A Sparse-View CT Reconstruction Method Based on Combination of DenseNet and Deconvolution

Zhicheng Zhang, Xiaokun Liang, Xu Dong, Yaoqin Xie, and Guohua Cao

***Abstract*— Sparse-view computed tomography (CT) holds great promise for speeding up data acquisition and reducing radiation dose in CT scans. Recent advances in reconstruction algorithms for sparse-view CT, such as iterative reconstruction algorithms, obtained high-quality image while requiring advanced computing power. Lately, deep learning (DL) has been widely used in various applications and has obtained many remarkable outcomes. In this paper, we propose a new method for sparse-view CT reconstruction based on the DL approach. The method can be divided into two steps. First, filter backprojection (FBP) was used to reconstruct the CT image from sparsely sampled sinogram. Then, the FBP results were fed to a DL neural network, which is a DenseNet and deconvolution-based network (DD-Net). The DD-Net combines the advantages of DenseNet and deconvolution and applies shortcut connections to concatenate DenseNet and deconvolution to accelerate the training speed of the network; all of those operations can greatly increase the depth of network while enhancing the expression ability of the network. After the training, the proposed DD-Net achieved a competitive performance relative to the state-of-the-art methods in terms of streaking artifacts removal and structure preservation. Compared with the other state-of-the-art reconstruction methods, the DD-Net method can increase the structure similarity by up to 18% and reduce the root mean square error by up to 42%. These results indicate that DD-Net has great potential for sparse-view CT image reconstruction.**

***Index Terms*— Sparse-view CT, CT reconstruction, deep learning, DenseNet, deconvolution.**

Manuscript received February 7, 2018; revised March 25, 2018 and March 29, 2018; accepted March 30, 2018. Date of publication April 5, 2018; date of current version May 31, 2018. This work was supported in part by the Dr. Guohua Cao's CAREER Award from the U.S. National Science Foundation under Grant CBET 1351936, in part by the National Key Research and Develop Program of China under Grant 2016YFC0105102, in part by the Union of Production, Study and Research Project of Guangdong Province under Grant 2015B090901039, in part by the Technological Breakthrough Project of Shenzhen City under Grant JSGG20160229203812944, in part by the Natural Science Foundation of Guangdong Province under Grant 2014A030312006, and in part by the UCAS Joint PhD Training Program. *(Corresponding authors: Yaoqin Xie; Guohua Cao.)*

Z. Zhang is with the Institute of Biomedical and Health Engineering, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, also with the Shenzhen Colleges of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 518055, China, and also with the Department of Biomedical Engineering and Mechanics, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 USA (e-mail: zc.zhang@siat.ac.cn).

X. Liang and Y. Xie are with the Institute of Biomedical and Health Engineering, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with the Shenzhen Colleges of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: xk.liang@siat.ac.cn; yq.xie@siat.ac.cn).

X. Dong and G. Cao are with the Department of Biomedical Engineering and Mechanics, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 USA (e-mail: xu14@vt.edu; ghcao@vt.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author.

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/TMI.2018.2823338

I. Introduction

SINCE its invention [1], [2], X-ray Computed Tomography (CT) as a tool to achieve "inner vision" [3] has been widely utilized in clinical, industrial and other applications [4]. With the broadened application of CT in clinics, however, the associated x-ray radiation dose which may potentially induce lifetime risk of cancers has attracted wide public attention [5]. As a result, the demand for radiation dose reduction techniques becomes more necessary [6] under the principle of ALARA (as low as reasonably achievable) [7]. Since x-ray imaging is mainly a photon-noise dominated process [8], reducing the x-ray exposure will lead to image degradation under other identical conditions. Balancing image quality and x-ray dose level has become a well-known trade-off problem in CT.

There are two common strategies to lower x-ray radiation dose. The first strategy is to lower the x-ray exposure in each view [8]–[10], by adjusting the tube current or exposure time of an x-ray source. This practice has been well performed but will produce noisy projections. The other strategy is to decrease the number of projections for a given scanning trajectory [11]–[13]. However, it will lead to the problem of insufficient projection data, to the extent that sparse-view CT will suffer from severe streaking artifacts [14]. On the other hand, sparse view CT from limited-angle CT data acquisition causes image distortion and other problems [15]. In this work, we focus on the strategy for obtaining a high-quality CT image from sparse-view CT with sufficient scanning angle.

Many algorithms have been proposed for sparse-view CT image reconstruction. They can be grouped into three categories. The first category can be classified as sinogram completion. Its key idea is to complement the sparse projections before reconstructing them with analytical algorithms, such as FBP. Li *et al.* [16] proposed a dictionary-based sinogram

completion method to in-paint the missing sinogram data by applying K-SVD algorithm [17], with database composed of the patches from simulated CT sinogram. Lee *et al.* [18] applied a convolution neural network (CNN) [19] to interpolate the missing data in sinogram from sparse-view CT, by utilizing residual learning for better convergence and using patch-wise training of the network to avoid memory overload. The second category of reconstruction algorithms for sparse-view CT is iterative reconstruction (IR). This category of reconstruction methods is in explosive growth due to the recent availability of high computing power in personal workstations [20]. With the emergence of compressed sensing (CS) in 2006 [21], [22], not only the statistical properties of data in projection domain but also lots of prior information in the image domain, such as sparsity in gradient domain [11] and low rank [23], have been taken into account into the IR methods. They can greatly improve the reconstructed image quality. Kim *et al.* [24] employed reweighted non-local total-variation (R-NLTV) to replace total variation (TV) [11] to preserve image details. Niu *et al.* [25] took full advantage of statistical properties of data in the projection domain and total generalized-variation regularization to avoid the blocky effect. Nien and Fessler [26] and Kim *et al.* [27] combined statistical properties of data in the projection domain with edge-preserving regularization and proposed corresponding acceleration technique. The last category of reconstruction algorithms for sparse-view CT is image post-processing. After reconstruction with analytical algorithm (such as FBP) from sparse-view CT projection data, streaking artifacts will be present in the CT images. The removal of streaking artifacts can be readily converted into an image post-processing problem. In this category of reconstruction methods, the iterative reconstruction in image space (IRIS) by Siemens [28] can achieve excellent performance. Han *et al.* [29] took the information from other energy bins in spectral CT into account and employed annihilating filter-based low-rank Hankel matrix approach (ALOHA) to restore CT images reconstructed from sparse-view CT.

Very recently, several deep-learning (DL) based algorithms [30] have been applied to CT reconstruction. For instance, Würf *et al.* [31] and Ma *et al.* [32] employed neural networks to express CT image reconstruction. Cheng *et al.* [33] simulated the iterative reconstruction process using a DL-based leapfrogging strategy. Analysis *et al.* [34] developed a deep residual learning network based approach for sparse-view CT reconstruction via the persistent homology analysis; it showed that the residual image composed of only streaking artifacts is much simpler than the original CT images. Jin *et al.* [35] proposed a deep convolutional network (FBPConvNet) that combines FBP and U-net, and made full use of multi-resolution CNN [36] and residual learning [37]. The image reconstruction for sparse-view CT has also been investigated with RED-CNN, which is based on the combination of deconvolution network and shortcut connections [4]. A forward-thinking review has been published on the subject of applying deep learning for CT [38].

In this work, a new DL based reconstruction method for sparse-view CT is proposed. It can outperform the other methods in terms of computational efficiency and image quality. The new method includes two steps. First, FBP was utilized to reconstruct CT images from sparsely sampled projection data from sparse view CT. As expected, the CT images were full of streaking artifacts. Next, the FBP images were enhanced by an end-to-end neural network that takes full advantage of DenseNet [39] and deconvolution [40] (hereafter the network is referred as DD-Net). The DD-Net reconstruction method takes advantage of the classical FBP reconstruction technique, which is an analytical process and can be calculated efficiently, and the DL based image optimization approach, which is known to be capable of efficiently learning low-level and high-level common features from dataset through a multi-layer network [30].

The detailed reasoning for choosing this particular DL neural network, as well as its advantages in image enhancement, can be found in the Result and Discussion sections. Here, we provide a brief summary on the advantages of the DD-Net. For any DL-based image enhancement tasks, there are two important procedures: feature extraction and image restoration. For feature extraction, to obtain more feature-maps from low-level image edges to high-level object features, the size of neural network (NN) needs to increase. However, increasing the network size will lead to many other problems, including network training difficulty due to gradient vanish or explosion, and increased number of model parameters, etc. In DD-Net, these problems can be overcome by feature reuse, which can make the network more condensed, easily trained, and parameter-efficient. For image restoration, the deconvolution layers employed in the DD-Net network are effective, because they can use every point in the front layer to restore details in an area in later layers. Furthermore, the DD-Net uses shortcut connections to concatenate the DenseNet and deconvolution layers, which can greatly improve the training of the network.

Section II introduces the proposed reconstruction framework, and experimental results are described in section III. Discussion and conclusion are in section IV.

II. Methods

A. Method Overview

The universal approximation theorem [41] states that a feedforward neural network with at least one hidden layer can approximate continuous functions. Although the exact theoretical foundation is not well understood, neural networks seem to be capable of representing a wide variety of interesting functions as long as parameters in the network were set appropriately. It is well known that sparse-view CT reconstruction with FBP will lead to lots of steaking artifacts in CT image X due to missing data in some scanning angles. To transform X into a high-quality CT image Y, the task can be converted to search for a function f according to the equation: $Y = f(X)$. This equation can be represented by a neural network according to the universal approximation theorem. In a high level, our method can be interpreted as the following figure.

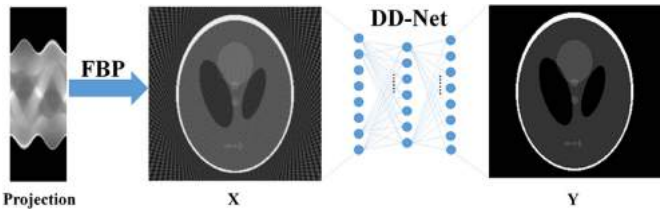

Fig. 1. Framework of the proposed sparse-view CT reconstruction method. Projection data from sparse-view CT are first reconstructed using FBP, and then the FBP reconstructed image is improved using a neural network (DD-Net).

1) DD-Net: The neural network used for image enhancement is constructed mainly based on the combination of DenseNet and deconvolution. The corresponding neural network architecture of the DD-Net is illustrated in Figure 2. It consists of 45 layers, including 4 dense blocks (i.e., the cells of DenseNet), 8 deconvolution layers, 5 convolution layers, and other operators such as batch normalization (BN) [42] and rectified linear unit (ReLU) [43], as well as pooling operators for more advanced feature-mapping (All parameters are determined experimentally as seen in section III.B). Deconvolution network consists of the remaining deconvolution layers and unpooling operators to restore a high-quality image from the extracted feature-maps. Shortcut connections are applied to concatenate the results from the front part of the DenseNet to the later parts of the deconvolution network. The ends of a concatenation must have the same image size. In Figure 2, the red dot-dashed box A consists of a pooling operator, a dense block, and a convolution layer; the green dot-dashed box B encloses an unpooling operator and two deconvolution layers. The proposed DD-Net works in the image domain, with the FBP reconstructed CT image as the input.

2) Dense Block: Dense blocks are the cells of DenseNet. One major difference between DenseNet and the previous neural networks (e.g., ResNets [37], Highway network [44] and FractalNets [45]) lies in the dense connections between different layers within a dense block. Within a dense block, feature-maps produced from all its previous layers are concatenated as the input for the subsequent layer. Furthermore, the inputs of all layers are modified by the dense connections. As a result, each layer in the dense block can benefit from both the low-level features and the high-level features produced before it in the feedforward setting, mitigating the risk of exploding or vanishing gradients [43], [46]. This design also allows the gradients to be sent to their respective places in the network more quickly in the feedbackward situation. The layout of the dense block is illustrated in Figure 3.

Another main difference between DenseNet and the previous neural networks (e.g., ResNets, Highway network and FractalNets) is feature reuse. Feature reuse means that the feature-maps learned from all previous layers are concatenated with the feature-map learned from the current layer as the input for the subsequent layer. Compared to previous neural networks such as ResNets, Highway network, and FractalNets, feature reuse greatly reduces the number of parameters and improves parameter utilization. In a dense block, any two adjacent layers are directly connected by multiple operations such as batch normalization (BN) [42], rectified linear unit (ReLU) [43], and convolution layers. A 1×1 convolution layer is also introduced to decrease the number of input feature-maps due to too many input features after concatenations, resulting in improved computational efficiency [47]. Motivated by [46], every L layer consists of BN-ReLU-Conv (1×1)-BN-ReLU-Conv (5×5). Here, batch normalization can normalize corresponding feature-maps, x, to a particular distribution of y, using the formula $y = \gamma\ (x - \mu)\ /\delta + \beta$, where μ is the mean value of x, δ is the variance of x, γ is the scale factor, and β is the offset value. More details about the batch normalization can be found in [42]. In our work, we followed the formula in [42] and processed each channel of the integrated features after concatenation. Conv $(n \times n)$ stands for two-dimensional convolution operation using convolution filter with size $n \times n$. If each layer L_i has k feature-maps as output, there are $k \times (i - 1) + k_0$ input feature-maps, where k_0 is the number of channels of input. In this paper, all the parameter strides of convolution and deconvolution are set to 1.

3) Deconvolution Network: For DenseNet, pooling operators and dense blocks are utilized interchangeably to extract the major features in [39], effectively ignoring image details. This approach is related to traditional object classification. The key idea of object classification is to extract common features for the same kind of objects and then the output is the identification probability [39] or location [48] or boundary [36] of objects. However, the task of image optimization in the second step of our proposed reconstruction method for sparse-view CT is an end-to-end regression problem. The output is a high-quality CT image.

Deconvolution can be viewed as the reverse version of convolution. In this work, the deconvolution network consists of unpooling operators and deconvolution layers. The deconvolution network reconstructs images from the extracted features obtained from the DenseNet. It has two purposes: keeping the size of output image consistent with the size of input image and restoring a high-quality image from extracted features.

After feature extraction, many features of different sizes are obtained. To restore a high-quality image with the same size as the input image, the first step is to adopt an unpooling operator to enlarge features size using interpolation. Then, feature-maps with the same size produced in the front layers are introduced to the deconvolution network via shortcut connections.

To restore image details, there are two deconvolution layers: Deconv (5×5)-ReLU-BN and Deconv (1×1)-ReLU-BN. Here, Deconv $(n \times n)$ stands for two-dimensional deconvolution operation using a filter with size $n \times n$. Each deconvolution layer allows each point in the front layer to be 'deconvoluted' to an area in the back layer whose size equals the size of the deconvolution filter. This process is capable of bringing the image details back to the output images, and hence helps to generate high-quality CT image from extracted features produced from the DenseNet. Previously, deconvolution, in combination with convolution, has been applied successfully in sematic segmentation problems [47], [49], [50].

4) Shortcut Connections: Shortcut connections play a significant role in the DD-Net. There are two kinds of shortcut

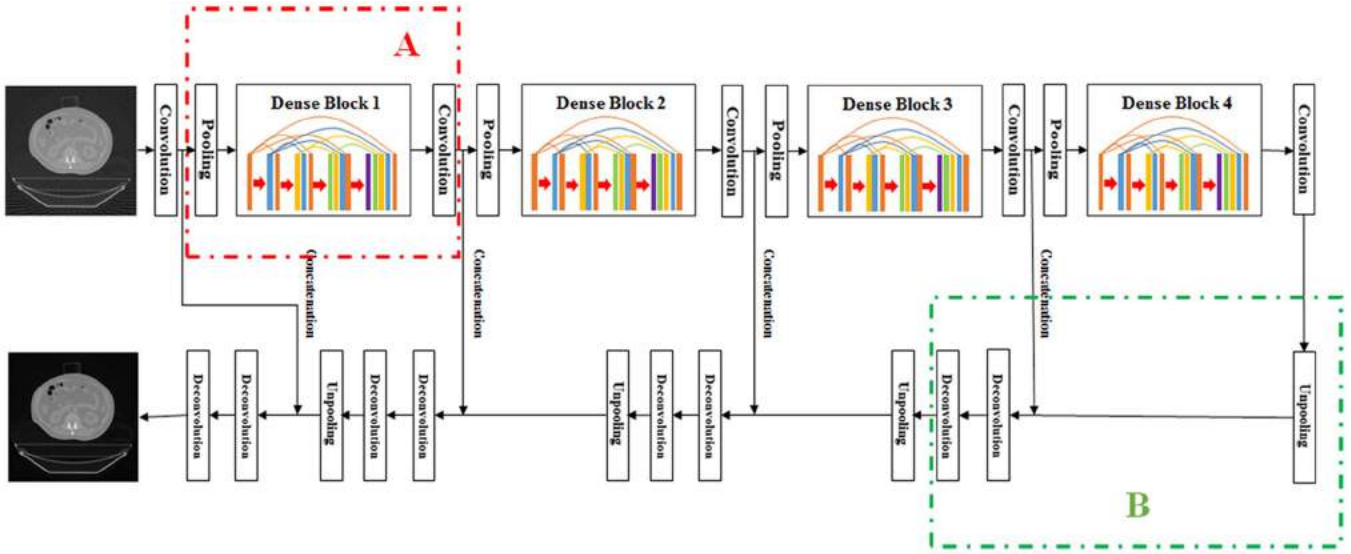

Fig. 2. The corresponding network architecture of the DD-Net.

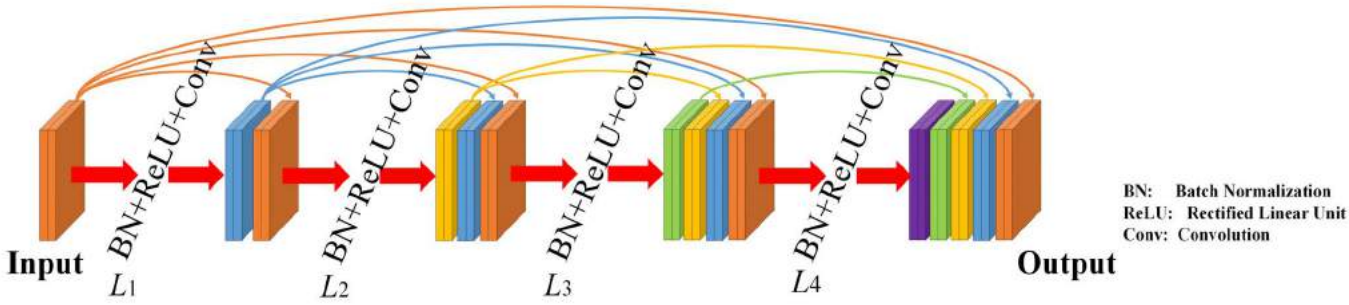

Fig. 3. A dense block with 4 L layers. Each L layer concatenates all preceding features together as its input.

connections: local shortcut connections within a dense block in the DenseNet and global ones between DenseNet network and deconvolution network. In previous deep neural networks such as ResNets, Highway network and FractalNets, shortcut connections from early layers to later layers are adopted to improve information flow to reduce the network training difficulty and avoid the gradient vanish or explosion, while simultaneously it helps to increase the depths or widths of the neural networks and improve results. In the same fashion, the paper uses shortcut connections in the DenseNet to obtain the same benefits.

With increase in the depth of neural network, the extracted features are more and more abstract. The extracted features keep only the main body structures of the CT image and lose the small textures. This problem can be addressed with the global shortcut connections by concatenating more features from the DenseNet network to act as input to the deconvolution network. In addition, when the network depth increases, the shortcut connections also help to suppress the gradient instability and make training the network a lot easier.

B. Network Parameter Selection

The proposed DD-Net is an end-to-end convolution neural network. It takes the CT images reconstructed with the FBP algorithm as the input. The input image is fed to a 7×7 convolution layer, followed by l (the number of denseblocks) A networks (marked as the red dot-dashed box in Figure 2) and l B networks (marked as the green dot-dashed box in Figure 2). In this paper, 2×2 MaxPooling and unpooling operators are employed before each dense block and the $n \times n$ (filter size) deconvolution layers, respectively. The 1×1 convolution layer and deconvolution layers are adopted after each dense block and the $n \times n$ deconvolution layers, respectively, to decrease the number of input feature-maps. In each dense block, h (the number of layers in a denseblock) L layers are used, and each L layer consists of Conv(1×1) and Conv($n \times n$). Shortcut connections are used to concatenate the output of A networks and the output of the unpooling operators. The parametric structure of all layers in the DD-Net is shown in Table 1. Because different parameter selections will lead to different performance, in this study we also tested the effects of different parameters on the performance of the network.

C. Network Training

Training the DD-Net network is essential to find a mapping function that improves the quality of FBP reconstructed CT images from sparse-view CT. Once the architecture of the

TABLE I
PARAMETRIC STRUCTURE FOR ALL THE LAYERS IN THE DD-NET

Layers	Parameters	Output Size (*e.g. k=16*)
Convolution	7×7conv	512×512×k
A1	2×2 MaxPooling	256×256×k
	Denseblock1	256×256×5k
	BN + ReLU+ 1×1 conv	256×256×k
A2	2×2 MaxPooling	128×128×k
	Denseblock2	128×128×5k
	BN + ReLU+ 1×1 conv	128×128×k
A3	2×2 MaxPooling	64×64×k
	Denseblock3	64×64×5k
	BN + ReLU+ 1×1 conv	64×64×k
A4	2×2 MaxPooling	32×32×k
	Denseblock3	32×32×5k
	BN + ReLU+ 1×1 conv	32×32×k
B1	2×2 unpooling + Concatenation	64×64×2k
	5×5deconv + ReLU+ BN	64×64×2k
	1×1deconv + ReLU+ BN	64×64×k
B2	2×2 unpooling + Concatenation	128×128×2k
	5×5deconv + ReLU+ BN	128×128×2k
	1×1deconv + ReLU+ BN	128×128×k
B3	2×2 unpooling + Concatenation	256×256×2k
	5×5deconv + ReLU+ BN	256×256×2k
	1×1deconv + ReLU+ BN	256×256×k
B4	2×2 unpooling + Concatenation	512×512×2k
	5×5deconv + ReLU	512×512×2k
	1×1deconv + ReLU	512×512×1

network is fixed, all the parameters can be estimated using backpropagation algorithm (BP) [51] by optimizing a loss function. Although the MSE (mean square error) loss function as fidelity term is widely used, it may blur edges, lose details, and ignore image structures [52]. Multi-scale structure similarity (MS-SSIM) [53] has been proposed to preserve the global structure similarity between images. Here, we use a mixed loss function that combines the MSE and MS-SSIM terms. It can be written as follow:

$$\mathcal{L} = ||Y - f(X)||_2^2 + \alpha \mathcal{L}_{MS-SSIM}(Y, f(X))$$

where $||Y - f(X)||_2^2$ is the MSE term, $\mathcal{L}_{MS-SSIM}$ is the MS-SSIM term, and α is the weighting factor for the MS-SSIM term; we used $\alpha = 0.1$.

The DD-Net was trained by the Adam algorithm [54]. The learning rate was initially set at 10^{-4} and slowly decreased continuously down to 10^{-5}. The size of mini-batch was 5. DD-Net was implemented using Tensorflow [55] on a personal workstation with Intel Core i5-7400 CPU and 16GB RAM. A GPU card (Nvidia GTX Titan X) accelerated the training process. All the convolution and deconvolution filters were initialized with random Gaussian distributions with zero mean and 0.01 standard deviation.

D. Datasets

A total of 3059 two-dimensional CT images of 512 × 512 pixels per image were obtained. The CT images were acquired from several parts of human body. Figure 4 illustrates some examples. From those original CT images, projection data were generated using the Beer's law under the assumption that the x-ray source was monochromatic at 60 keV. In the forward projection, Siddon's ray-driven forward projection method [56] was used, and Poisson noise was added according to formula $P_i \sim Poisson\left\{b_i e^{-l_i}\right\}, i = 1, \ldots, N$, where P_i is the detector measurement along the i^{th} ray path, b_i is the blank scan factor, and l_i is the line integral of attenuation coefficients along the i^{th} ray path. No electronic readout noise was simulated. The Poisson noise (and hence dose) level can be adjusted by setting the number of photons per ray for the blank scan factor b_i. In this study, b_i was uniformly set to 10^6 photons for each ray and denoted as $b_i = b_0 = 10^6, i = 1, \ldots, N$. Then, the sparse-view CT images with streaking artifacts were reconstructed using the FBP algorithm. The distance between source and detector was 1500 mm, and the distance between source and rotation center was 1100 mm. For each sparse-view CT, projections were gathered every three degrees in the fan-beam geometry and a total of 120 projections were used for reconstruction for each CT image. To train and verify the DD-Net, we divided all CT images into three sets: the training set, the validation set, and the testing set. The training set is a group of 2425 CT images downloaded from The Cancer Imaging Archive (TCIA) (about 16% head, 39% chest, 34% abdomen and 11% hip) [57]. The validation set is a group of 163 CT images downloaded from TCIA (about 14% head, 33% chest, 25% abdomen and 28% hip). The testing set is a group of 471 CT images downloaded from the National Biomedical Imaging Archive (NBIA) (about 22% head, 35% chest, 35% abdomen and 8% hip) [58]. The three datasets have different images.

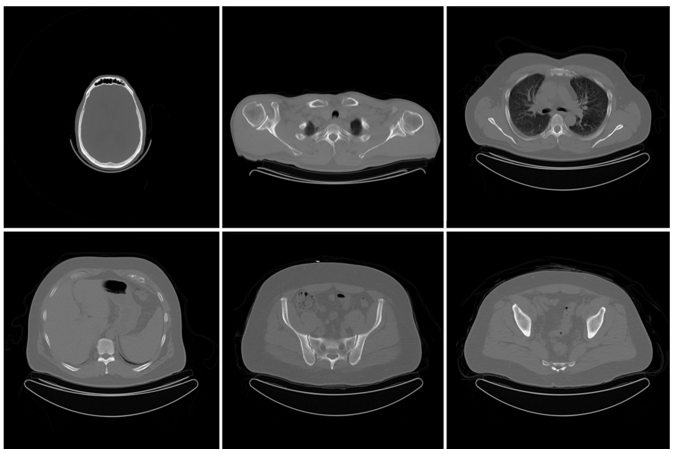

Fig. 4. Example CT slice images used for training and testing the DD-Net.

E. Evaluation Methods

In this study, two CT images (one from the chest region and one from the hip region) were randomly selected from the testing datasets to evaluate the performance of the proposed reconstruction method. The same two CT images were reconstructed by two other reconstruction methods (PWLS-TGV method [25], and R-NLTV method [24]), to compare with

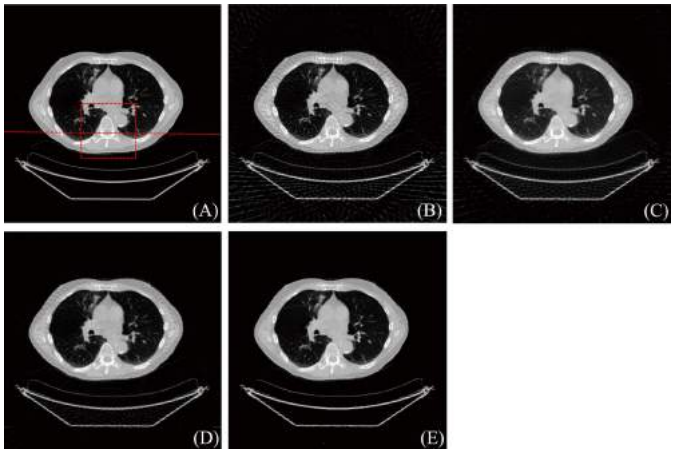

Fig. 5. Reconstruction results from the chest CT dataset. (A) Original image, (B) FBP, (C) PWLS-TGV, (D) R-NLTV, (E) DD-Net. Display window was set at [0, 0. 029] mm^{-1}.

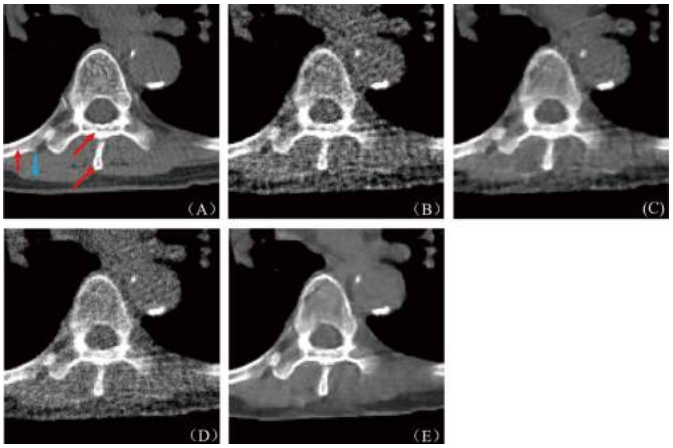

Fig. 6. The zoomed regions marked by the red box in Figure 5 (A). (A) Original image, (B) FBP, (C) PWLS-TGV, (D) R-NLTV, (E) DD-Net. The red arrows show the locations of the small features, and the blue arrow indicates the boundary between two soft tissues. Display window was set at [0.00164, 0. 0287] mm^{-1}.

the state-of-the-art reconstruction algorithms for sparse-view CT in literature. Both CT images were reconstructed from the 120 projections generated from the forward projection using the PWLS-TGV, R-NLTV and DD-Net. The initial value for all the methods was the FBP reconstructed CT image.

Image assessments were carried out qualitatively, such as visual inspections and line intensity profile comparisons, as well as quantitatively, such as comparisons for root-mean-square error (RMSE) and structural similarity (SSIM). Haralick texture features [59], [60] were also used to quantify the capability of texture preservation. All 14 Haralick texture measures were calculated from each entire image. Then, the Euclidean distance between the textures of the original CT image and a newly reconstructed CT image was calculated as a quantitative measure [61]. The shorter the Euclidean distance, the better the texture preservation.

To test the performance of the reconstruction algorithms in various sparse-view-sampling conditions, CT images were also reconstructed with 60, 72, 90 and 180 projections and analyzed qualitatively as well as quantitatively.

III. Results

A. Experimental Results

1) Chest CT Image: Figure 5 shows the results from the chest dataset. As expected, severe streaking artifacts can be observed in the FBP reconstructed CT image in Figure 5 (B). For the PWLS-TGV and R-NLTV methods, there are trade-offs between the effect of artifact removal and image resolution. We tried our best to tune the parameters under the suggestions from the original papers to preserve image details as much as possible. From (C) to (E), it can be seen that the streaking artifacts were suppressed to different degrees by different methods. Through visual inspection, DD-Net is best for artifact removal and preserving small features.

Figure 6 shows the zoomed regions marked by the dashed red box in Figure 5 (A). Compared to PWLS-TGV, the image from DD-Net has more well-defined boundaries and more details. Compared to R-NLTV, DD-Net is better at suppressing the image noise. All three methods somewhat smoothened out the small features indicated by the red arrows in Figure 6 (A). However, the DD-Net method performed slightly better than the PWLS-TGV and the R-NLTV methods, and retained part of those small details. The boundary between the two soft tissues marked by the blue arrow in Figure 6 (A) is still visible in the CT image from the DD-Net method, while it is more blurry in the CT images from the FBP, PWLS-TGV, and R-NLTV methods.

TABLE II
Quantitative Results From the Different Reconstruction Algorithms for the Chest CT Image

	FBP	PWLS-TGV	R-NLTV	DD-Net
RMSE(mm^{-1})	0.0019	0.00092	0.0011	**0.00067**
SSIM	0.47	0.86	0.79	**0.93**
Haralick	2.16	1.31	1.95	**1.12**

Quantitative analysis for the entire chest CT image reconstructed using these methods has also been carried out and the results are shown in Table 2. DD-Net clearly outperformed the other methods and produced the highest SSIM and lowest RMSE. The Haralick measure for the DD-Net image had fallen by about 48% compared to the result from FBP. The smallest Haralick measure for the DD-Net image indicates that the DD-Net has the strongest capability of texture preservation.

Figure 7 plots the 1D line intensity profile passing through the red dashed line in Figure 5 (A). It compares the same line intensity profiles from the CT image reconstructed by various methods. Through visual inspection, it is clear that the line intensity profile from our proposed method resembles most closely to the one from the original CT image. The comparison demonstrated the advantage of the proposed reconstruction method over the other iterative reconstruction algorithms on edge preservation.

To better illustrate the effectiveness of artifact removal by DD-Net, Figure 8 shows the absolute difference images relative to the original CT image in Figure 5 (A). It is clearly

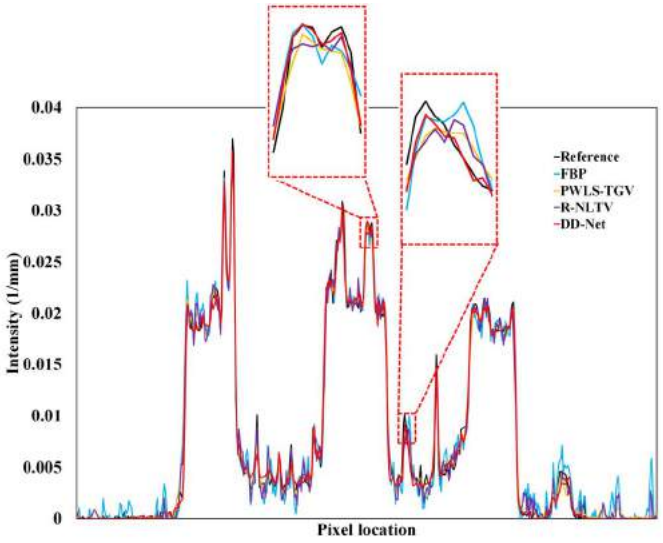

Fig. 7. 1D intensity profile passing through the red dashed line in Figure 5 (A). The insets clearly show the better performance from the DD-Net compared to the other methods.

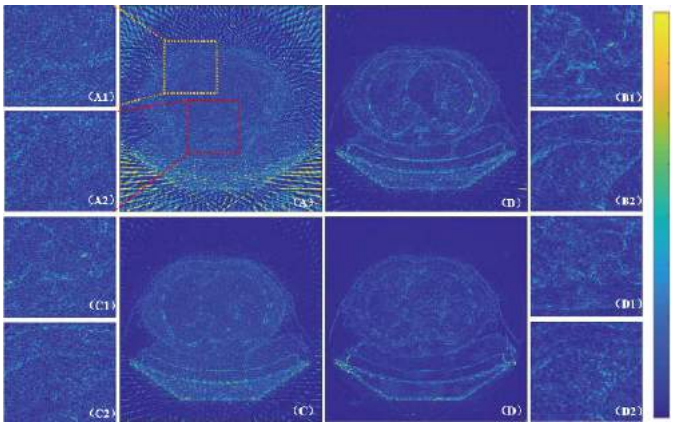

Fig. 8. The absolute difference images between the original CT image and the CT images reconstructed from the different algorithms: (A) FBP, (A1) enlarged area marked by orange box and (A2) enlarged area marked by red box; (B) PWLS-TGV, (B1) and (B2) are corresponding enlarged areas; (C) R-NLTV, (C1) and (C2) are corresponding enlarged areas; and (D) DD-Net, (D1) and (D2) are corresponding enlarged areas. Display window was set at [0, 0.008] mm^{-1}. A larger view of the figure is available in the supplemental document.

observed from the enlarged area marked by red and orange boxes in Figure 8 (A) that DD-Net has the smallest difference, and the results in Figure 8 (C) and (D) still include visible streaking artifacts and other structures. In Figure 8, the darker the color, the smaller the error.

2) Hip CT Image: For the hip CT image, results reconstructed from the different reconstruction methods are shown in Figure 9. It can be observed clearly that all the methods (PWLS-TGV, R-NLTV, and DD-Net) can suppress streaking artifacts. However, the result from the PWLS-TGV method suffered a bit from over-smoothing, and the streaking artifacts have not been completely eliminated from the R-NLTV method.

To show the performance of different reconstruction methods at local regions, ROIs as marked by the blue dashed

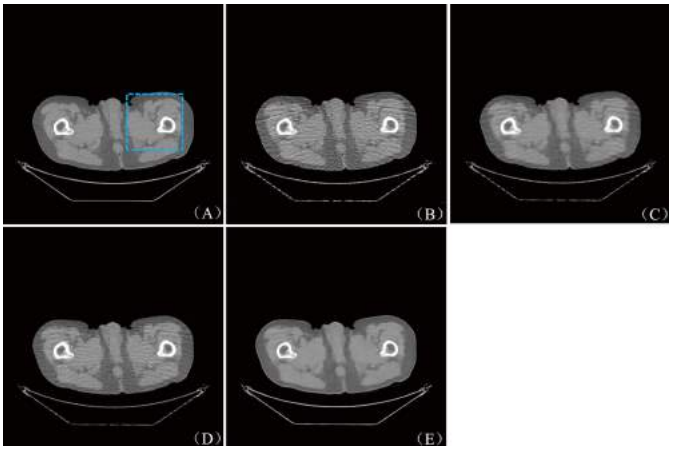

Fig. 9. Results from the hip dataset. (A) Original image, (B) FBP, (C) PWLS-TGV, (D) R-NLTV, (E) DD-Net. Display window was set at [0.012, 0.029] mm^{-1}.

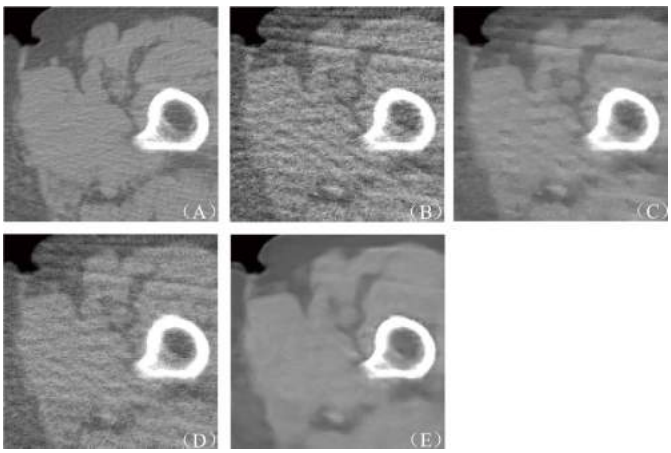

Fig. 10. The zoomed region marked by the blue box in Figure 9 (A). (A) Original image, (B) FBP, (C) PWLS-TGV, (D) R-NLTV, (E) DD-Net. Display window was set at [0.012, 0.029] mm^{-1}.

TABLE III
QUANTITATIVE RESULTS FROM DIFFERENT RECONSTRUCTION METHODS FOR THE HIP CT IMAGE

	FBP	PWLS-TGV	R-NLTV	DD-Net
RMSE(mm^{-1})	0.0018	0.00075	0.00083	**0.00048**
SSIM	0.49	0.90	0.88	**0.96**
Haralick	2.37	1.09	1.44	**0.61**

rectangle in Figure 9 (A) are enlarged to compare them in details in Figure 10. The quantitative results for the entire hip CT image reconstructed using these methods are listed in Table 3. It is clear that DD-Net performed better than the other methods in a trend similar to what we have seen from the chest CT image.

B. Network Parameter Tuning

The DD-Net has several parameters to be optimized, including the number of denseblocks l, the number of layers in a denseblock h, the number of filters k, and the filter size n. The optimum value for each parameter was determined by

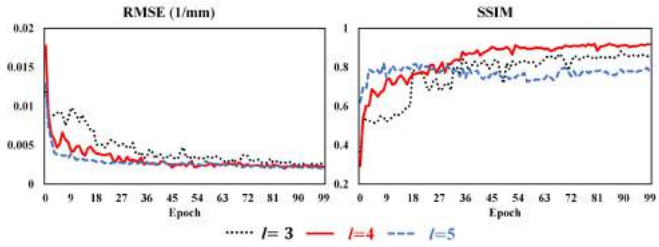

Fig. 11. RMSE and SSIM results on the validation dataset during the network trainings. The display ranges of RMSE and SSIM are [0, 0.02] and [0.2, 1], respectively.

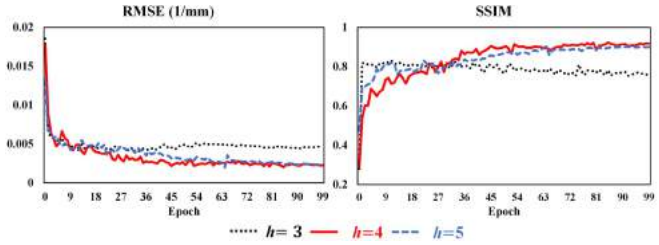

Fig. 12. RMSE and SSIM results on the validation dataset during the network trainings. The display ranges of RMSE and SSIM are [0, 0.02] and [0.2, 1], respectively.

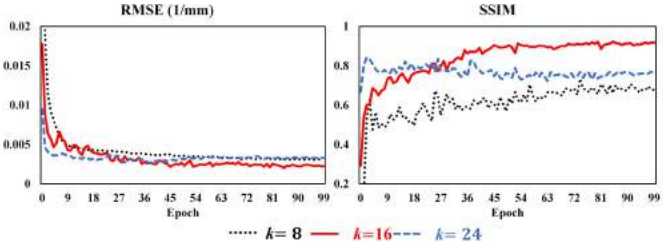

Fig. 13. RMSE and SSIM results on the validation dataset during the network trainings. The display ranges of RMSE and SSIM are [0, 0.02] and [0.2, 1], respectively.

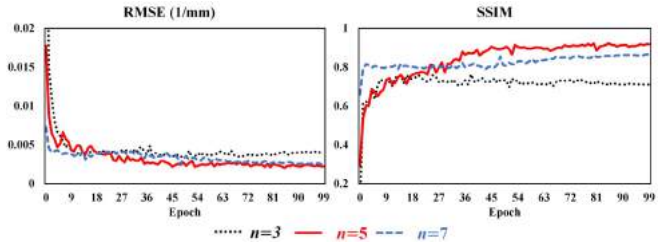

Fig. 14. RMSE and SSIM results on the validation dataset during the network trainings. The display ranges of RMSE and SSIM are [0, 0.02] and [0.2, 1], respectively.

perturbing one parameter while keeping the others fixed and analyzing the resulted quantitative performance in the reconstructed CT images for the validation dataset. The average RMSE and SSIM values of all the CT images in the validation set were calculated to plot the curves in Figures 11-16.

1) Number of Denseblocks: For this parameter, we changed l into three possible values: 3, 4 and 5. The corresponding results are shown in Figure 11. It can be seen that the RMSE and SSIM become better from 3 blocks to 4 blocks, while they become slightly worse from 4 blocks to 5 blocks. One possible

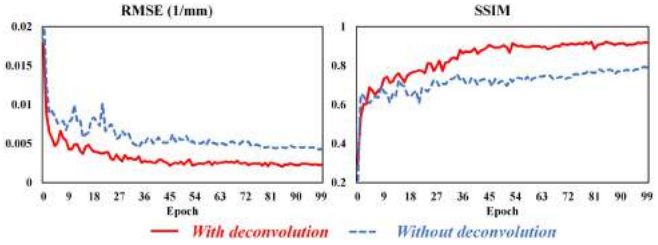

Fig. 15. RMSE and SSIM results on the validation dataset during network trainings. The display ranges of RMSE and SSIM are [0, 0.02] and [0.2, 1], respectively.

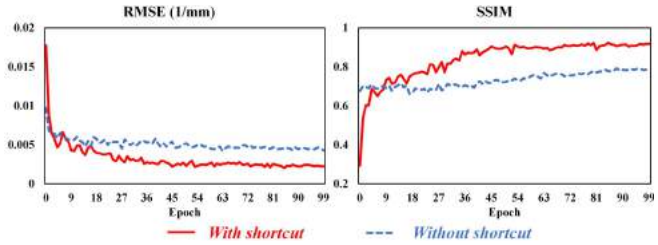

Fig. 16. RMSE and SSIM results on the validation dataset during the network trainings. The display ranges of RMSE and SSIM are [0, 0.02] and [0.2, 1], respectively.

explanation is that as the depth of the network increases, more training samples are needed to avoid overfitting. Therefore, we determined that the best number of denseblocks l is at 4.

2) Number of Layers in One Denseblock: Different h (3, 4 and 5) were tested. We changed the number of layers in one denseblock and trained the networks under the same other conditions. The results are shown in Figure 12. As the number of layers in one denseblock was increased from 3 to 4, the performance was improved. While when h was increased from 4 to 5, the performance was barely changed. Meanwhile, the training time and the number of parameters in the neural network will significantly increase. Therefore, we determined that the best number of layers in one denseblock h is at 4.

3) Number of Filters: A comparative trial on the number of filters has been carried out. We found that the performance is best when $k = 16$, as shown in Figure 13. The potential explanation is that as the width of the network increases, the number of parameters on the network increased significantly. It may be more difficult to converge. Therefore, we determined that the best number of filters k is 16.

4) Impact of Filter Size: Convolution neural networks differ from other neural networks by introducing the concept of receptive field (i.e., filters). The filter size affects the performance of CNNs. Larger n will help extract more abstract feature-maps but it leads to the loss of many details and the increase in the number of parameters. From the experimental result shown in Figure 14, network with 5×5 filters outperform others with 3×3 and 7×7 filters. Therefore, we determined that the best filter size n is 5.

C. Computational Cost

Computational cost is an important factor for any reconstruction algorithm. All the algorithms were carried out with Matlab except the DD-Net, which was implemented with

TABLE IV
QUANTITATIVE RESULTS FROM DIFFERENT RECONSTRUCTION METHODS USING DIFFERENT NUMBERS OF PROJECTIONS

	Method	60	72	90	120	180
RMSE (mm^{-1})	FBP	0.0032	0.0027	0.0023	0.0019	0.0015
	PWLS-TGV	**0.0015**	0.0013	0.0011	0.00092	0.00073
	R-NLTV	0.0017	0.0015	0.0013	0.0011	0.00086
	DD-Net	0.0016	**0.0012**	**0.00089**	**0.00067**	**0.00057**
SSIM	FBP	0.36	0.38	0.42	0.47	0.57
	PWLS-TGV	**0.80**	0.81	0.84	0.86	0.89
	R-NLTV	0.71	0.72	0.75	0.79	0.83
	DD-Net	**0.80**	**0.86**	**0.90**	**0.93**	**0.94**

Python. The training process for a DL method is very time-consuming. Although the algorithms were realized using different programming languages, the efficiency can be roughly compared on a same computer. In this study, the average computing time for FBP, PWLS-TGV, R-NLTV, and DD-Net (after training) is 1.2, 424.5, 613.5, and 1.3 seconds per CT slice, respectively. As can be seen, once DD-Net is fully trained, it owns a considerable advantage in computational efficiency compared to the other iterative reconstruction methods.

In this study, after the DD-Net was trained using the training set, FBP reconstruction from a sparse-view CT with 120 projections was used as the input for the DD-Net. However, in actual application, one will not always collect 120 projections in all situations. To verify the robustness of the DD-Net, four other tests have been carried out using FBP reconstructed chest CT image from 180, 90, 72 and 60 projections using the same well-trained DD-Net, respectively. These results are shown in Table 4. With only 60 projections, PWLS-TGV outperformed other methods in terms of the quantitative metrics (RMSE and SSIM). In other cases, DD-Net always obtained the best results. We think this can be explained by the fact that the DD-Net method only works in the image domain and it utterly relies on the results from the FBP reconstruction. For better demonstration, corresponding CT image appearance for the different number of projections is included in supplement materials. Supplementary materials are available in the supplementary files /multimedia tab.

IV. DISCUSSION AND CONCLUSION

In this paper, we described an image reconstruction method for sparse-view CT. The method is based on a DL neural network that combines DenseNet and deconvolution. The DenseNet and deconvolution were united with shortcut connections, such that an end-to-end network – DD-Net – was formed to significantly increase the depth and the expression ability of the network. The network takes a FBP reconstructed CT image from sparse-view CT as input and tries to produce an enhanced, artifact-free CT image. We trained the neural network using a training set, and applied the well-trained neural network to two sets of clinical CT data acquired from different parts of human body. The results show that the proposed method can effectively suppress the streaking artifact induced by missing data from sparse-view CT, and can partially recover image details for sparse-view CT.

Conventional iterative reconstruction algorithms have been extensively used in image reconstruction for sparse-view CT. When compared traditional reconstruction algorithms such as FBP, iterative reconstruction algorithms exhibit significant image improvement. For instance, PWLS-TGV [25] and R-NLTV [24] can outperform FBP or SART-TV [11]. Nevertheless, through our experiments, PWLS-TGV suffers a bit from over-smoothing, and R-NLTV still shows unsatisfactory noise suppression effect.

Compared to the other reconstruction methods for sparse-view CT, the DD-Net method has its own advantages. Compared to PWLS-TGV and R-NLTV, DD-Net can increase SSIM by up to 18% and reduce RMSE by up to 42%, for sparse view CT with 120 projections. Compared to the popular iterative reconstruction algorithms, the DD-Net method is much faster, although it takes time to train the network. Once the DD-Net neural network has been trained, the computation time for CT reconstruction using the trained network is just about 1 second per slice on a PC workstation (Intel Core i5-7400 CPU, 16GB RAM and Nvidia GTX Titan X GPU card). In comparison, the average computing time for PWLS-TGV and R-NLTV is 425.5 and 613.5 seconds per CT slice, respectively.

In this work, we used deconvolution layers to restore CT images from extracted feature-maps. It has been previously demonstrated that convolution has excellent performance on image restoration from extracted feature-maps [35], [62], [63]. To examine the difference between deconvolution and convolution, we replaced the deconvolution layers by convolution layers in the DD-Net, and then trained the new network using the same training datasets. We then evaluated its sparse-view CT reconstruction at each epoch (defined as a single forward and backward pass of all the training datasets through the entire network). We compared the image quality (in terms of the average values of RMSE and SSIM of all the CT images in the validation set) of the sparse-view CT image at each epoch from the two network versions. The comparison results are shown in Figure 15. It can be seen that DD-Net with the deconvolution layer outperformed the counterpart with convolution layer.

In DD-Net, the shortcut connections to concatenate DenseNet and deconvolution layers played a significant role during network training. The influence of shortcut connections on the DD-Net was evaluated in a way similar to the one used to evaluate the effect of deconvolution versus convolution, by comparing the quality of a sparse-view CT image at each epoch of the two network versions with and without shortcut connections. The results are shown in Figure 16. It can be seen that significantly better RMSE and SSIM results can be obtained with shortcut connections. One potential explanation is that shortcut connections improve the information flow and shorten the distance between input and output, and all layers receive additional supervision from the loss function through the shortcut connections.

In this study, the outcomes of the DD-Net method heavily depend on the FBP reconstruction results, which were fed as inputs to the DD-Net. The improvement in image quality from the DD-Net method could be barely satisfactory, if the input images from FBP reconstruction exhibit severe artifacts due to the loss of a large amount of information from the missing

views in a sparse-view CT scan. In the future, the missing information from a sparse-view CT scan could be compensated to a certain extent by a big and relevant training dataset (e.g. same organ or same diseases). The training dataset could somehow make up the missing information in one sparse-view CT scan by the complementary information from the many other CT scans in the training dataset. However, the detailed improvement and associated computing cost deserve a thorough study.

In conclusion, leveraging the advantages of DenseNet and deconvolution, we proposed a new DL neural network based reconstruction method (DD-Net) for sparse-view CT. The advantage of the DD-Net lies in the fact that it uses feature reuse, deconvolution, and shortcut connections to increase the depth and expressive ability of the neural network, to effectively restore image details, and to accelerate the training speed of the neural network. Results from the experiments using two sets of CT data from different parts of human body demonstrated the great potential of this technique in improving the quality of CT images from sparse-view CT. The proposed method could lead to high-quality CT imaging with faster imaging speed and lower radiation dose.

REFERENCES

[1] A. M. Cormack, "Representation of a function by its line integrals, with some radiological applications. II," *J. Appl. Phys.*, vol. 35, no. 10, pp. 2908–2913, 1964.

[2] G. N. Hounsfield, "Computerized transverse axial scanning (tomography). 1. Description of system," *Brit. J. Radiol.*, vol. 46, no. 552, pp. 1016–1022, 1973.

[3] G. Wang, H. Yu, and B. De Man, "An outlook on X-ray CT research and development," *Med. Phys.*, vol. 35, no. 3, pp. 1051–1064, 2008.

[4] H. Chen *et al.*, "Low-dose CT with a residual encoder-decoder convolutional neural network," *IEEE Trans. Med. Imag.*, vol. 36, no. 12, pp. 2524–2535, Dec. 2017.

[5] 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.

[6] M. K. Kalra *et al.*, "Strategies for CT radiation dose optimization," *Radiology*, vol. 230, no. 3, pp. 619–628, Mar. 2004.

[7] T. L. Slovis, "The ALARA concept in pediatric CT: Myth or reality?" *Pediatric Radiol.*, vol. 223, no. 1, pp. 5–6, 2002.

[8] Q. Xu, H. Yu, X. Mou, L. Zhang, J. Hsieh, and G. Wang, "Low-dose X-ray CT reconstruction via dictionary learning," *IEEE Trans. Med. Imaging*, vol. 31, no. 9, pp. 1682–1697, Sep. 2012.

[9] Y. Chen *et al.*, "Artifact suppressed dictionary learning for low-dose CT image processing," *IEEE Trans. Med. Imag.*, vol. 33, no. 12, pp. 2271–2292, Dec. 2014.

[10] J. Wang, T. Li, H. Lu, and Z. Liang, "Penalized weighted least-squares approach to sinogram noise reduction and image reconstruction for low-dose X-ray computed tomography," *IEEE Trans. Med. Imag.*, vol. 25, no. 10, pp. 1272–1283, Oct. 2006.

[11] 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.

[12] G.-H. Chen, J. Tang, and S. Leng, "Prior image constrained compressed sensing (PICCS): A method to accurately reconstruct dynamic CT images from highly undersampled projection data sets," *Med. Phys.*, vol. 35, no. 2, pp. 660–663, 2008.

[13] 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.

[14] J. Bian *et al.*, "Evaluation of sparse-view reconstruction from flat-panel-detector cone-beam CT," *Phys. Med. Biol.*, vol. 55, no. 22, pp. 6575–6599, 2010.

[15] M. E. Davison, "The ill-conditioned nature of the limited angle tomography problem," *SIAM J. Appl. Math.*, vol. 43, no. 2, pp. 428–448, 1983.

[16] S. Li, Q. Cao, Y. Chen, Y. Hu, L. Luo, and C. Toumoulin, "Dictionary learning based sinogram inpainting for CT sparse reconstruction," *Opt. J. Light Electron Opt.*, vol. 125, no. 12, pp. 2862–2867, 2014.

[17] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[18] H. Lee, J. Lee, and S. Cho, "View-interpolation of sparsely sampled sinogram using convolutional neural network," *Proc. SPIE*, vol. 10133, p. 1013328, Feb. 2017.

[19] A. Krizhevsky and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[20] M. Beister, D. Kolditz, and W. A. Kalender, "Iterative reconstruction methods in X-ray CT," *Phys. Med.*, vol. 28, no. 2, pp. 94–108, Apr. 2012.

[21] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[22] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[23] J.-F. Cai, X. Jia, H. Gao, S. B. Jiang, Z. Shen, and H. Zhao, "Cine cone beam CT reconstruction using low-rank matrix factorization: Algorithm and a proof-of-principle study," *IEEE Trans. Med. Imag.*, vol. 33, no. 8, pp. 1581–1591, Aug. 2014.

[24] H. Kim, J. Chen, A. Wang, C. Chuang, M. Held, and J. Pouliot, "Non-local total-variation (NLTV) minimization combined with reweighted L1-norm for compressed sensing CT reconstruction," *Phys. Med. Biol.*, vol. 61, no. 18, pp. 6878–6891, 2016.

[25] S. Niu *et al.*, "Sparse-view X-ray CT reconstruction via total generalized variation regularization," *Phys. Med. Biol.*, vol. 59, no. 12, pp. 2997–3017, 2014.

[26] H. Nien and J. A. Fessler, "Relaxed linearized algorithms for faster X-ray CT image reconstruction," *IEEE Trans. Med. Imag.*, vol. 35, no. 4, pp. 1090–1098, Apr. 2016.

[27] D. Kim, S. Ramani, and J. A. Fessler, "Combining ordered subsets and momentum for accelerated X-ray CT image reconstruction," *IEEE Trans. Med. Imag.*, vol. 34, no. 1, pp. 167–178, Jan. 2015.

[28] K. Grant and T. Flohr, "Iterative reconstruction in image space (IRIS)," Siemens Medical Solutions Whitepaper, 2010. [Online]. Available: http://www.medical.siemens.com/siemens/en_US/gg_ct_FBAs/files/Case_Studies/IRIS_Whitepaper.pdf

[29] Y. S. Han, K. H. Jin, K. Kim, and J. C. Ye, "Sparse-view X-ray spectral CT reconstruction using annihilating filter-based low rank Hankel matrix approach," in *Proc. IEEE 13th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2016, pp. 573–576.

[30] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[31] T. Würfl, F. C. Ghesu, V. Christlein, and A. Maier, "Deep learning computed tomography," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, 2016, pp. 432–440.

[32] X. F. Ma, M. Fukuhara, and T. Takeda, "Neural network CT image reconstruction method for small amount of projection data," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 449, nos. 1–2, pp. 366–377, 2000.

[33] L. Cheng, S. Ahn, S. G. Ross, H. Qian, and B. De Man, "Accelerated iterative image reconstruction using a deep learning based leapfrogging strategy," in *Proc. Int. Conf. Fully Three-Dimensional Image Reconstruct. Radiol. Nucl. Med.*, 2017, pp. 715–720.

[34] H. Analysis, Y. S. Han, J. Yoo, and J. C. Ye. (2016). "Deep residual learning for compressed sensing CT reconstruction via persistent homology analysis." [Online]. Available: https://arxiv.org/abs/1611.06391

[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] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, 2015, pp. 234–241.

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

[38] G. Wang, "A perspective on deep imaging," *IEEE Access*, vol. 4, pp. 8914–8924, 2016.

[39] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. (Jan. 2016). "Densely connected convolutional networks." [Online]. Available: https://arxiv.org/abs/1608.06993

[40] L. Xu, J. S. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1790–1798.

[41] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximator," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, 1989.
[42] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[43] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.
[44] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Proc. NIPS*, 2015, pp. 2377–2385.
[45] G. Larsson, M. Maire, and G. Shakhnarovich. (2016). "FractalNet: Ultra-deep neural networks without residuals." [Online]. Available: https://arxiv.org/abs/1605.07648
[46] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 630–645.
[47] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2015, pp. 1520–1528.
[48] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.
[49] S. Hong, H. Noh, and B. Han, "Decoupled deep neural network for semi-supervised semantic segmentation," in *Proc. NIPS*, 2015, pp. 1495–1503.
[50] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Learning to deblur," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 7, pp. 1439–1451, Jul. 2016.
[51] R. Hecht-Nielsen, "Theory of the backpropagation neural network," *Neural Netw.*, vol. 1, pp. 445–448, Jun. 1988.
[52] 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.
[53] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 1–11, 2017.
[54] D. P. Kingma and J. Ba. (Jan. 2014). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980
[55] M. Abadi *et al.* (Mar. 2016). "TensorFlow: Large-scale machine learning on heterogeneous distributed systems." [Online]. Available: https://arxiv.org/abs/1603.04467
[56] R. L. Siddon, "Fast calculation of the exact radiological path for a three-dimensional CT array," *Med. Phys.*, vol. 12, no. 2, pp. 252–255, 1985.
[57] *The Cancer Imaging Archive (TCIA)*. Accessed: Dec. 2014. [Online]. Available: http://www.cancerimagingarchive.net/
[58] *National Biomedical Imaging Archive (NBIA)*. Accessed: 1999. [Online]. Available: https://imaging.nci.nih.gov/ncia/login.jsf
[59] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.
[60] L. Tesař, A. Shimizu, D. Smutek, H. Kobatake, and S. Nawano, "Medical image analysis of 3D CT images based on extension of Haralick texture features," *Comput. Med. Imag. Graph.*, vol. 32, no. 6, pp. 513–520, 2008.
[61] H. Zhang *et al.*, "Extracting information from previous full-dose CT scan for knowledge-based Bayesian reconstruction of current low-dose CT images," *IEEE Trans. Med. Imag.*, vol. 35, no. 3, pp. 860–870, Mar. 2016.
[62] A. Radford, L. Metz, and S. Chintala. (Jan. 2015). "Unsupervised representation learning with deep convolutional generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1511.06434
[63] C. Wang, C. Xu, and C. Wang. (Jun. 2017). "Perceptual adversarial networks for image-to-image transformation." [Online]. Available: https://arxiv.org/abs/1706.09138