

R²-Net: Recurrent and Recursive Network for Sparse-View CT Artifacts Removal

Tiancheng Shen¹, Xia Li², Zhisheng Zhong², Jianlong Wu²,³, and Zhouchen $Lin^{2(\boxtimes)}$

- ¹ Center for Data Science, Peking University, Beijing 100871, China tianchengShen@pku.edu.cn
- ² Key Laboratory of Machine Perception (MOE), School of EECS, Peking University, Beijing 100871, China {ethanlee,zzs1994,jlwu1992,zlin}@pku.edu.cn
- ³ School of Computer Science and Technology, Shandong University, Tsingtao 266000, China

Abstract. We propose a novel neural network architecture to reduce streak artifacts generated in sparse-view 2D Computed Tomography image reconstruction. This architecture decomposes the streak artifacts removal into multiple stages through the recurrent mechanism, which can fully utilize information in previous stages and guide the learning of later stages. In each recurrent stage, the key components of the architecture operate recursively. The recursive mechanism is helpful to save parameters and enlarge the receptive field efficiently with exponentially increased dilation of convolution. To verify its effectiveness, we conduct experiments on the AAPM's CT dataset through 5-fold cross-validation. Our proposed method outperforms the state-of-the-art methods both quantitatively and qualitatively.

Keywords: Computed Tomography · Sparse-view reconstruction · Convolutional recurrent neural network

1 Introduction

In the past twenty years, the radiation risk issue of CT receives much attention and the demand for radiation dose reduction becomes more intense. One way to reduce radiation dose and shorten acquisition time is the sparse-view CT reconstruction, which is achieved by reducing the number of radiation angles,

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-32226-7_36) contains supplementary material, which is available to authorized users.

T. Shen and X. Li—Equal contribution.

[©] Springer Nature Switzerland AG 2019 D. Shen et al. (Eds.): MICCAI 2019, LNCS 11769, pp. 319–327, 2019.

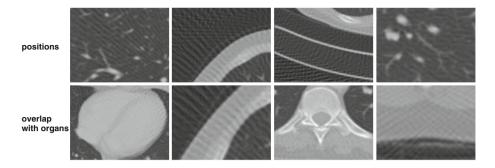


Fig. 1. Streak artifacts in different positions or overlapped with different organs have different characteristics.

i.e., views. However, this process introduces some streak artifacts, thus reduces the image quality and causes it confusing for doctors to diagnose.

Great efforts have been devoted to improving sparse-view CT reconstruction's quality. Existing approaches to address the streak artifacts can be mainly divided into two categories: (1) Classical methods: ASD-POCS [5] and PICCS [6] are based on the compressed sensing theory; AwTV [7] and TVS-POCS [8] are total variation based methods; ASDL [9] and AS-LNLM [10] develop from dictionary learning, and so on. (2) Deep learning methods: almost all state-of-theart (SOTA) deep learning methods are based on U-Net [11] framework, such as Tight Frame U-Net (TF U-Net) [3], cascade of U-Nets [12], etc. In terms of PSNR and SSIM, deep learning methods outperform traditional ones.

In most mainstream deep learning algorithms, there always exist two weaknesses. (1) It is supposed that a sparse-view CT image I_s can be decomposed into a dense-view CT image I_d and a streak artifacts image A as follows:

$$I_s = I_d + A. (1)$$

However, if we observe the sparse-view CT images more carefully, we will find that the streak artifacts in different positions or overlapped with different organs have different characteristics, which is shown in Fig. 1. So it is more proper to decompose the original sparse-view CT image into a dense-view CT image and several different mixture modes of streak artifacts, which can be formulated as:

$$I_s = I_d + \sum_{i=1}^n \alpha_i A_i, \tag{2}$$

where A_i represents a certain mixture mode of streak artifacts' image, and α_i denotes the intensity of this certain kind of streak artifacts mixture mode. (2) In order to enlarge the receptive field size, U-Net adopts the "contracting path" and "expansive path" [11]. This feature's downsampling design saves the computation, however, introduces many more parameters. Most SOTA deep learning methods are based on U-Net, so they also suffer from this weakness.

In order to address the above two issues, we propose an architecture, named R^2 -Net. On the one hand, we decompose the streak artifacts removal into multiple stages and adopt the **recurrent** mechanism. The visualization results about different mixture modes of streak artifacts in various stages and the progressive learning process are shown in Fig. 4. Moreover, we incorporate the SE block [1] to explicitly learn the α_i in Eq. (2). On the other hand, our proposed network also has a large receptive field without contracting and expanding. This scheme is achieved by the **recursive** mechanism and exponentially increased dilation of convolution. Because of the two mechanisms, we call our model \mathbb{R}^2 -Net.

Main contributions of this paper are listed as follows:

- 1. We propose a novel network with the recurrent mechanism to remove streak artifacts in sparse-view CT images stage by stage. Besides, SE blocks [1] are used to assign different alpha-values to various mixture modes of streak artifacts in one recurrent stage.
- 2. In order to achieve capability comparable to U-Net-like networks, recursive mechanism and exponentially increased dilation are utilized in our proposed model. Our network requires fewer parameters than U-Net-like networks.
- 3. To the best of our knowledge, this is the first paper to introduce recurrent and recursive mechanisms into the artifacts removal of sparse-view CT. Experiments show that R²-Net outperforms several SOTA methods on the American Association of Physicists in Medicine (AAPM) CT dataset [13].

2 Method

2.1 Overview

Our proposed R^2 -Net consists of several components, including an encoder network \mathbf{E} to transform a 2D CT image to feature maps, a recursive transformer \mathbf{T} in the feature space and a decoder network \mathbf{D} to estimate the streak artifacts.

For the overall framework, it contains several **recurrent** stages, and each recurrent stage has several **recursive** stages. In one recurrent stage, firstly, a sparse-view CT image I_s is used as input of the encoder network **E**. Then the recursive transformer **T** extracts feature maps recursively.

Later on, the decoder \mathbf{D} aggregates refined feature maps from different recursive stages of \mathbf{T} and estimates the streak artifacts. In the next recurrent stage, the output and feature maps of the previous recurrent stage are used as input and hidden states to predict the streak artifacts more precisely.

In the following, we first describe the architecture of the base model, i.e., the model in one recurrent stage. Then we describe the recurrent mechanism.

2.2 Base Model

The base model, i.e., the model in one recurrent stage, of our proposed method is illustrated in Fig. 2. It is a forward network that transforms I_s to an ideally artifacts-free image \widehat{I}_d that looks like the dense-view CT image I_d .

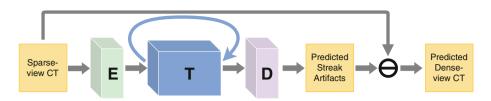


Fig. 2. The architecture of our proposed method in one recurrent stage.

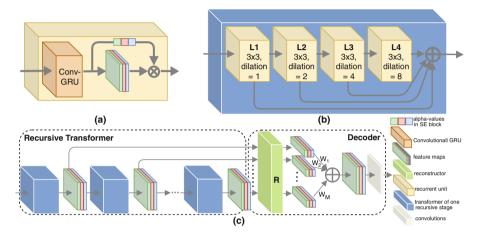


Fig. 3. (a) Basic unit. (b) The recursive transformer in one recursive stage, i.e., a ConvGRU group with exponentially increasing dilations and the sum of all layers' output. (c) Unrolling recursive transformer, reconstructor, the weighted sum of feature maps and final convolutions, i.e., $\bf T$ and $\bf D$ in Fig. 2. The number of the recursive stages is M in the figure.

Basic Unit. In our proposed model, we choose Convolutional GRU (ConvGRU) as the basic component, which is shown in Fig. 3(a). ConvGRU includes two convolutional kernels: one on the input tensor and the other on the hidden state tensor. The input tensor comes from the previous layer. The hidden state tensor comes from the same layer in the previous recurrent stage, which is 'zeroes' for the first stage. This design intends to adapt to our recurrent mechanism.

Besides, we regard each channel of ConvGRU's output as the embedding of one mixture mode of streak artifacts. So we extend ConvGRU with the SE block [1] to explicitly compute the alpha-value (α_i) for every channel. Through multiplying alpha-values, feature maps are reweighted to better fit the distribution of streak artifacts with different characteristics.

Encoder E. The encoder is comprised of a basic unit and designed to transform a sparse-view CT image to feature maps.

Recursive Transformer T. This module is recursively used in the base model and can be unrolled as Fig. 3(c). T's convolution parameters are shared. Through

this recursive mechanism, we can enlarge the receptive field without introducing more convolution parameters. In detail, if the recursive number is M and the original receptive field size is S^2 , the final receptive field size will be $(MS)^2$.

As for the inner structure of \mathbf{T} , due to that larger receptive field is very helpful to acquire more contextual information, dilation scheme is adopted in our recursive transformer. As shown in Fig. 3(b), for layers L1 to L4, the dilation increases from $1(2^{1-1})$ to $8(2^{4-1})$ exponentially, which leads to the exponential growth in receptive field size of every element in high-level feature maps. In each recursive stage, the outputs of each basic unit are added up to create new feature maps as the input of the next recursive stage. This design allows the low-level contextual information to be used directly together with the high-level contextual information, which is helpful to the next recursive stage.

Decoder D. While the recursive module is simple and powerful, we find training a deeply-recursive module is difficult due to the gradients vanishing and exploding. To solve the above issue, we feed outputs of all recursive stages to the reconstructor \mathbf{R} , and compute their weighted sum, as shown in Fig. 3(c). Through this design, those feature maps are simultaneously supervised during training.

In detail, \mathbf{D} is comprised of \mathbf{R} and two convolutions. \mathbf{R} is built on a basic unit, whose weights are shared for feature maps from each recursive stage. All outputs of \mathbf{R} , as shown in Fig. 3(c), are summed by optimal weights which are automatically learned during training. The next part of \mathbf{D} consists of a 3 × 3 convolution and a 1 × 1 convolution, which transforms feature maps to streak artifacts. Finally, the predicted steak artifacts are subtracted from the sparse-view CT image to create the prediction of the dense-view CT image.

2.3 Recurrent Model

As it is not easy to remove all streak artifacts in one stage, we incorporate the recurrent mechanism to remove streak artifacts in multiple stages. ConvGRU containing memory mechanism is selected to fully investigate the recurrent connections between different recurrent stages. The recurrent model is shown in Fig. 4. What's more, the feature map size is not changed in R²-Net's process.

In each recurrent stage, our proposed model predicts the whole residual, i.e., streak artifacts. Our scheme can be formulated as:

$$I_s^1 = I_{ori}, \ H^0 = Zeroes, \eqno(3)$$

$$\widehat{Res^i} = F_i(I_s^i, H^{i-1}), \ 1 \le i \le N, \tag{4}$$

$$I_s^{i+1} = \widehat{I}_d^i = I_{ori} - \widehat{Res^i}, \tag{5}$$

where I_{ori} indicates the original sparse-view CT image, I_s^i represents the input of the *i*-th recurrent stage, N is the number of recurrent stages, F_i indicates the computing process of the *i*-th recurrent stage, H^{i-1} represents the hidden states, i.e., the feature maps of the i-1-th stage, Res^i indicates the output of

the *i*-th recurrent stage, and \widehat{I}_d^i is the predicted dense-view CT image as well as intermediate artifacts-free image after the *i*-th recurrent stage.

The overall loss function is defined as the sum of all recurrent stages' loss, which is formulated as:

$$L(\Theta) = \sum_{i=1}^{N} \left\| \widehat{Res^i} - Res \right\|_F^2, \tag{6}$$

where Res is the residual between the original sparse-view CT image and dense-view CT image, and Θ represents the network's parameters.

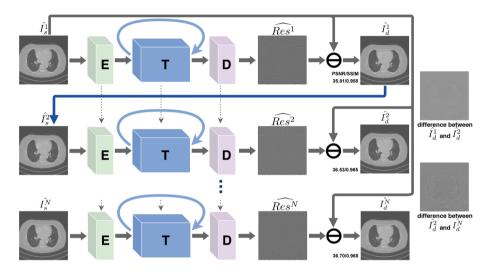


Fig. 4. The unrolling architecture of our proposed method. The two images on the far right show the differences of streak artifacts learned between different stages, which demonstrates that the recurrent mechanism can refine the outputs progressively. Best viewed on the screen.

3 Experiments

3.1 Dataset

We evaluate our proposed neural network architecture on the normal-dose part of the AAPM [13] dataset, which consists of 2,378 normal dose CT images from 10 patients and is the most commonly used benchmark dataset in the field. The 3D volumes contain from 128 to 343 slices per patient.

3.2 Experimental Setup

The performance is reported on 5-fold cross-validation. In each cross-validation loop, among the 10 patients, 8 patients' data are used for training and the other 2 patients' data are used for testing.

For the training set, we use the 2D FBP reconstruction images from 30, 60, 120, 180, and 240 projection views as input. The signals of 30, 60, 120, 180, and 240 projection views are generated through simulated downsampling in the transform domain, as the well-known Radon Transform algorithm. And the residual images, i.e., streak artifacts, are used as the label. The residual images are the difference between the dense view (720 views) reconstructions and the sparse view reconstructions. On account of fully utilizing the receptive field, we use the original 512×512 images without cropping as input. What's more, in order to avoid the influence of outliers, we normalize the dataset according to the upper and lower 25 points of all pixels' values.

All architectures are trained by Adam algorithm. As for learning rate, we choose the best decrease range and scheduler for each method, such as 10^{-3} to 10^{-5} for R²-Net, 5×10^{-4} to 10^{-5} for DD-Net [2] and 10^{-3} to 10^{-6} for TF U-Net [3]. The decrease range and scheduler are chosen based on experiments of that method. For evaluation metrics, we adopt SSIM and PSNR.

Table 1. Quantitative comparison (SSIM/PSNR(dB)) between R²-Net and other SOTA methods on the AAPM CT dataset.

SSIM/PSNR	30 views	60 views	120 views	180 views	240 views
FBP [4]	0.459/19.83	0.661/25.54	0.904/33.56	0.976/39.88	0.992/44.83
TF U-Net [3]	0.948/33.87	0.957/36.42	0.979/40.31	0.992/44.16	0.997/48.66
DD-Net [2]	0.945/33.74	0.963/37.13	0.983/40.54	0.993/43.66	0.995/47.11
R ² -Net	0.951/34.31	0.971/38.07	0.991/43.97	0.996/47.19	0.998/49.68

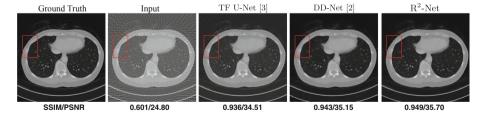


Fig. 5. Qualitative results of R²-Net and other SOTA methods on 60 views. Best viewed at the screen. More visualization results are in supplementary material.

3.3 Experimental Results

Comparison Between R²-Net and Other SOTA Models. In Table 1, we present the average SSIM and PSNR values of FBP, other SOTA methods and R²-Net. All methods are significantly better than FBP. Our R²-Net achieves much better results than other two SOTA methods. Qualitative results of 60 view are shown in Fig. 5. More visualization results of different views are presented in supplementary material.

As for the parameters cost, we calculate the number of parameters used in the three methods, which is illustrated in Table 2. R^2 -Net costs only 4.94% parameters of DD-Net and 0.05% parameters of TF U-Net. Small parameter number makes it easy to deploy and store.

What's more, we present the GPU memory cost of the inference stage in supplementary material.

Ablation Study About Recurrent and Recursive Mechanisms. In this part, we conduct experiments to compare the effect of different settings of recurrent and recursive numbers. This study may help us to understand how important roles are these two mechanisms playing in the architecture. In Table 3, we report the ablation study's results. In the table, M indicates the number of recursive stages, N indicates the number of recurrent stages. It's obvious that both recurrent and recursive mechanisms are beneficial to the performance. In detail, the recurrent mechanism contributes more than the recursive mechanism.

Table 2. Parameters cost comparison among TF U-Net, DD-Net and \mathbb{R}^2 -Net.

	Number of params (K)
TF U-Net [3]	40,837
DD-Net [2]	423
R ² -Net	21

Table 3. Quantitative measures comparison among different numbers of $\operatorname{recursive}(M)$ and $\operatorname{recurrent}(N)$ stages of \mathbb{R}^2 -Net of 120 views.

(M, N)	(1, 1)	(3, 1)	(1, 3)	(3, 3)
SSIM	0.9885	0.9889	0.9908	0.9912
PSNR	42.66	43.03	43.71	43.97

4 Conclusion

In this work, we propose a novel neural network architecture based on recurrent and recursive mechanisms. We incorporate the recurrent mechanism and SE block to progressively suppress streak artifacts stage by stage. What's more, in order to enlarge the receptive field and reduce the parameters number, we introduce the exponentially increased dilation and recursive mechanism into our architecture. Benefited from these two mechanisms, R²-Net outperforms SOTA methods of streak artifacts removal.

Acknowledgment. We thank Dr. Cynthia McCollough (the Mayo Clinic, USA) for providing CT data of Low Dose CT Grand Challenge for research purpose.

Zhouchen Lin is supported by National Basic Research Program of China (973 Program) (grant no. 2015CB352502), National Natural Science Foundation (NSF) of China (grant nos. 61625301 and 61731018), and Microsoft Research Asia.

References

- Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: IEEE conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
- 2. Zhang, Z., Liang, X., Dong, X., et al.: A sparse-view CT reconstruction method based on combination of DenseNet and deconvolution. IEEE Trans. Med. Imaging 37(6), 1407–1417 (2018)
- Han, Y., Ye, J.C.: Framing U-Net via deep convolutional framelets: application to sparse-view CT. IEEE Trans. Med. Imaging 37(6), 1418–1429 (2018)
- Kak, A.C., Slaney, M., Wang, G.: Principles of computerized tomographic imaging. Med. Phys. 29(1), 107 (2002)
- Sidky, E.Y., Pan, X.: Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization. Phys. Med. Biol. 53(17), 4777 (2008)
- Chen, G.H., Tang, J., Leng, S.: Prior image constrained compressed sensing (PICCS): a method to accurately reconstruct dynamic CT images from highly undersampled projection data sets. Med. Phys. 35(2), 660–663 (2008)
- Liu, Y., Ma, J., Fan, Y., et al.: Adaptive-weighted total variation minimization for sparse data toward low-dose x-ray computed tomography image reconstruction. Phys. Med. Biol. 57(23), 7923 (2012)
- Liu, Y., Liang, Z., Ma, J., et al.: Total variation-stokes strategy for sparse-view X-ray CT image reconstruction. IEEE Trans. Med. Imaging 33(3), 749–763 (2014)
- 9. Chen, Y., Shi, L., Feng, Q., et al.: Artifact suppressed dictionary learning for low-dose CT image processing. IEEE Trans. Med. Imaging 33(12), 2271–2292 (2014)
- Chen, Y., Yang, Z., Hu, Y., et al.: Thoracic low-dose CT image processing using an artifact suppressed large-scale nonlocal means. Phys. Med. Biol. 57(9), 2667 (2012)
- Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
- 12. Kofler, A., Haltmeier, M., Kolbitsch, C., Kachelrieß, M., Dewey, M.: A U-Nets cascade for sparse view computed tomography. In: Knoll, F., Maier, A., Rueckert, D. (eds.) MLMIR 2018. LNCS, vol. 11074, pp. 91–99. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00129-2_11
- 13. AAPM Low Dose CT Grand Challenge Homepage. https://www.aapm.org/grandchallenge/lowdosect/. Accessed 3 July 2019