

# A hybrid Neural Network combining explicit priors for Low-dose CT reconstruction

Xiangli Jin<sup>a</sup>, Yinghui Zhang<sup>a</sup>, Ran An<sup>a</sup>, and Hongwei Li<sup>a</sup>

<sup>a</sup> School of Mathematical Sciences, Capital Normal University, Beijing, 100048, China

## ABSTRACT

Low-dose CT reconstructions suffer from severe noise and artifacts. Many methods have been proposed to increase the ratio between image quality and radiation dose through incorporating various priors. By learning priors in labelled data-set, neural network methods have achieved great success for this purpose. In CT applications, however, paired training data-sets are rarely available or difficult to obtain. Recently, unsupervised learning has attracted a lot of attention. Along this line, Noise2Inverse, an unsupervised neural network architecture, has shown the possibility of applying unsupervised learning for low-dose CT reconstructions. When the training data-sets (unlabelled) are not large enough or the training is insufficient, however, Noise2Inverse might perform not well. Another important issue is that network methods might suffer from intrinsic instability. In this regard, we propose to hybrid neural networks, especially the Noise2Inverse architecture, with traditional optimization models such that hand-crafted priors come into play as a remedy. Numerical experiments show that the proposed architecture improves Noise2Inverse in terms of both quality measures PSNR and SSIM, especially in the case of inadequate training.

**Keywords:** Low-dose CT, deep learning, image denoising, TV regularization, primal-dual

## 1. INTRODUCTION

To avoid exposing patients to more X-ray radiation, it makes sense to reduce the X-ray dose. However, compared with conventional CT, low-dose CT usually introduce serious noise and artifacts. In recent years, low-dose CT (LDCT) reconstruction has been a major challenge in medical CT applications. In general, LDCT reconstruction methods can be roughly divided into three categories. The first category is the filtering method, which directly performs filtering and smoothing on the projection data or the noisy reconstructions. Popular methods include NLM<sup>1</sup> and BM3D,<sup>2</sup> which explore the structural similarity prior within a single image. The defect of these methods is that it can not distinguish well image structures and artifacts. The second category is the model-based optimization approach. By combining image priors into the objective function, noise and artifacts are removed in the reconstruction process. However, hand-crafted priors are often not accurate enough and might introduce negative effects like blurring. In addition, the optimization model usually need iterative solvers which are very time-consuming. The third category is deep learning methods like U-Net<sup>3</sup> and DnCNN.<sup>4</sup> Based on the powerful fitting ability of neural networks, priors or patterns concealed in ‘big data’ could be extracted and utilized. Neural networks have demonstrated to be superior to traditional methods if adequate labelled training data are available. However, for CT applications, high quality training data are usually difficult to acquire.

Unsupervised learning methods for image denoising gain popularity in recent years. The X2Y series algorithms have attracted much attention. The Noise2noise<sup>5</sup> method explains the MSE loss in Bayesian framework by which noisy images are allowed to be used as reference images, when certain independence conditions are satisfied. Later on, by introducing the idea of blind spot, Noise2Self<sup>6</sup> and Noise2Void<sup>7</sup> were proposed which ruled out the need for independent multiple snapshots of the same scene. It has been demonstrated that these unsupervised methods could achieve competitive denoising results compared to the supervised ones.

Based on the idea of Noise2Void, Noise2Inverse was proposed in<sup>8</sup> for CT image reconstruction. Basically, it consists of two procedures. The first one is to prepare the training data-sets. The projection data are divided

---

Further author information: (Send correspondence to Hongwei Li)  
Hongwei Li: E-mail: hongwei.li91@cnu.edu.cn

into non-overlapping groups which are then used to reconstruct noisy images of the same ‘scene’ (object). Since the projections are independent of each other, the noise in these images should follow the same distribution and be independent of each other. The second procedure is to train a denoising convolutional neural network (CNN) with the constructed training data in the first procedure. Noise2Inverse has shown promising results in LDCT denoising.

In practice, the training data sets might be quite limited available, such that under-fitting or over-fitting occurs. Another important issue is the stability of neural networks. It has been demonstrated that neural networks could suffer from intrinsic instability issue.<sup>9</sup> In our experiments, when the training data sets were relatively small or insufficient training were performed, i.e. the number of training epochs was set to be relatively small, the effectiveness of Noise2Inverse would be compromised.

To further improve the quality of LDCT reconstructions, we propose to combine the convolutional neural networks (CNN) with traditional optimization models. So, the proposed hybrid neural network (Hybrid NN) architecture consists of two blocks: the denoising CNN block and a training-parameter free optimization-based denoising block. Especially, in this paper, the second block is mapped from the primal-dual<sup>10</sup> algorithm for total-variation (TV) denoising, which encodes the piecewise-constant prior of the ideal image. Since the hybrid neural network builds in hand-crafted priors, its stability should have been improved.

The main contributions of this paper are two-fold.

- A hybrid neural network that blends neural networks and traditional optimization models is proposed. The hybrid model could leverage both the advantages of the prior concealed in data (through neural network learning) and hand-crafted priors (through explicit regularization).
- A special blending mechanism is devised which allows manually adjusting the “weights” of the neural network block and the optimization block, such that the designed architecture could better fit the size of the training datasets.

The remainder of this paper is organized as follows. The hybrid neural network is introduced and described in Sec. 2. In Sec. 3, experiments are carried out to validate the proposed hybrid neural network. We conclude the paper in Sec. 4.

## 2. METHOD

As stated before, the proposed hybrid NN consists of two blocks: CNN denoiser and optimization-based denoiser. To leverage the power of unsupervised learning, we borrow the idea of Noise2Inverse to prepare the training datasets. This is illustrated in Fig. 1. The backbone of the proposed architecture, i.e. the hybrid NN, is illustrated in Fig. 2

### 2.1 Noise2Inverse for LDCT

The key idea of the Noise2Inverse method is that it partitions the projection data into non-overlapping groups of equal size, each of them are then reconstructed into noisy images. Since there are no explicit correlations between the projection data groups, the reconstructed images could be thought of being independent observations, so the assumption for unsupervised training is satisfied. With training data-sets prepared, conventional convolutional neural networks like U-Net<sup>3</sup> and DCNN<sup>9</sup> could be utilized to achieve denoising.

### 2.2 LDCT noise simulation

The noise is mainly divided into two parts: statistical photon noise and electronic noise, and the Poisson-Gaussian mixture model is generally used to model the noise distribution:

$$N \sim \text{Poisson}(N_0 \exp(-y)) + \text{Gaussian}(0, \sigma_e^2)$$

Here,  $N_0$  means the photon number,  $y$  means the project data,  $\sigma_e^2$  means the variance of electronic noise;  $\text{Poisson}(\ast)$  means the *Poisson* distribution,  $\text{Gaussian}(\ast)$  means the *Gaussian* distribution.

### 2.3 Traditional optimization model

Let  $o$  denote the output of the first block. In the forward pass of the proposed hybrid NN,  $o$  is fed to the second block, i.e. an optimization model for further processing. Let  $A$  denote the desired output of the optimization model, then by adding TV regularizer and non-negative constraint, the optimization model can be written as

$$A = \arg \min_{A \geq 0} \left\{ \frac{1}{2} \|o - A\|_2^2 + \lambda \|\nabla A\|_1 \right\}. \quad (1)$$

A popular algorithm for solving the above model is the primal-dual Chambolle-Pock method. Given  $A^0 = 0$ ,  $\xi^t = 0$ :

$$\begin{cases} \xi^{t+1} = \xi^t + \tau \lambda \nabla A^t \\ \eta^{t+1} = P_B(\xi^{t+1}) \\ A^{t+1} = \frac{A^t + \tau(o + \lambda \operatorname{div}(\eta^{t+1}))}{1 + \tau} \\ A^{t+1} = \max(0, A^{t+1}) \end{cases}$$

where  $t$  is the iteration number,  $\tau$  is the time step, and  $\lambda$  is the regularization parameter which controls the denoising strength.  $P_B$  is an element-wise projection operator onto  $l_2$  Ball:

$$P_B(y) = \begin{cases} y, & \text{if } \|y\|_2 \leq 1 \\ \frac{y}{\|y\|_2}, & \text{if } \|y\|_2 > 1 \end{cases}.$$

When performing backpropagation, one needs to calculate the gradients regarding to the variables  $A^t$ . Deep learning frameworks like Pytorch provide tools for automatic gradients calculation. However, in our tests, automatic calculations were rather slow. So, we use the python package `cupy` to wrap up cuda kernels to serve our need. The required gradients can be derived as below.

Let  $o_{t+1} = o_t = \dots o_1$ , then we have

$$\begin{cases} dA_{t+1} = \mathcal{R} \left( \frac{1}{1 + \tau} + \frac{\tau^2 \lambda^2}{1 + \tau} \operatorname{div} \mathcal{M}_{t+1} \nabla \right) dA_t \\ \quad + \mathcal{R} \frac{\tau \lambda}{1 + \tau} \operatorname{div} \mathcal{M}_{t+1} d\xi_t + \mathcal{R} \frac{\tau}{1 + \tau} do_t \\ d\xi_{t+1} = d\xi_t + \tau \lambda \nabla dA_t \\ do_{t+1} = do_t \end{cases} \quad (2)$$

where,  $\mathcal{R}$  denotes the derivative for the non-negative constraint,  $\operatorname{div}$  is the divergence operation, and  $\mathcal{M}_{t+1}$  represents the derivative of the projection operator  $P_B$  in the  $(t + 1)$ th iteration. The matrix form reads

$$\begin{pmatrix} dA_{t+1} \\ d\xi_{t+1} \\ do_{t+1} \end{pmatrix} = V_{t+1} \begin{pmatrix} dA_t \\ d\xi_t \\ do_t \end{pmatrix} \quad (3)$$

with

$$V_{t+1} = \mathcal{N} \begin{pmatrix} \frac{1}{1 + \tau} + \frac{\tau^2 \lambda^2}{1 + \tau} \operatorname{div} \mathcal{M}_{t+1} \nabla & \frac{\tau \lambda}{1 + \tau} \operatorname{div} \mathcal{M}_{t+1} & \frac{\tau}{1 + \tau} \\ \tau \lambda \nabla & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where

$$\mathcal{N} = \begin{pmatrix} \mathcal{R} & & \\ & 1 & \\ & & 1 \end{pmatrix}.$$

Let  $L$  denote the loss function, then

$$\begin{aligned}
 dL &= \text{Tr} \left[ \left( \frac{\partial L}{\partial o} \right)^T do \right] = \text{Tr} \left[ \left( \frac{\partial L}{\partial A_{t+1}} \right)^T dA_{t+1} \right] \\
 &= \text{Tr} \left[ \begin{pmatrix} \frac{\partial L}{\partial A_{t+1}} \\ 0 \\ 0 \end{pmatrix}^T V_{t+1} \begin{pmatrix} dA_t \\ d\xi_t \\ do_t \end{pmatrix} \right] \\
 &= \text{Tr} \left[ \begin{pmatrix} \frac{\partial L}{\partial A_{t+1}} \\ 0 \\ 0 \end{pmatrix}^T V_{t+1} \cdot V_t \cdots V_2 \begin{pmatrix} \mathcal{R} \frac{\tau}{1+\tau} \\ 0 \\ 1 \end{pmatrix} do \right]
 \end{aligned}$$

where,  $\text{Tr}(Q)$  computes the trace of matrix  $Q$ . Finally, the required gradient can be extracted as

$$\frac{\partial L}{\partial o} = \begin{pmatrix} \mathcal{R} \frac{\tau}{1+\tau} \\ 0 \\ 1 \end{pmatrix}^T V_2^T \cdots V_t^T \cdot V_{t+1}^T \begin{pmatrix} \frac{\partial L}{\partial A_{t+1}} \\ 0 \\ 0 \end{pmatrix}.$$

Please note that, regularized CNN<sup>10</sup> is a semantic segmentation NN that emphasizes the effectiveness of explicitly adding TV regularization to the activation functions. The proposed hybrid NN is a fusion of the functionality of CNN and the optimization model, so, it could not only be employed for denoising, but also for other image processing applications. In fact, the neural network implicitly encodes priors into the network output as an initial value for the traditional optimization model layer to iterate to the desired output by incorporating the explicit priors, thus realizing the deep integration of data-driven and model-driven. The flow chart of proposed method is shown in Fig.1. Furthermore, traditional optimization model could be added to anywhere in the CNN networks, whenever the results of NN fail to meet expectations.

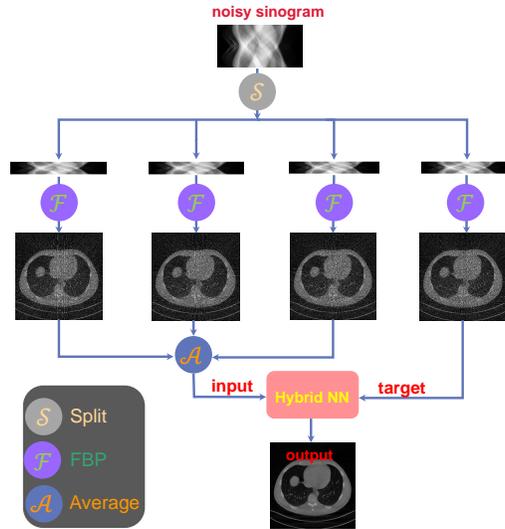


Figure 1. The flow chart of hybrid NN for LDCT. The input noisy sinogram is split into four groups. Then, FBP is applied to reconstruct 4 noisy images from which the training data-sets are constructed.

## 2.4 Loss function

Clearly, there exists an interaction between the two designed blocks, and the hyper-parameters, e.g. the number of primal-dual iterations, the depth of the CNN block, the weighting parameter  $\lambda$ , etc. would affect the networks' performance. Particularly, the two blocks actually run for the same goal, i.e. act as denoisers, the good

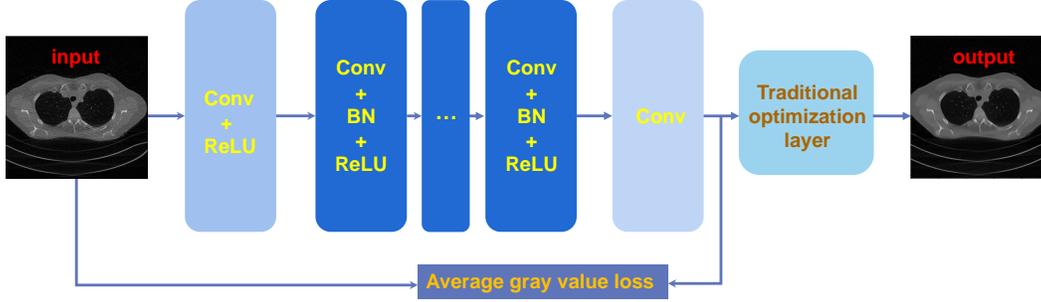


Figure 2. Illustration of hybrid NN.

performance of the second block, i.e. the pridual-dual algorithm, might inhibit the performance of the CNN block. To combat such a negative effect, we add a skip connection from the CNN block to the final output of the network, which requires that the CNN block should preserve the average gray values of its input images. This is achieved by adding an additional term to the loss function:

$$loss_{agv} = \|\bar{G}_{input} - \bar{G}_{CNNoutput}\|^2$$

where  $\bar{G}_{(I)}$  means the average gray value of image  $I$ . The final loss function is

$$loss_{total} = MSE(target, output) + \alpha \cdot loss_{agv},$$

where MSE means the mean squared error loss, and  $\alpha$  is a positive scalar parameter.

### 3. EXPERIMENTS AND RESULTS

#### 3.1 Datasets and settings

To verify the effectiveness of the proposed hybrid NN, a real clinical dataset was used, which was for "LUNA (LUng Nodule Analysis) 16–ISBI 2016 Challenge" (<https://luna16.grand-challenge.org/download/>). The dataset contains 1308 thoracic volumes from 1010 people, including 244,527 image slices size of  $512 \times 512$ . We randomly selected 7 patients, 2270 slices, from which 200 slices were randomly chosen as the training dataset and 10 slices as the testing dataset. The projection data are acquired with a virtual CT system equip with a parallel beam source and a linear detector consisting of 720 cells. The projection data are acquired for 360 projection views uniformly distributed in the angular range  $[0, \pi]$ . To simulate low-dose radiation, Poisson noise with incident intensity  $I_0$  is added to the raw data as follows

$$p_{noisy} = -\ln\left(\frac{I_0 \times e^{-p}}{I_0}\right), \quad (4)$$

where  $p$  and  $p_{noisy}$  denote the noise-free and noisy sinogram data, respectively. In our tests, we set  $I_0 = 3 \times 10^4$ . As Fig.1 shows, we split the sinogram into four non-overlapping groups in the way that each group consists of projection angles uniformly distributed in  $[0, \pi]$ .

The proposed hybrid NN is implemented with the PyTorch framework. All the experiments are executed on a single graphic processing NVIDIA card RTX 2080Ti with 11GB video memory. The gradient  $\frac{\partial L}{\partial o}$  is computed by cuda kernels wrapped by cupy <https://github.com/cupy/cupy>.

#### 3.2 Hyperparameters selection

For the proposed hybrid NN, the hyperparameters include the number of layers of the CNN, the number of primal-dual iterations, the scalar parameters  $\tau$ ,  $\lambda$  for the optimization layer and  $\alpha$  for the loss function. Considering the convergence requirement of the primal-dual iterations, We set  $\tau = \frac{1}{2\lambda}$ . In our tests, the CNN has 12 layers, and  $\alpha$  is set in a trial and error manner.

### 3.3 The effect of the explicit prior on the performance of CNN

To verify whether the second block, i.e. the explicit regularization has an influence on the performance of the CNN block, the following experiments were carried out: training the hybrid NN with increasing number of the primal-dual iterations  $t = 10, 30$  and  $50$ , then testing the first block, i.e. CNN along with the learned weights. The results shown in Fig.3 indicate that, with increasing  $t$ , the performance of the CNN block becomes poorer and poorer. This coincides with our intuition. By increasing  $t$ , the performance of the optimization model (second block) is improved, so the importance of the CNN block gets weakened.

As Section 2.4 has stated, to preserve the performance of the CNN block, the contrast-preserving loss  $loss_{agv}$  is added to the loss function. The experimental results shown in Fig.4 suggest that, to some extent, the  $loss_{agv}$  is indeed helpful to get back the performance of the CNN block.

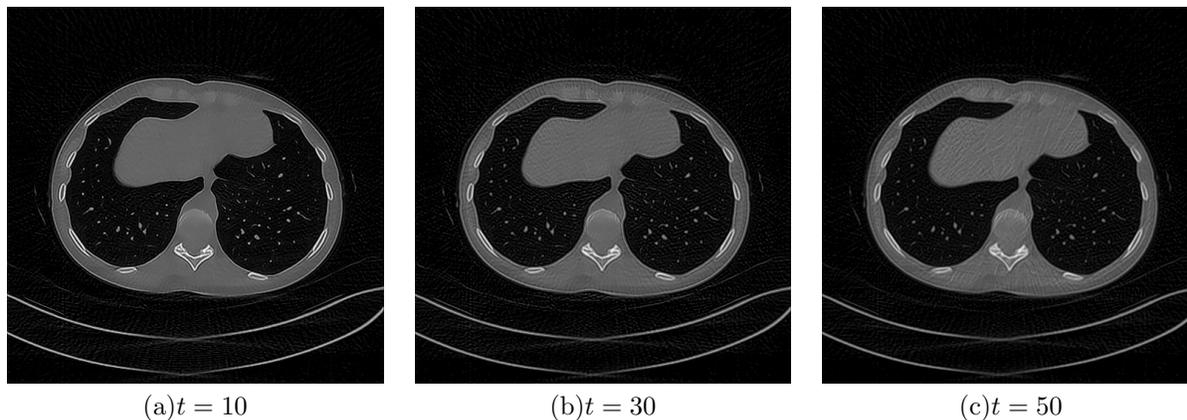


Figure 3. Testing results of the CNN block, with iteration number  $t = 10, 30, 50$  for the primal-dual algorithm, respectively.

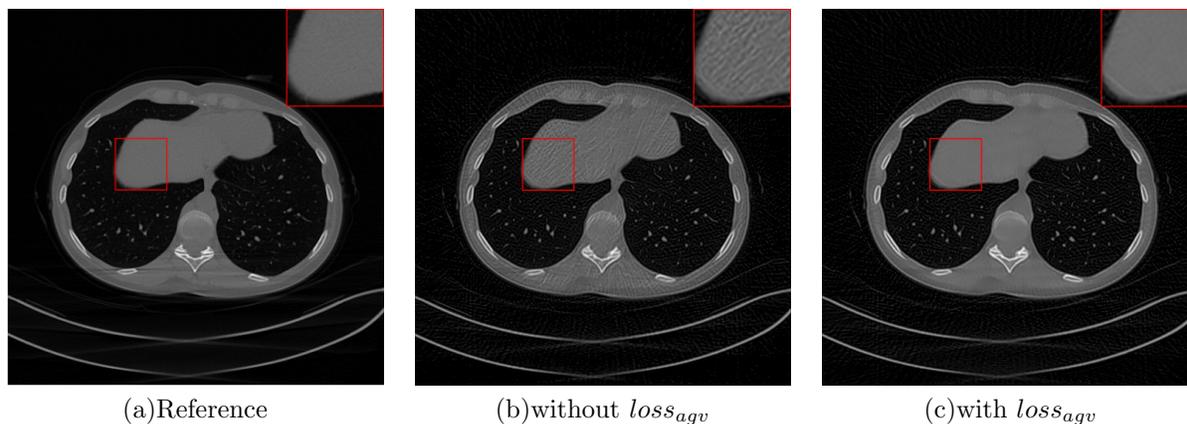


Figure 4. Results without and with  $loss_{agv}$ ,  $t = 50$ . The upper right corners show the zoomed-ins for the red-framed regions.

### 3.4 Comparison with the Noise2Inverse model

To verify the superiority of the proposed hybrid NN, experiments against the Noise2Inverse model are performed. For both methods, the training runs for 150 epochs. Even though there are many perspectives for comparison, in this experiment, we check how the two methods behave against noise change. Two tests with different noise levels, i.e.  $I_0 = 3 \times 10^4$  and  $I_0 = 1 \times 10^4$  are performed, and the results are shown in Fig.5. For the low level noise case ( $I_0 = 3 \times 10^4$ ), the two methods achieve similar quality. As Fig.5 (b) and (c) show there are little visual difference. The quantitative PSNR indices are 32.522 and 33.820, and the SSIM values are 0.735 and 0.799,

respectively, which might weakly indicate the advantages of the proposed hybrid NN. When checking the results with higher noise level ( $I_0 = 1 \times 10^4$ ), however, the advantages of the proposed hybrid NN become apparent. As shown in Fig.5(e) and (f), there are remaining noise with the result of Noise2Inverse, while hybrid NN still achieve high quality denoising, similar to the low level noise case.

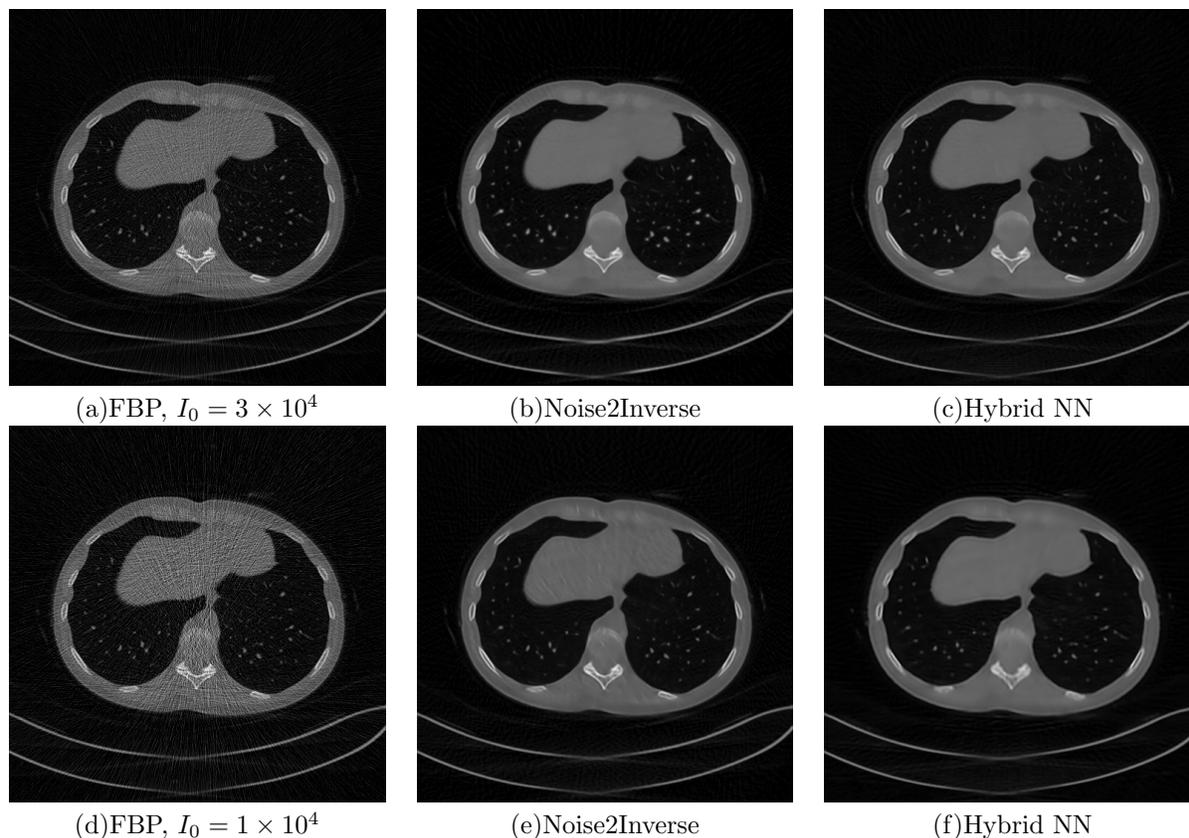


Figure 5. Comparison with Noise2Inverse. The two rows show the results with noise levels corresponding to  $I_0 = 3 \times 10^4$  and  $I_0 = 1 \times 10^4$ , respectively.

#### 4. CONCLUSION

Deep learning based methods achieve state-of-the-art results for low-dose CT reconstructions. The required training data-sets, however, might not be available for real applications. Another important issue is that neural networks usually suffer from intrinsic instability. In this paper, we propose a hybrid NN aiming to leverage the power of both learning based methods and conventional optimization based methods. By adding an average contrast preserving loss, the two blocks, i.e. CNN and optimization algorithm, could work in harmony such that the hybrid NN performs better than any of them alone.

Even though the hybrid NN in this paper consists of a CNN block and a TV denoising algorithm, the basic idea actually allows for any possible combinations of a neural network architecture and a optimization model.

#### ACKNOWLEDGMENTS

Thanks for the support of the National Natural Science Foundation of China (NSFC) (61971292, 61827809 and 61871275).

## REFERENCES

- [1] Buades, A., Coll, B., and Morel, J.-M., “A non-local algorithm for image denoising,” in [*2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*], **2**, 60–65 vol. 2 (2005).
- [2] Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K., “Image denoising by sparse 3-d transform-domain collaborative filtering,” *IEEE Transactions on Image Processing* **16**(8), 2080–2095 (2007).
- [3] Ronneberger, O., Fischer, P., and Brox, T., “U-net: Convolutional networks for biomedical image segmentation,” in [*International Conference on Medical image computing and computer-assisted intervention*], 234–241, Springer (2015).
- [4] Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L., “Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising,” *IEEE Transactions on Image Processing* **26**(7), 3142–3155 (2017).
- [5] Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., and Aila, T., “Noise2noise: Learning image restoration without clean data,” *arXiv preprint arXiv:1803.04189* (2018).
- [6] Batson, J. and Royer, L., “Noise2self: Blind denoising by self-supervision,” in [*International Conference on Machine Learning*], 524–533, PMLR (2019).
- [7] Krull, A., Buchholz, T.-O., and Jug, F., “Noise2void-learning denoising from single noisy images,” in [*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*], 2129–2137 (2019).
- [8] Hendriksen, A. A., Pelt, D. M., and Batenburg, K. J., “Noise2inverse: Self-supervised deep convolutional denoising for tomography,” *IEEE Transactions on Computational Imaging* **6**, 1320–1335 (2020).
- [9] Antun V, Renna F, P. C. A. B. H. A., “On instabilities of deep learning in image reconstruction and the potential costs of ai,” *Proceedings of the National Academy of Sciences*, *117*(48):30088–30095 (2020).
- [10] Jia, F., Liu, J., and Tai, X., “A regularized convolutional neural network for semantic image segmentation,” *ArXiv abs/1907.05287* (2020).