A U-Nets Cascade for Sparse View Computed Tomography

Andreas Kofler[1(✉)], Markus Haltmeier[2], Christoph Kolbitsch[3,4], Marc Kachelrieß[5], and Marc Dewey[1]

[1] Department of Radiology, Charité - Universitätsmedizin Berlin, Berlin, Germany
andreas.kofler@charite.de
[2] Department of Mathematics, University of Innsbruck, Innsbruck, Austria
[3] Physikalisch-Technische Bundesanstalt, Braunschweig and Berlin, Germany
[4] Division of Imaging Sciences and Biomedical Engineering, King's College London, London, UK
[5] Medical Physics in Radiology, German Cancer Research Center, Heidelberg, Germany

Abstract. We propose a new convolutional neural network architecture for image reconstruction in sparse view computed tomography. The proposed network consists of a cascade of U-nets and data consistency layers. While the U-nets address the undersampling artifacts, the data consistency layers model the specific scanner geometry and make direct use of measured data. We train the network cascade end-to-end on sparse view cardiac CT images. The proposed network's performance is evaluated according to different quantitative measures and compared to the one of a cascade with fully convolutional neural networks with residual connections and to the one of a single U-net with approximately the same number of trainable parameters. While in both experiments the methods show similar performance in terms of quantitative measures, our proposed U-nets cascade yields superior visual results and better preserves the overall image structure as well as fine diagnostic details, e.g. the coronary arteries. The latter is also confirmed by a statistically significant increase of the Haar-wavelet-based perceptual similarity index measure in all the experiments.

Keywords: Deep learning · Convolutional neural networks
Data consistency · Computed tomography · Sparse sampling

1 Introduction

Sparse data-acquisition protocols are widely used in magnetic resonance imaging (MRI) in order to shorten scanning times. In contrast, in computed tomography (CT), the data acquisition process is fast while reducing radiation exposure is an important clinical issue. One possible way to reduce radiation exposure is to decrease the tube current of the X-ray emitting source. However, the direct consequence is decreased image quality due to higher image noise. In this paper,

F. Knoll et al. (Eds.): MLMIR 2018, LNCS 11074, pp. 91–99, 2018.
https://doi.org/10.1007/978-3-030-00129-2_11

we use a sparse view data-acquisition scheme to reach a significant radiation exposure reduction in CT. This can be achieved by masking the X-ray source at certain angular positions during the rotation of the scanner and therefore preventing some X-rays to pass through the patient. Using standard algorithms, images reconstructed from sparse view data exhibit undersampling structures which are related to the scanner geometry as well as the sub-sampling scheme used for data acquisition.

Recently, deep neural networks have shown to be a promising alternative to current state-of-the-art iterative methods for the reconstruction from heavily undersampled CT data. In particular, the U-net [6] has shown its excellent performance in the restoration of undersampled images in CT and MRI [4]. However, these standard network designs can be viewed as post-processing methods, as the network used to remove the artifacts is the only learned component in the reconstruction pipeline. As a consequence, these methods may lack data consistency. In this paper we propose a new network architecture for the image reconstruction from undersampled data in sparse view CT. Our network structure is inspired by the network cascade developed in [7] and consists of a cascade of convolutional neural networks and data consistency layers which minimize a properly-chosen functional. However, while the approach in [7] is based on the isometry of the full MRI forward operator, our data consistency layer is directly applicable to general inverse problems as well. Furthermore, the fully convolutional neural networks (FCNNs) with residual connections are replaced by U-nets. For different, gradient-descent-like data consistency layers, see [2,3].

1.1 Sparse View Computed Tomography

Here and after we work with the discrete setting. By $\mathbf{x} \in \mathbb{R}^n$ we refer to the vector of size $m \times m$ with $m^2 = n$ as representation of the two-dimensional X-ray attenuation function and write $\mathbf{y} \in \mathbb{R}^d$ for a fully sampled sinogram. Further, we use $\mathbf{R}$ to denote the discretized forward operator of a CT scanner, i.e. the discrete X-ray transform specified by the scanner's geometry. We denote the pseudoinverse of the discretized forward operator by $\mathbf{R}^\dagger$. Note that the continuous form of the Radon transform is injective but not surjective. Therefore, we may assume that the Radon transform $\mathbf{R}$ is sampled sufficiently fine such that the discretized full data operator is injective but not surjective as well. Anyway, the approach presented below works for an arbitrary discrete transform $\mathbf{R} \in \mathbb{R}^{d \times n}$.

Assume the data is measured only for lines corresponding to a subset $I \subset J \triangleq \{1, \dots, d\}$, where J is the full set of projections. The corresponding discretized sparse data forward operator can be modeled by $\mathbf{R}_I = \mathbf{S}_I \mathbf{R}$, where the subsampling operator is given by

$$\mathbf{S}_I \mathbf{y}(i) \triangleq \begin{cases} \mathbf{y}(i) & \text{if } i \in I \\ 0 & \text{if } i \in I^c := J \setminus I. \end{cases} \tag{1}$$

The sparse data image reconstruction problem then consists in recovering the image $\mathbf{x} \in \mathbb{R}^n$ from the set of projections, i.e. we want to solve

$$\mathbf{R}_I \mathbf{x} = \mathbf{y}_I. \tag{2}$$

2 Proposed Network Architecture

In the full data case, (2) can be be solved by filtered back-projection, which is a stable numerical implementation of $\mathbf{R}^\dagger$. However, in the sparse view case we have $|I| \ll |I^c|$ and the application of $\mathbf{R}^\dagger$ to data $\mathbf{y}_I$ yields images with severe artifacts. Images with diagnostic quality can usually be obtained by iterative reconstruction methods designed for minimizing $\mathcal{R}(\mathbf{x}) + \lambda \|\mathbf{R}_I \mathbf{x} - \mathbf{y}_I\|_p^p$, where $\mathcal{R}(\mathbf{x})$ is a regularizer and $\|\cdot\|_p$ denotes a norm which ensures data consistency. Typical choices for the regularizer are the total variation, or the ℓ^1-norm with respect to a frame or a trained dictionary. As a drawback, these methods are usually computationally expensive since they rely on a repeated application of the forward and adjoint operators. Furthermore, using regularization solely based on prior assumptions will likely bias the result.

Methods based on neural networks as for example in [4] propose non-iterative regularization approaches. Given an estimate solution $\mathbf{x}_I$ of (2), regularized images are obtained as the output of a CNN f which is previously trained on a dataset of pairs $(\mathbf{x}_I, \mathbf{x}_{\text{full}})$, where $\mathbf{x}_{\text{full}}$ is an image obtained from the reconstruction of a fully-sampled measurement. Such a procedure consists in a subsequent regularization of the initial solution $\mathbf{x}_I$ rather than a joint minimization of $\mathcal{R}(\mathbf{x}) + \lambda \|\mathbf{R}_I \mathbf{x} - \mathbf{y}_I\|_p^p$. Therefore, following [7], we propose to train different networks intercepted by data consistency (DC) layers.

2.1 Data Consistency Layer

Let f_Θ be a previously trained CNN with parameters Θ. Given measured data $\mathbf{y}_I$, we can apply a CNN to map $\mathbf{x}_I$ to its corresponding label, i.e. $f_\Theta(\mathbf{x}_I) \simeq \mathbf{x}_{\text{full}}$ where $\mathbf{x}_I \triangleq \mathbf{R}^\dagger \mathbf{y}_I$. However, the CNN reconstruction $f_\Theta(\mathbf{x}_I)$ may not satisfy the data consistency condition $\mathbf{R}_I(f_\Theta(\mathbf{x}_I)) \simeq \mathbf{y}_I$.

In order to improve data consistency, we define a new reconstruction $f_{\text{dc}}(\mathbf{x}_{\text{cnn}}, \mathbf{y}_I, \lambda) \triangleq \mathbf{R}^\dagger(\mathbf{z}_{\text{dc}})$ where $\mathbf{z}_{\text{dc}} \in \mathbb{R}^d$ is the minimizer of the functional given by

$$F_{\Theta, \mathbf{y}_I, \mathbf{x}_{\text{cnn}}, \lambda}(\mathbf{z}) \triangleq \|\mathbf{R}(\mathbf{x}_{\text{cnn}}) - \mathbf{z}\|_2^2 + \lambda \|\mathbf{y}_I - \mathbf{S}_I \mathbf{z}\|_2^2, \tag{3}$$

with $\mathbf{x}_{\text{cnn}} = f_\Theta(\mathbf{x}_I)$ denoting the output of the trained CNN.

Here, the term $\|\mathbf{y}_I - \mathbf{S}_I \mathbf{z}\|_2^2$ enforces data consistency and $\|\mathbf{R}(\mathbf{x}_{\text{cnn}}) - \mathbf{z}\|_2^2$ uses $\mathbf{x}_{\text{cnn}}$ to regularize in Radon space. Opposed to [7], where the regularization term $\|\mathbf{x}_{\text{cnn}} - \mathbf{x}\|_2^2$ in image space has been used, the proposed regularization in data space yields the following representation of the DC layer for general, possibly non-orthogonal transforms.

Theorem 1. *Let* $\mathbf{R} \in \mathbb{R}^{d \times n}$ *be a real valued matrix and* $\mathbf{R}_I = \mathbf{S}_I \mathbf{R}$*, where* $\mathbf{S}_I$ *is the subsampling operator defined in (1). The data consistency layer* $f_{\mathrm{dc}}(\mathbf{x}_{\mathrm{cnn}}, \mathbf{y}_I, \lambda)$ *is well defined by (3) and takes the explicit form*

$$f_{\mathrm{dc}}(\mathbf{x}_{\mathrm{cnn}}, \mathbf{y}_I, \lambda) = \mathbf{R}^{\dagger}\big(\mathbf{A}\mathbf{R}\mathbf{x}_{\mathrm{cnn}} + \frac{\lambda}{1+\lambda}\mathbf{y}_I\big), \tag{4}$$

where $\mathbf{A} = \mathrm{diag}(a_1, \ldots, a_n)$ *is a diagonal matrix of size* $d \times d$ *with diagonal entries* $a_i = 1$ *if* $i \notin I$ *and* $a_i = 1/(1+\lambda)$ *otherwise.*

Proof. The functional in (3) takes the separable form $\sum_{i \in J} |\mathbf{R}\mathbf{x}_{\mathrm{cnn}}(i) - \mathbf{z}(i)|_2^2 + \lambda |\mathbf{y}_I(i) - (\mathbf{S}_I \mathbf{z})(i)|_2^2$. Hence, the minimizer of $F_{\Theta, \mathbf{y}_I, \mathbf{x}_{\mathrm{cnn}}, \lambda}$ is unique and can be found by component-wise minimization. Elementary computations show (4).

The matrix $\mathbf{A}$ ensures that, when the i-th projection is not available from the measurements, $(\mathbf{R}\mathbf{x})(i)$ is directly estimated from the projection data of the output of the CNN. Otherwise, $(\mathbf{R}\mathbf{x})(i)$ is calculated as a linear combination of the CNN coefficient $\mathbf{R}\mathbf{x}_{\mathrm{cnn}}(i)$ and the measured coefficient $\mathbf{y}_I(i)$. Note that the evaluation of (4) requires the application of the pseudoinverse, which might be numerically unstable. In the numerical implementation, the pseudoinverse $\mathbf{R}^{\dagger}$ is replaced by an appropriate regularization. We emphasize that this issue is not present in MRI reconstruction, as the corresponding full data operator is bijective and the inverse well-conditioned. Therefore, the extension of the corresponding data consistency layer from MRI to CT is a non-trivial issue.

2.2 U-Nets Cascade

Here, we always refer to a U-net as any residual encoder-decoder network architecture with a similar structure to the one presented in [4]. However, in our experiments we vary the number of stages which are used to encode the input, the number of convolutional layers per stage, the initial number of feature maps which are extracted from the input and the factor by which the feature maps are augmented after each max-pooling layer. In order to satisfy the data consistency condition $\mathbf{R}_I(f_\Theta(\mathbf{x}_I)) \simeq \mathbf{y}_I$, we propose to construct a sequence of U-nets which are intercepted by DC layers as described in Subsect. 2.1. While the U-nets tackle the removal of the undersampling artifacts, the DC layers account for data consistency in Radon space. Figure 1 shows the structure of a U-nets cascade, where each U-net consists of three encoding stages and two convolutional layers per stage.

3 Numerical Experiments

3.1 Dataset

We test our proposed network architecture on a dataset consisting of cardiac CT images from 52 patients. The $3D$ volumes contain from 240 up to 640 slices per patient. For each slice, the undersampled data $\mathbf{y}_I$ is generated according to a

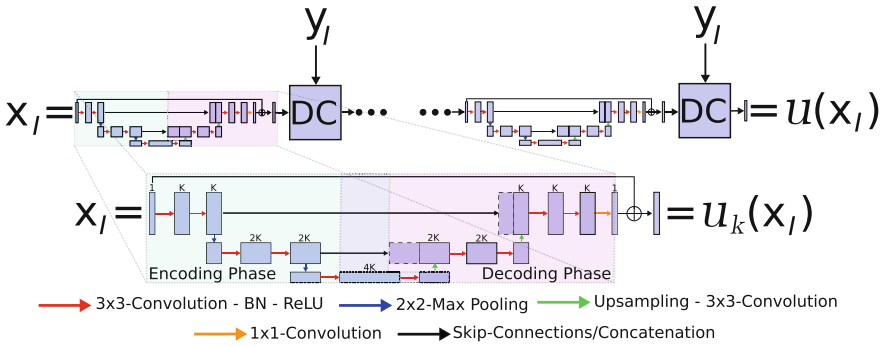

Fig. 1. A cascade of U-nets with intermediate data consistency layers.

parallel-beam geometry where we cover a half rotation of 180° of the scanner by only 32 angles. The images $\mathbf{x}_I$ are obtained by applying filtered back-projection $\mathbf{R}^\dagger$ with Ram-Lak filter to $\mathbf{y}_I$. The operator $\mathbf{R}$ is assumed to perform 512 projections. We use the images of 40 patients for training, of 6 for validation and of 6 for testing. For computational reasons and in order to allow us to build neural networks with a certain depth, the images are first downsampled from 512×512 to 256×256 pixels.

3.2 Network Architectures and Training

In all our experiments we train the U-nets cascade to minimize the L_2-error between the predicted output of the cascade and the corresponding label. All architectures are trained for 20 epochs by stochastic gradient descent. When one single U-net is used, we decrease the learning rate from 10^{-7} to 10^{-9}. For all other architectures which contain the operators $\mathbf{R}$ and $\mathbf{R}^\dagger$, a more conservative learning rate which is decreased from 10^{-10} to 10^{-14} has to be chosen for numerical stability. The network architectures are implemented in `TensorFlow` and the scanner geometry, the forward and the pseudoinverse operators $\mathbf{R}$ and $\mathbf{R}^\dagger$ are implemented in `ODL` [1]. We parametrize a U-net cascade according to the following hyperparameters:

- U - the number of U-nets employed in the cascade
- E - the number of stages used for the encoding of each U-net
- C - the number of convolutional layers per stage for each U-net
- K - the number of feature maps which are initially extracted from the input of each U-net
- F - the factor by which the number of feature maps is increased after the max-pooling layers of each U-net.

For example, U1 E5 C4 K64 F2 denotes a single U-net architecture similar to the one presented in [4]. On the other hand, U4 E1 C4 K64 denotes a FCNN cascade as discussed in [7]. Note that, in such a case, we omit the hyperparameter F

in the notation, since due to the absence of max-pooling layers, the number of extracted feature maps stays constant over the different stages.

For a fair comparison, we try to keep the number of trainable parameters approximately equal for the architectures we compare. Note that due to the large number of possible combinations of hyperparameters, it is computationally demanding to conduct experiments which clearly reveal the effect of each hyperparamter. However, we identify the presence of max-pooling layers to be the main difference between the proposed U-net cascade and the cascade in [7] in terms of feature-extraction-operations of the subnetworks. Therefore, in order to reach a certain number of trainable parameters, we choose to always favour to increase the number of encoding stages rather than increasing the number of convolutional layers per stage, the number of extracted feature maps or the factor by which they are increased after the max-pooling layers.

For the evaluation of the performance of the network we report the peak signal-to-noise ratio (PSNR), the relative L_2-error (NRMSE), the structural similarity index measure (SSIM) and the Haar-wavelet based perceptual similarity index measure (HPSI, [5]) which has been reported to achieve higher correlation with human opinion scores than SSIM on various benchmark databases.

Effect of the U-Net: Here, we investigate the effect of the replacement of the FCNNs discussed in [7] by the U-nets. Table 1 lists the average of the aforementioned quantitative measures over the test set. In terms of PSNR, SSIM and NRMSE, both cascades deliver similar results. On the other hand, we report a statistically significant increase of the mean value of HPSI for all tested U-nets cascades, ($p < 0.001$ for all cases). Figure 2 shows two examples of reconstructed images of the test set. Due the relatively small number of trainable parameters and the high undersampling factor, both approaches do not entirely remove the

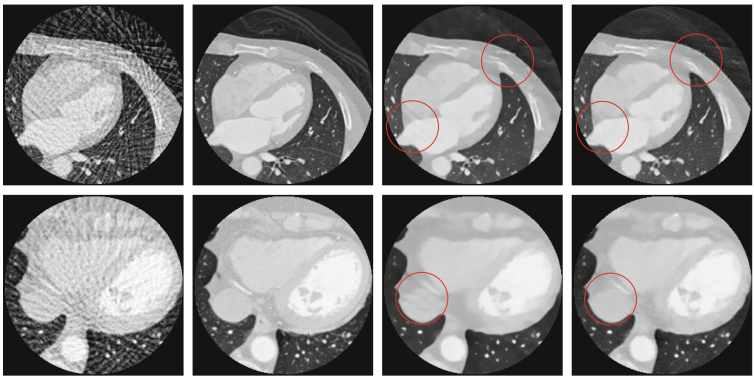

Fig. 2. Comparison of different cascades. 32-views FBP-reconstruction (first column), ground truth (second column), U4 E1 C4 K64 (third column), U4 E4 C2 K32 (fourth column). The red circles indicate newly introduced or not correctly removed artifacts from the reconstruction with the FCNNs-cascade. (Color figure online)

undersampling artifacts and fail at recovering fine details. Note that, however, the cascade with the FCNNs even introduces new artifacts. The phenomenon can be observed in several images reconstructed with the FCNNs cascade. On the other hand, the U-nets cascade seems to better preserve the overall structure of the images.

Table 1. Comparison of the proposed U-nets cascade with a cascade of FCNNs with residual connections. The measures are averaged over the test set.

Model	n_{params}	PSNR	SSIM	HPSI	NRMSE
U2 E1 C4 K64	371 459	30.63	0.8961	0.7236	0.1597
U2 E4 C2 K32	352 899	30.56	0.8939	0.7433	0.1612
U3 E1 C4 K64	557 187	30.26	0.8737	0.7311	0.1692
U3 E4 C2 K32	529 347	30.33	0.8744	0.7499	0.1679
U4 E1 C4 K64	742 915	29.89	0.8581	0.7326	0.1799
U4 E4 C2 K32	705 795	29.92	0.8603	0.7540	0.1782

Effect of the Cascade: In this experiment, we test different network architectures where we vary the length of the cascade. Figure 3 shows an image reconstructed with different network cascades. The results show that the left coronary artery is better visible in the images reconstructed with the U-nets cascades compared to a single U-net. In contrast to the results presented in [7], increasing the

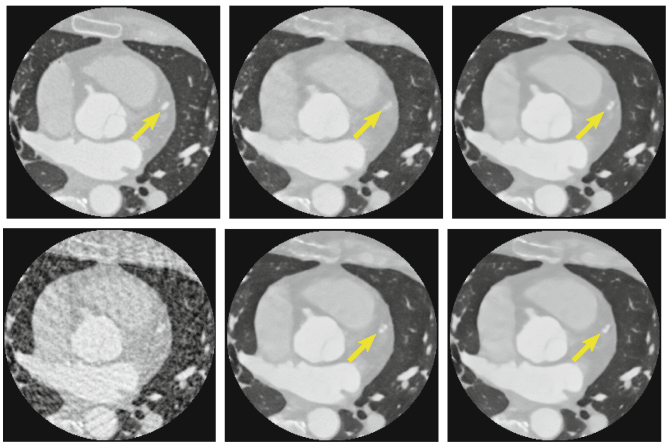

Fig. 3. Variation of the length of the cascade. Ground truth (top left), FBP-reconstruction from undersampled data (bottom left), U1 E3 C2 K64 F2-reconstruction (top middle), U2 E3 C4 K32 F2-reconstruction (bottom middle), U3 E3 C3 K64-reconstruction (top right), U4 E3 C2 K32 F2-reconstruction (bottom right). The yellow arrows point at the left coronary artery. (Color figure online)

Table 2. Variation of the length of the U-nets cascade. The measures are averaged over the test set.

Model	n_{params}	PSNR	SSIM	HPSI	NRMSE
U1 E3 C2 K64 F2	1 957 251	30.19	0.9532	0.7304	0.1832
U2 E3 C4 K32 F2	1 941 379	31.14	0.8905	0.7659	0.1531
U3 E3 C3 K64	1 999 107	30.85	0.8686	0.7732	0.1621
U4 E3 C2 K32 F2	1 960 707	30.38	0.8559	0.7729	0.1732

length of the cascades does not further improve the results. We attribute this to the fact that the inversion of the Radon-transform is ill-posed and therefore, numerical errors due to the inversion of $\mathbf{R}$ prevail over the presence of the data consistency layers. However, when we replace a single U-net by a U-nets cascade, the network's performance statistically significantly increases ($p < 0.001$) with respect to all measures except for SSIM, where a single U-net yields the best results, see Table 2.

3.3 Conclusion

In this work, we have presented a new network architecture for image reconstruction in sparse view CT. Replacing the FCNNs by U-nets in the cascade in [7] visually improves the reconstruction in sparse view CT. The proposed U-nets cascade outperforms the single U-net architecture with respect to all reported quantitative measures except for SSIM and better preserves fine anatomic details. By adapting the data-acquisition process and the index set I, the architecture is directly applicable to other limited data inverse problems such as limited angle CT where we expect the method to deliver even better results as the portion of measured data which can be used in the reconstruction is significantly larger. Furthermore, we expect the extension of the network cascade employing U-nets as sub-networks also to further improve the image reconstruction in MRI.

Acknowledgements. The authors would like to thank the reviewers for the helpful feedback. A. Kofler acknowledges support of the German Research Foundation (DFG), project number GRK 2260, BIOQIC. M. Haltmeier acknowledges support of the Austrian Science Fund (FWF), project P 30747-N32.

References

1. Jonas, A.: ODL - operator discretization library (2013). https://github.com/odlgroup/odl
2. Adler, J., Öktem, O.: Solving ill-posed inverse problems using iterative deep neural networks. Inverse Prob. **33**(12), 124007 (2017)
3. Hammernik, K., et al.: Learning a variational network for reconstruction of accelerated MRI data. Magn. Reson. Med. **79**(6), 3055–3071 (2018)

4. Jin, K.H., McCann, M.T., Froustey, E., Unser, M.: Deep convolutional neural network for inverse problems in imaging. IEEE Trans. Image Process. **26**(9), 4509–4522 (2017)
5. Reisenhofer, R., Bosse, S., Kutyniok, G., Wiegand, T.: A Haar wavelet-based perceptual similarity index for image quality assessment. Signal Process. Image Commun. **61**, 33–43 (2018)
6. 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
7. Schlemper, J., Caballero, J., Hajnal, J.V., Price, A., Rueckert, D.: A deep cascade of convolutional neural networks for MR image reconstruction. In: Niethammer, M. (ed.) IPMI 2017. LNCS, vol. 10265, pp. 647–658. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59050-9_51