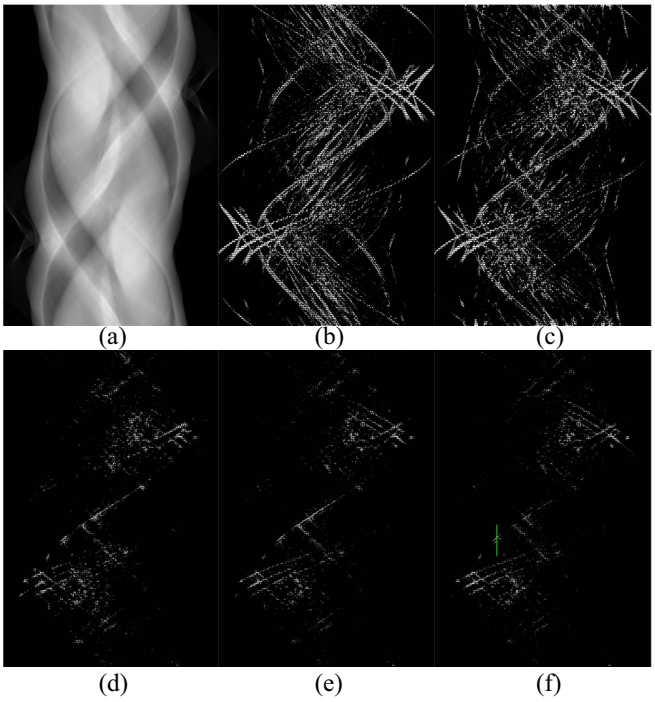


Fig. 8. (a) Ground truth sinogram of patient #5, difference between ground truth sinogram and interpolated sinogram using (b) linear interpolation, (c) directional interpolation, (d) 20 convolution layers, (e) conventional U-Net, and (f) proposed U-Net, display window: [0.25, 0.4].

Reconstructed images from the onogram in Figs. 7 and 8 are shown in Figs. 10 and 11, respectively. For a better visual comparison, we display enlarged images of the red-boxed region-of-interest in Figs. 12 and 13, respectively. Quantitative comparison results, similarly to the onogram comparison, of the reconstructed images are also summarized in Tables VI–VIII.

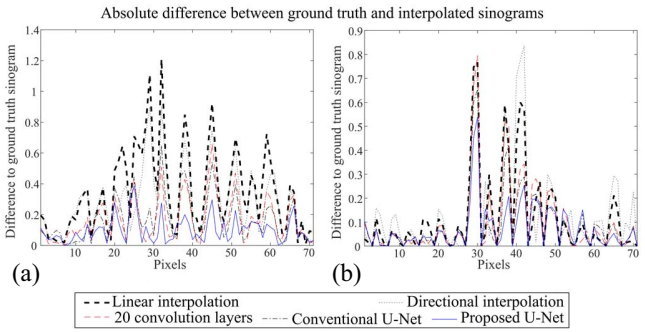


Fig. 9. Line profiles of the green line in difference sinograms of (a) patient #1 and (b) patient #5.

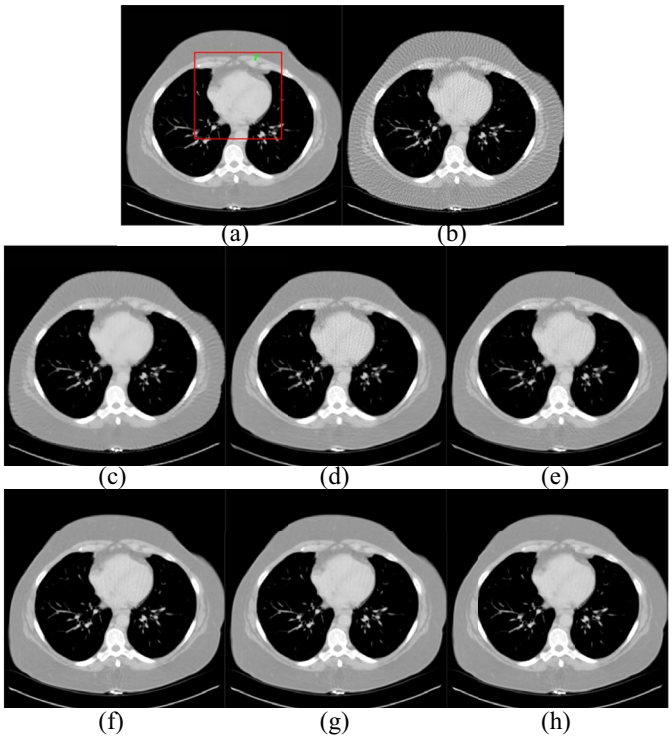


Fig. 10. Reconstructed image of (a) ground truth sinogram, (b) sparse-view sinogram (180 views), (c) POCS-TV (180 views), interpolated sinogram using (d) linear interpolation, (e) directional interpolation, (f) 20 convolution layers, (g) conventional U-Net, and (h) proposed U-Net from patient #1, display window: [−700HU, 300HU].

As shown in Figs. 10 and 11, the images reconstructed by FBP algorithm directly from the sparsely sampled data suffer from severe streak artifacts. The images reconstructed by the TV minimization algorithm are subject to cartoon artifacts and they seem to miss small structures. Moderate streak artifacts remain in the images reconstructed from the synthesized onogram by linear and directional interpolation methods. Images synthesized by CNN have smaller streak artifacts than other methods. Particularly, the reconstructed images from the onogram synthesized by the proposed convolution only U-Net shows the least streak artifacts. Results in Tables VI–VIII also support the visual findings in a quantitative way; NRMSE, PSNR, and SSIM were best in the proposed U-Net case among all the tested methods.

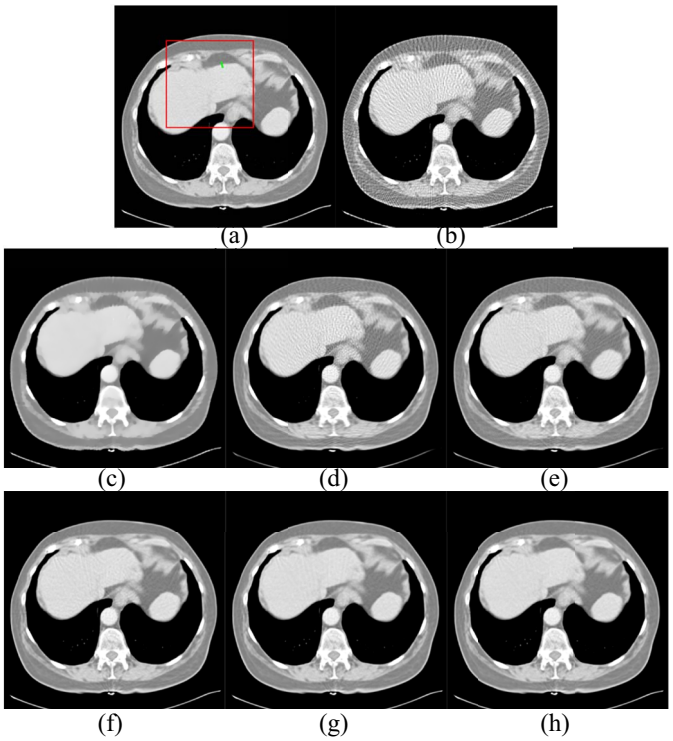


Fig. 11. Reconstructed image of (a) ground truth sinogram, (b) sparse-view sinogram (180 views), (c) POCS-TV (180 views), interpolated sinogram using (d) linear interpolation, (e) directional interpolation, (f) 20 convolution layers, (g) conventional U-Net, and (h) proposed U-Net from patient #5, display window: [−400HU, 170HU].

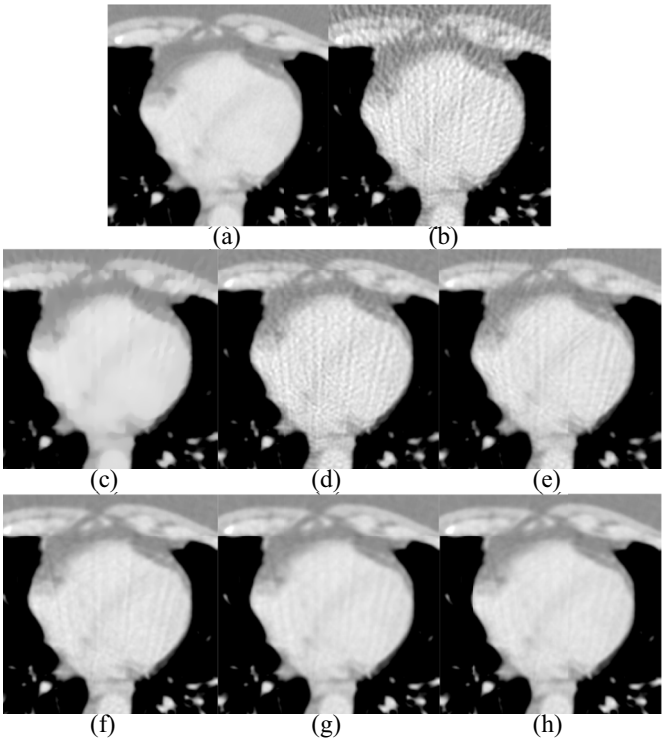


Fig. 12. Enlarged ROI of Fig. 10, display window: [−500HU, 200HU].

C. Sharpness Comparison

To compare sharpness of the reconstructed images acquired from different interpolation schemes, we computed modulation transform function (MTF) in the images. We selected perpendicular line segments from edge structures in the images. Each line profile from the image is regarded as an edge spread function and differentiated it to obtain the line spread function (LSF). MTF was obtained by performing a Fourier transform of LSF. We used green line segments shown in Figs. 10 and 11 to calculate MTF and the results are plotted in Fig. 14(a) and (b), respectively.

The plots in Fig. 14 show that the reconstructed images from the data synthesized by CNNs have sharpness closer to the ground truth than the other methods. The linear and directional interpolations led to rather a smoother edge. Sharpness from the iterative reconstruction algorithm lies between the CNN-based methods and the analytic interpolation methods.

D. Application to Different Sampling Schemes

We applied the proposed U-Net trained at a subsampling factor of 4 to various data with different subsampling factors in order to see whether the matching such subsampling factor is important. We subsampled the sinograms of patient #1 with different sampling factors and synthesized the sinogram using the proposed network. We used subsampling factors of 8, 6, 3, and 2, each of which resulting in 90, 120, 240, and 360 views from the original sinogram, respectively.

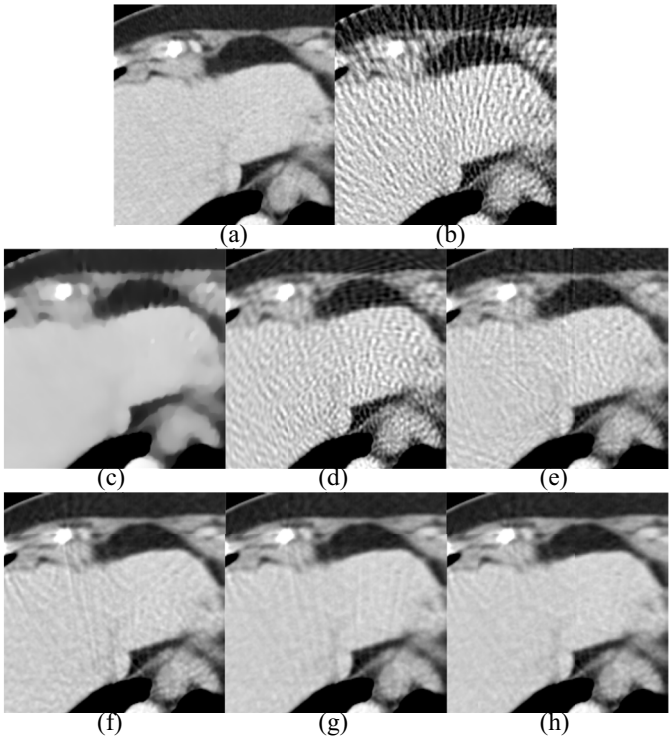


Fig. 13. Enlarged ROI of Fig. 11, display window: [−150HU, 150HU].

Sinogram synthesis results are shown in Fig. 15 and their quantitative comparison results in terms of NRMSE, PSNR, and SSIM are summarized in Table IX. Also, the reconstructed images are shown in Fig. 16 and their quantitative comparison results are summarized in Table X. The proposed network outperforms the linear interpolation at every sampling

TABLE VI
NRMSE OF RECONSTRUCTED IMAGES

($\times 10^{-3}$)	*POCS-TV*	*Linear interpolation*	*Directional interpolation*	*20 Convolution*	*Conventional U-Net*	*Proposed U-Net*
Patient #1	5.45	5.61	4.57	2.39	2.22	**2.19**
Patient #2	8.04	10.43	7.52	2.83	2.77	**2.76**
Patient #3	6.40	7.01	5.88	2.68	2.59	**2.56**
Patient #4	6.09	6.18	4.70	2.04	1.89	**1.83**
Patient #5	7.50	7.88	6.63	3.11	2.89	**2.82**
Patient #6	5.20	5.51	4.64	2.23	2.12	**2.09**
Patient #7	7.02	7.17	5.91	2.77	2.57	**2.51**
Patient #8	12.58	13.43	9.24	4.46	**4.23**	**4.23**

TABLE VII
PSNR OF RECONSTRUCTED IMAGES

	POCS-TV	*Linear interpolation*	*Directional interpolation*	*20 Convolution*	*Conventional U-Net*	*Proposed U-Net*
Patient #1	42.62	42.36	44.14	49.77	50.41	**50.53**
Patient #2	38.62	36.36	39.20	47.68	47.88	**47.89**
Patient #3	40.38	39.60	41.12	47.94	48.23	**48.35**
Patient #4	41.71	41.58	43.95	51.20	51.88	**52.14**
Patient #5	39.89	39.47	40.96	47.54	48.19	**48.39**
Patient #6	43.15	42.65	44.14	50.52	50.94	**51.09**
Patient #7	39.86	39.67	41.36	47.94	48.59	**48.80**
Patient #8	33.26	32.69	35.94	42.26	**42.73**	**42.73**

TABLE VIII
AVERAGE SSIM OF RECONSTRUCTED IMAGES

	POCS-TV	*Linear interpolation*	*Directional interpolation*	*20 Convolution*	*Conventional U-Net*	*Proposed U-Net*
Patient #1	0.9641	0.9698	0.9743	0.9889	0.9903	**0.9905**
Patient #2	0.9490	0.9414	0.9642	0.9908	0.9915	**0.9916**
Patient #3	0.9582	0.9670	0.9721	0.9892	0.9902	**0.9905**
Patient #4	0.9609	0.9640	0.9738	0.9916	0.9928	**0.9932**
Patient #5	0.9558	0.9582	0.9655	0.9872	0.9890	**0.9894**
Patient #6	0.9690	0.9733	0.9779	0.9918	0.9926	**0.9928**
Patient #7	0.9643	0.9694	0.9748	0.9904	0.9916	**0.9919**
Patient #8	0.9364	0.9362	0.9581	0.9815	0.9832	**0.9833**

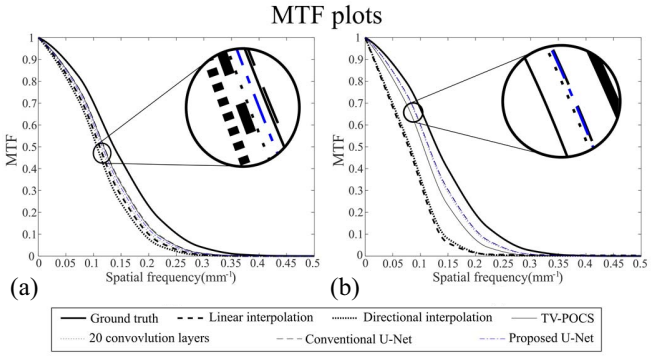


Fig. 14. MTF plots of reconstructed images.

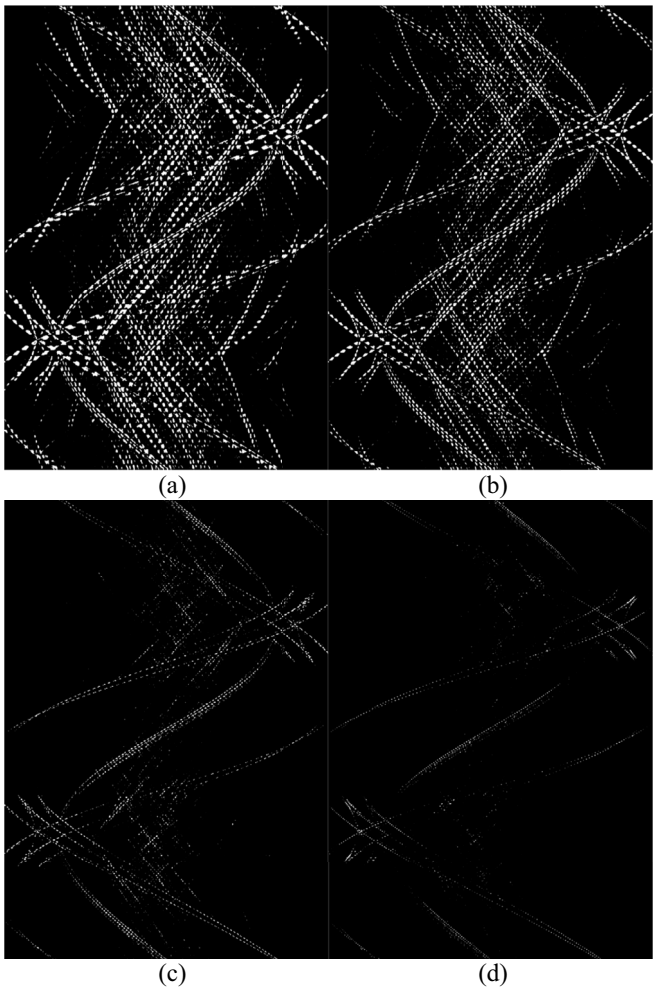


Fig. 15. Difference between ground truth sinogram and interpolated sinogram using the proposed U-Net from (a) 90 views, (b) 120 views, (c) 240 views, and (d) 360 views, display window: [0.25, 0.45].

TABLE IX
QUANTITATIVE COMPARISON RESULTS OF INTERPOLATED SINOGRAMS WITH DIFFERENT SAMPLING SCHEMES

	Linear interpolation			Proposed U-Net		
	NRMSE ($\times 10^{-3}$)	PSNR	SSIM	NRMSE ($\times 10^{-3}$)	PSNR	SSIM
90 views	4.68	46.59	0.8677	4.43	47.06	0.8703
120 views	3.50	49.10	0.9063	2.91	50.69	0.9179
180 views (Sampling rate used for training)	2.14	53.38	0.9496	0.80	61.98	0.9820
240 views	1.48	56.60	0.9706	1.03	59.75	0.9779
360 views	0.84	61.47	0.9881	0.71	62.94	0.9891

rate. However, its outperformance is maximized when the same sampling rate is used for both training and testing, i.e., 180 views.

E. Effect of Residual Learning

To demonstrate the significance of the residual learning scheme, we implemented and trained the same network as the proposed one without residual connection. Training loss of the network is presented in Fig. 17. Compared to Fig. 6, the training loss shows quite a poorer convergence at the same number of iterations than the proposed method.

We synthesized the sinograms using the above trained network and compared the sinograms and reconstruction results as shown in Figs. 18 and 19 for the patient #1 and #5. Fig. 20 shows the enlarged ROI images of Fig. 19.

From a visual comparison, the network trained without the residual learning scheme does not seem to improve sinogram synthesis from the linear interpolation method. Quantitative comparison results are summarized in Tables XI and XII.

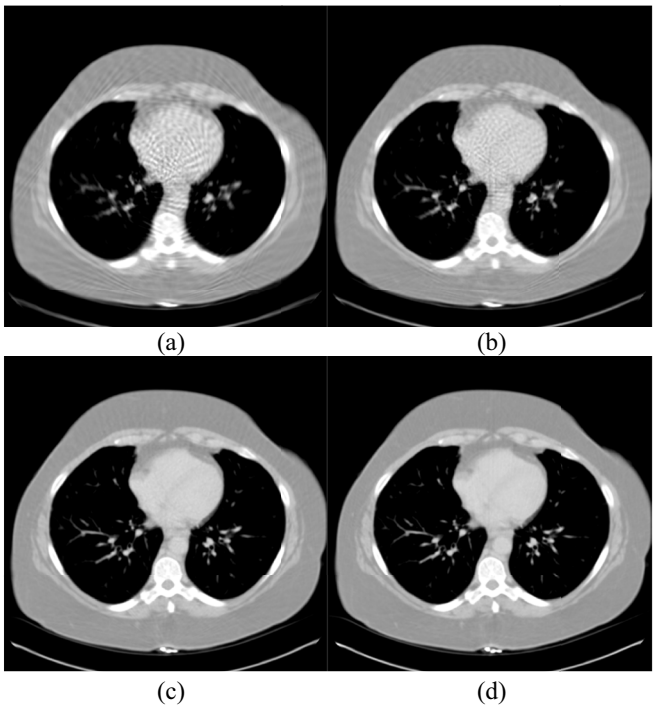


Fig. 16. Reconstructed image of interpolated sinogram from (a) 90 views, (b) 120 views, (c) 240 views, and (d) 360 views, display window: [−700HU, 300HU].

TABLE X
QUANTITATIVE COMPARISON RESULTS OF RECONSTRUCTED IMAGES OF INTERPOLATED SINOGRAMS WITH DIFFERENT SAMPLING SCHEMES

	Linear interpolation			Proposed U-Net		
	NRMSE ($\times 10^{-3}$)	PSNR	SSIM	NRMSE ($\times 10^{-3}$)	PSNR	SSIM
90 views	12.18	35.62	0.8994	11.37	36.23	0.9060
120 views	9.20	38.06	0.9347	7.56	39.78	0.9489
180 views (Sampling rate used for training)	5.61	42.36	0.9698	2.19	50.53	0.9905
240 views	3.74	45.88	0.9847	2.56	49.18	0.9898
360 views	1.90	51.78	0.9949	1.73	52.61	0.9950

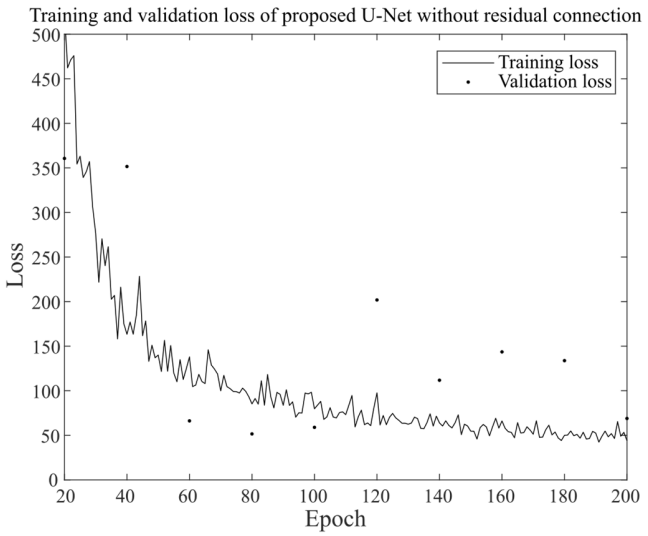


Fig. 17. Training and validation loss of proposed U-Net without residual connection.

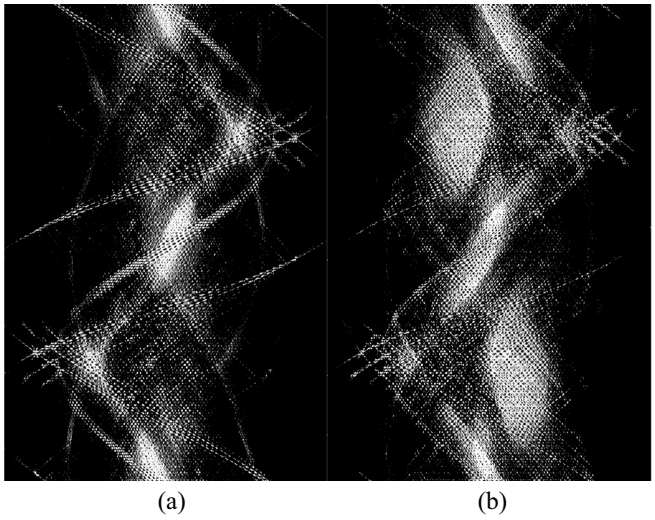


Fig. 18. Difference of interpolated sinogram using proposed U-Net without residual connection and ground truth sinogram of (a) patient #1, display window: [0.25, 0.45], and (b) patient #5, display window: [0.25, 0.4].

IV. DISCUSSION

This paper reveals that the CNN-based interpolation or synthesis of the sparsely sampled sinogram can effectively make up the missing data and can produce reconstructed images of comparable quality to the ones reconstructed from the fully sampled sinogram. It is thought that an over-smoothing during the POCS-TV reconstruction resulted in higher discrepancies compared to the ground-truth FBP reconstructed image, which contains its own noise characteristics, in this paper. Although the POCS-TV reconstruction results from the sparsely sampled data are rather poor in this paper, we would like to note that such an iterative algorithm strongly depends on the optimization cost function and reconstruction parameters. Therefore, we cannot exclude a chance that a fine-tuned iterative algorithm can produce a reasonably acceptable image quality in a given imaging task. However, it is a common understanding that such compressed-sensing-inspired algorithms are in general subject to cartoon image artifacts and that they may miss small structures in the reconstructed images from the sparsely sampled data. In contrast, the proposed method is free of such parameter tuning, which indeed highlights its strength.

We would like to note that we have used different display window for each patient case to better highlight the performance differences in the investigated methods. Therefore, the visual perception of image quality between patient #1 case and #5 may be different as shown in Fig. 19. The proper comparison, however, should be made between Figs. 10–13 and Fig. 19 accordingly. The performance of the proposed method is indeed consistently higher than the others in all the patient cases as summarized in Tables VII and VIII, which shows its robustness against patient diversity.

The training took about ten days for the successive convolutional layers, and about 18 and 24 days for the conventional U-Net, and the proposed U-Net in our computing

Fig. 19. Reconstructed image of interpolated sinogram using proposed U-Net without residual connection from (a) patient #1, display window: [−700HU, 300HU], and (b) patient #5, display window: [−400HU, 170HU].

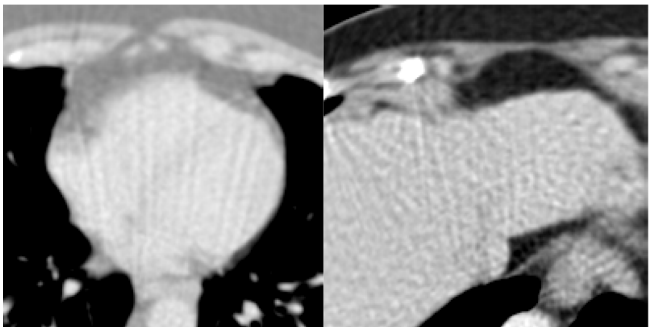


Fig. 20. Enlarged ROI of Fig. 19. (a) Display window: [−500HU, 200HU]. (b) Display window: [−150HU, 150HU].

environment, respectively. However, the sinogram synthesis after the networks complete the training took about 10 s, 40 s, and 50 s for the successive convolutional layers, conventional U-Net, and the proposed U-Net, respectively. One may argue that using more layers in the successive convolutional network so as to require similar training time to that of the U-Nets can possibly reach a higher performance. While we do not exclude such a chance that the successive convolutional network can achieve a similar performance to the U-Nets, we want to emphasize that the number of layers would be much larger and the training time accordingly would be way longer than the ones in the U-Nets. Considering the number of training parameters in the network that would contribute to the computation complexity, the successive convolutional network would need substantially larger number of layers to have a similar computational complexity since the U-Nets use varying number of channels and convolution schemes at different layers.

Since we used relatively a small stride while making the training dataset, the data redundancy in the dataset is relatively high. While it helps increasing the number of training data, it also increases the training time. We will further investigate on reducing redundancies of the dataset as an attempt to increase the training speed in the future without compromised performance of the network. The network showed the best performance when the test sinograms had the same subsampling factor with the trained sinograms. It would be interesting to investigate on the image quality as the subsampling factor varies assuming that the network has been trained by each data set at a given subsampling factor.

TABLE XI
QUANTITATIVE COMPARISON RESULTS OF INTERPOLATED SINOGRAMS USING PROPOSED U-NET WITHOUT RESIDUAL CONNECTION

	NRMSE ($\times 10^{-3}$)	PSNR	SSIM
Patient #1	3.02	50.38	0.8528
Patient #2	2.32	52.59	0.8902
Patient #3	2.87	50.72	0.8362
Patient #4	2.14	53.37	0.7848
Patient #5	2.70	51.36	0.7763
Patient #6	2.41	52.23	0.7849
Patient #7	2.87	50.83	0.8168
Patient #8	2.89	50.74	0.8701

TABLE XII
QUANTITATIVE COMPARISON RESULTS OF RECONSTRUCTED IMAGES OF INTERPOLATED SINOGRAMS USING PROPOSED U-NET WITHOUT RESIDUAL CONNECTION

	NRMSE ($\times 10^{-3}$)	PSNR	SSIM
Patient #1	3.15	47.37	0.9800
Patient #2	3.74	45.26	0.9839
Patient #3	3.66	45.24	0.9767
Patient #4	2.80	48.46	0.9818
Patient #5	3.92	45.52	0.9792
Patient #6	2.86	48.34	0.9857
Patient #7	3.66	45.51	0.9821
Patient #8	5.52	40.41	0.9686

Based on our preliminary study, we will continue investigating the utility of the deep network in clinical environments that include fan-beam CT, cone-beam CT, and helical multiple fan-beam CT. Additionally, an irregular angular sampling in the sparse-view data acquisition as well as handling missing detector channel problem would be our future study.

V. CONCLUSION

In this paper, we implemented a U-Net structure for interpolating sparsely sampled singoram to reconstruct CT images by an FBP algorithm. We trained the network with the reprojected data from the real patients' CT images. We compared the performance of the proposed method to the linearly interpolated sinogram, the directionally interpolated sinogram, and the interpolated sinograms using the other CNNs. Reconstructed images have also been compared likewise, and the reconstructed image by the proposed method was also compared to the image reconstructed by a TV-minimization algorithm directly from the sparsely sampled data. The proposed network produced promising results and is believed to play an important role as an option to the low-dose CT imaging.

REFERENCES

[1] E. J. Hall and D. J. Brenner, "Cancer risks from diagnostic radiology," *Brit. J. Radiol.*, vol. 81, no. 965, pp. 362–378, 2008.
[2] D. J. Brenner and E. J. Hall, "Computed tomography—An increasing source of radiation exposure," *New England J. Med.*, vol. 357, no. 22, pp. 2277–2284, 2007.
[3] L. Yu *et al.*, "Radiation dose reduction in computed tomography: Techniques and future perspective," *Imag. Med.*, vol. 1, no. 1, pp. 65–84, 2009.

[4] C. H. McCollough *et al.*, "Strategies for reducing radiation dose in CT," *Radiol. Clin. North America*, vol. 47, no. 1, pp. 27–40, 2009.

[5] J. Huang *et al.*, "Iterative image reconstruction for sparse-view CT using normal-dose image induced total variation prior," *PLoS ONE*, vol. 8, no. 11, 2013, Art. no. e79709.

[6] T. Lee, C. Lee, J. Baek, and S. Cho, "Moving beam-blocker-based low-dose cone-beam CT," *IEEE Trans. Nucl. Sci.*, vol. 63, no. 5, pp. 2540–2549, Oct. 2016.

[7] S. Abbas, J. Min, and S. Cho, "Super-sparsely view-sampled cone-beam CT by incorporating prior data," *J. X-Ray Sci. Technol.*, vol. 21, no. 1, pp. 71–83, 2013.

[8] S. Singh *et al.*, "Adaptive statistical iterative reconstruction technique for radiation dose reduction in chest CT: A pilot study," *Radiology*, vol. 259, no. 2, pp. 565–573, 2011.

[9] A. H. Andersen and A. C. Kak, "Simultaneous algebraic reconstruction technique (SART): A superior implementation of the art algorithm," *Ultrason. Imag.*, vol. 6, no. 1, pp. 81–94, 1984.

[10] E. Y. Sidky and X. Pan, "Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization," *Phys. Med. Biol.*, vol. 53, no. 17, pp. 4777–4807, 2008.

[11] R. A. Brooks, G. H. Weiss, and A. J. Talbert, "A new approach to interpolation in computed tomography," *J. Comput. Assist. Tomography*, vol. 2, no. 5, pp. 577–585, 1978.

[12] Z. Chen, B. J. Parker, D. D. Feng, and R. Fulton, "Temporal processing of dynamic positron emission tomography via principal component analysis in the sinogram domain," *IEEE Trans. Nucl. Sci.*, vol. 51, no. 5, pp. 2612–2619, Oct. 2004.

[13] H. Kostler, M. Prummer, U. Rude, and J. Hornegger, "Adaptive variational sinogram interpolation of sparsely sampled CT data," presented at the 18th Int. Conf. Pattern Recognit., Jan. 2006. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp = &arnumber=1699641 &contentType=Conference+Publications

[14] M. Pohlmann, M. Berger, A. Maier, J. Hornegger, and R. Fahrig, "Estimation of missing fan-beam projections using frequency consistency conditions," in *Proc. Int. Conf. Image Formation X-Ray Comput. Tomography*, 2014, pp. 203–207.

[15] M. Bertram, J. Wiegert, D. Schafer, T. Aach, and G. Rose, "Directional view interpolation for compensation of sparse angular sampling in cone-beam CT," *IEEE Trans. Med. Imag.*, vol. 28, no. 7, pp. 1011–1022, Jul. 2009.

[16] H. Zhang and J.-J. Sonke, "Directional sinogram interpolation for sparse angular acquisition in cone-beam computed tomography," *J. X-Ray Sci. Technol.*, vol. 21, no. 4, pp. 481–496, 2013.

[17] S. Li *et al.*, "Dictionary learning based sinogram inpainting for CT sparse reconstruction," *Optik Int. J. Light Electron Opt.*, vol. 125, no. 12, pp. 2862–2867, 2014.

[18] D. Karimi and R. Ward, "Reducing streak artifacts in computed tomography via sparse representation in coupled dictionaries," *Med. Phys.*, vol. 43, no. 3, pp. 1473–1486, 2016.

[19] S. Siltanen and M. Kalke, "Sinogram interpolation method for sparse-angle tomography," *Appl. Math.*, vol. 5, no. 3, pp. 423–441, Jan. 2014.

[20] C. Toumoulin *et al.*, "Strategy of computed tomography sinogram inpainting based on sinusoid-like curve decomposition and eigenvector-guided interpolation," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 29, no. 1, p. 153, Jan. 2012.

[21] G. H. Weiss, A. J. Talbert, and R. A. Brooks, "The use of phantom views to reduce CT streaks due to insufficient angular sampling," *Phys. Med. Biol.*, vol. 27, no. 9, pp. 1151–1162, Sep. 1982.

[22] E. G. Amaro, M. A. Nuno-Maganda, and M. Morales-Sandoval, "Evaluation of machine learning techniques for face detection and recognition," presented at the 22nd Int. Conf. Elect. Commun. Comput., 2012. [Online]. Available: http://ieeexplore.ieee.org/document/6189911/

[23] K. Usman and K. Rajpoot, "Brain tumor classification from multi-modality MRI using wavelets and machine learning," *Pattern Anal. Appl.*, vol. 20, no. 3, pp. 871–881, 2017.

[24] N. Subash and J. Rajeesh, "Brain tumor classification using machine learning," *Int. J. Comput. Technol. Appl.*, vol. 8, no. 5, pp. 2335–2341, 2015.

[25] X. Ren and J. Malik, "Learning a classification model for segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1, 2003, pp. 10–17.

[26] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1646–1654.

[27] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1637–1645.

[28] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," presented at the Eur. Conf. Comput. Vis., 2014.

[29] S. Antol *et al.*, "VQA: Visual question answering," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2425–2433.

[30] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2414–2423.

[31] I. Goodfellow *et al.*, "Generative adversarial nets," presented at the Adv. Neural Inf. Process. Syst., 2014.

[32] H. Chen *et al.*, "Low-dose CT via convolutional neural network," *Biomed. Opt. Express*, vol. 8, no. 2, pp. 679–694, 2017.

[33] E. Kang, W. Chang, J. Yoo, and J. C. Ye, "Deep convolutional framelet denoising for low-dose CT via wavelet residual network," *arXiv preprint arXiv:1707.09938*, 2017.

[34] E. Kang, J. Min, and J. C. Ye, "A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction," *Med. Phys.*, vol. 44, no. 10, pp. e360–e375, 2017.

[35] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017.

[36] Y. Han, J. Yoo, and J. C. Ye, "Deep residual learning for compressed sensing CT reconstruction via persistent homology analysis," *arXiv preprint arXiv:1611.06391*, 2016.

[37] K. Liang, H. Yang, K. Kang, and Y. Xing, "Improve angular resolution for sparse-view CT with residual convolutional neural network," in *Proc. SPIE Med. Imag.*, vol. 10573, 2018, p. 11.

[38] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, Orlando, FL, USA, 2014, pp. 675–678.

[39] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[40] A. Vedaldi and K. Lenc, "MatConvNet: Convolutional neural networks for MATLAB," in *Proc. ACM Int. Conf. Multimedia*, 2015, pp. 689–692.

[41] K. Clark *et al.*, "The cancer imaging archive (TCIA): Maintaining and operating a public information repository," *J. Digit. Imag.*, vol. 26, no. 6, pp. 1045–1057, 2013.

[42] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*. Cambridge, MA, USA: MIT Press, 1995, pp. 255–258.

[43] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," presented at the 14th Int. Conf. Artif. Intell. Stat. Proc. Mach. Learn. Res., 2011. [Online]. Available: http://proceedings.mlr.press

[44] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," presented at the Int. Conf. Med. Image Comput. Comput. Assist. Intervention, 2015.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[46] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.

[47] O. Grove *et al.*, "Quantitative computed tomographic descriptors associate tumor shape complexity and intratumor heterogeneity with prognosis in lung adenocarcinoma," *PLoS ONE*, vol. 10, no. 3, 2015, Art. no. e0118261.

[48] B. De Man and S. Basu, "Distance-driven projection and backprojection in three dimensions," *Phys. Med. Biol.*, vol. 49, no. 11, pp. 2463–2475, 2004.

[49] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[50] H. Lee, J. Lee, and S. Cho, "View-interpolation of sparsely sampled sinogram using convolutional neural network," in *Proc. SPIE Med. Imag.*, Orlando, FL, USA, 2017, pp. 1–8.

[51] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. Philadelphia, PA, USA: SIAM, 2001.

[52] E. Y. Sidky, C.-M. Kao, and X. Pan, "Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT," *J. X-Ray Sci. Technol.*, vol. 14, no. 2, pp. 119–139, 2006.

[53] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.