Study on analytical noise propagation in convolutional neural network methods used in computed tomography imaging

Xiao-Yue Guo[1,2] · Li Zhang[1,2] · Yu-Xiang Xing[1,2]

Abstract Neural network methods have recently emerged as a hot topic in computed tomography (CT) imaging owing to their powerful fitting ability; however, their potential applications still need to be carefully studied because their results are often difficult to interpret and are ambiguous in generalizability. Thus, quality assessments of the results obtained from a neural network are necessary to evaluate the neural network. Assessing the image quality of neural networks using traditional objective measurements is not appropriate because neural networks are nonstationary and nonlinear. In contrast, subjective assessments are trustworthy, although they are time- and energy-consuming for radiologists. Model observers that mimic subjective assessment require the mean and covariance of images, which are calculated from numerous image samples; however, this has not yet been applied to the evaluation of neural networks. In this study, we propose an analytical method for noise propagation from a single projection to efficiently evaluate convolutional neural networks (CNNs) in the CT imaging field. We propagate noise through nonlinear layers in a CNN using the Taylor expansion. Nesting of the linear and nonlinear layer noise propagation constitutes the covariance estimation of the CNN. A commonly used U-net structure is adopted for validation. The results reveal that the covariance estimation obtained from the proposed analytical method agrees well with that obtained from the image samples for different phantoms, noise levels, and activation functions, demonstrating that propagating noise from only a single projection is feasible for CNN methods in CT reconstruction. In addition, we use covariance estimation to provide three measurements for the qualitative and quantitative performance evaluation of U-net. The results indicate that the network cannot be applied to projections with high noise levels and possesses limitations in terms of efficiency for processing low-noise projections. U-net is more effective in improving the image quality of smooth regions compared with that of the edge. LeakyReLU outperforms Swish in terms of noise reduction.

Keywords Noise propagation · Convolutional neural network · Image quality assessment

✉ Yu-Xiang Xing
xingyx@tsinghua.edu.cn

1 Department of Engineering Physics, Tsinghua University, Beijing 100084, China

2 Key Laboratory of Particle and Radiation Imaging, Tsinghua University, Beijing 100084, China

1 Introduction

In recent years, neural networks have been applied in computed tomography (CT) imaging. Several studies on the applications of such networks have been published, and their potential in solving several problems in the field of CT imaging has been extensively evaluated. Such networks have been used in applications such as spectral distortion correction [1, 2] for photon-counting X-ray CT, dual-domain learning for two-dimensional [3, 4] and three-dimensional [5–7] low-dose CT reconstruction, sparse-view [8] and limited-angle [9] CT reconstruction, noise suppression in the sinogram domain [10] and image domain [11], dual-energy imaging with energy-integrating

detectors [12] and photon-counting detectors [13], and CT artifact reduction [14, 15]. However, neural networks have not yet been widely used in practice owing to their confidence scores. Hence, researchers have adopted objective and subjective image quality assessments to evaluate the feasibility of using neural networks in CT imaging [16, 17]. However, traditional metrics are preferred for objective assessments. The commonly used contrast-to-noise ratio measures the region of interest (ROI) clarity, signal-to-noise ratio measures the noise level, noise power spectrum (NPS) measures the noise correlation, and modulation transfer function measures the spatial frequency [18, 19]. Neural networks are normally nonlinear, nonstationary, and unexplainable; however, to the best of our knowledge, limited research on theoretically tractable methods has been conducted.

Subjective assessments are commonly used in the field of CT imaging. Radiologists are invited to observe and score images obtained from various methods [20, 21]. Because subjective assessments are time-consuming and laborious, researchers study model observers to simulate the evaluation behavior of radiologists. The assessment of image quality carried out by a radiologist, that is, a human observer is modeled as a classification problem solved by hypothesis testing. The likelihood ratio is used as the decision variable to obtain an ideal observer (IO). Because the IO is intractable, a linear approximation of the IO is assumed to obtain a Hotelling observer (HO). Combining the human-eye model with HO, researchers have obtained the most widely used channelized Hotelling observer (CHO) [22]. The CHO agrees well with human observers [23, 24]; however, it requires knowledge of the image mean and covariance. Recently, neural networks have been introduced to explore nonlinear model observers that enable better approximation of the human observer or better detectability for auxiliary diagnoses [25–27]. However, these methods only target traditional reconstruction methods [28, 29]. The application of these methods to neural network reconstruction has not yet been explored.

Noise propagation through a reconstruction neural network is necessary for assessing the performance of the network. Covariance prediction can reveal the uncertainty in inference, thus providing a safer answer to the CT reconstruction problem. Furthermore, it can be used in the calculation of model observers and subjective assessments.

The covariance estimation of neural-network outputs is currently of significant interest. Abdelaziz et al. [30] studied the uncertainty propagation of deep neural networks for automatic speech recognition with the assumption that the output was Gaussian. Lee et al. [31] used a Gaussian process equivalent to a deep fully connected neural network to obtain an exact Bayesian inference under the assumption that the parameters and layer outputs follow an independent and identical distribution. Tanno et al. [32] simultaneously estimated the mean and covariance of high-resolution outputs from low-resolution magnetic resonance images based on the assumption of a Gaussian distribution. However, for CT imaging, covariance estimation of the neural network output has not yet been studied.

In this study, we propose a new analytical noise propagation method, that is, covariance estimation, particularly for a convolutional neural network (CNN) in CT imaging of noisy projections. With a trained CNN ready for inference, we propagate the noise layer by layer. For linear layers, the output covariance can be calculated accurately. For nonlinear activation layers, we perform a Taylor expansion to obtain a linear approximation, which enables linear propagation of noise through the nonlinear layers. Because a CNN is a stack of linear and nonlinear layers, its covariance estimation is a combination of layer noise propagations. We validate the results of the covariance estimation method by comparing the results with those of statistical estimations using noisy projection and reconstruction samples with different phantoms, noise levels, and activation functions.

2 Methods

2.1 Physical model for CT imaging

A simple model for data acquisition in a CT scan is formulated as

$$I = I_0 e^{-p} = I_0 e^{-\int \mu dr}, \tag{1}$$

where I represents detected photons, and I_0 represents incident photons. Projection p is a line integral of the linear attenuation coefficient μ. Usually, the noise distribution of I is assumed to be Poisson or Gaussian, and the noise distribution of p can be approximated as Gaussian with mean $\overline{p}$ and variance $\exp(\overline{p})/I_0$, that is, $p \sim \mathcal{N}(\overline{p}, \exp(\overline{p})/I_0)$.

Analytical, iterative, and neural network methods can be used to reconstruct a linear attenuation coefficient map $\boldsymbol{\mu}$ from its projection $\mathbf{p}$. A CNN is a commonly used reconstruction method for CT imaging. A CNN typically consists of five types of basic layers: convolution, activation, batch normalization, pooling, and up-sampling layers. The overall operation of the network is a cascade of these layers, that is,

$$\boldsymbol{\mu} = \Phi(\mathbf{p}) = \varphi_L(\varphi_{L-1}(\cdots \varphi_l \cdots (\varphi_2(\varphi_1(\mathbf{p}))))), \tag{2}$$

where φ_l denotes the operation function of one layer, and Φ denotes the overall function of the neural network. Evidently, the noise in $\mathbf{p}$ will result in a noisy $\boldsymbol{\mu}$, even though

several networks are used to reduce noise. We discovered that the noise propagation through φ_l's to output $\boldsymbol{\mu}$ can be studied step-by-step if a network model is ready to serve for inference, that is, network parameters are set. The key lies in sorting out the covariance estimation through the five basic layers constituting the entire CNN.

2.2 Covariance propagation through basic layers of a CNN

Let vector $\mathbf{x} \in \mathbb{R}^{M \times 1}$ be an arbitrary layer input, and let $\mathbf{y}$ be the corresponding layer output. This section presents the covariance estimation of $\mathbf{y}$ from $\mathbf{x}$.

2.2.1 Convolution layer

For a convolutional layer, its output $\mathbf{y}_{\text{conv}} \in \mathbb{R}^{N \times 1}$ can be expressed as a linear combination of inputs:

$$\mathbf{y}_{\text{conv}} = \sum_{i=1}^{C} \mathbf{W}_i \mathbf{x}_i + b_i, \tag{3}$$

where C denotes the number of input channels, $\mathbf{W} \in \mathbb{R}^{N \times M}$ represents the convolutional weighting matrix, and b_i denotes bias. The convolution is linear; hence, it is easy to obtain:

$$\text{Cov}(\mathbf{y}_{\text{conv}}) = \sum_{j=1}^{C} \sum_{i=1}^{C} \mathbf{W}_i \text{Cov}(\mathbf{x}_i, \mathbf{x}_j) \mathbf{W}_j^{\text{T}}. \tag{4}$$

2.2.2 Activation layer

In an activation layer, an input is normally fed into a nonlinear function, $f(\cdot)$:

$$\mathbf{y}_{\text{af}} = f(\mathbf{x}). \tag{5}$$

Because it is nonlinear, we perform the 1st order Taylor expansion to obtain its linear approximation:

$$\begin{aligned} \mathbf{y}_{\text{af}} &= f(\mathbf{x}) \simeq f(\overline{\mathbf{x}}) + f'(\overline{\mathbf{x}}) \cdot (\mathbf{x} - \overline{\mathbf{x}}) \\ &= \mathbf{F}(\overline{\mathbf{x}})\mathbf{x} + (f(\overline{\mathbf{x}}) - \mathbf{F}(\overline{\mathbf{x}})\overline{\mathbf{x}}) \end{aligned} \tag{6}$$

where the Taylor-based coefficient matrix $\mathbf{F} \in \mathbb{R}^{M \times M}$ is diagonal to $[\mathbf{F}]_{m,m} = f'([\overline{\mathbf{x}}]_m)$. Thus, the covariance of the nonlinear transformation layer can be estimated by

$$\text{Cov}(\mathbf{y}_{\text{af}}) \simeq \mathbf{F}\text{Cov}(\mathbf{x})\mathbf{F}^{\text{T}}. \tag{7}$$

2.2.3 Batch normalization layer

For a batch normalization layer, the input is normalized as

$$\mathbf{y}_{\text{bn}} = \frac{\gamma}{\sqrt{\sigma_{\text{B}}^2 + \varepsilon}}\mathbf{x} + \left(\beta - \frac{\gamma u_{\text{B}}}{\sqrt{\sigma_{\text{B}}^2 + \varepsilon}} \right). \tag{8}$$

Here, γ and β are hyperparameters that are learned during training and frozen when inferencing. u_{B} and σ_{B}^2 represent the batch mean and variance, respectively, which are also frozen during inference. Hence, the covariance propagating through the batch normalization layer is

$$\text{Cov}(\mathbf{y}_{\text{bn}}) = \frac{\gamma^2}{\sigma_{\text{B}}^2 + \varepsilon}\text{Cov}(\mathbf{x}) \equiv \eta\text{Cov}(\mathbf{x}). \tag{9}$$

2.2.4 Pooling layer

Average pooling is a widely used method in this field. This can be interpreted as a convolution operation of kernel size k with stride s, where the convolution kernel is a constant matrix with value $1/k^2$. Similar to the operation of a convolution layer, its output $\mathbf{y}_{\text{ap}} \in \mathbb{R}^{N \times 1}$ can be formulated as a linear transformation of the input as follows:

$$\mathbf{y}_{\text{ap}} = \mathbf{A}\mathbf{x}. \tag{10}$$

Here, the average pooling matrix $\mathbf{A} \in \mathbb{R}^{N \times M}$ is sparse with $N = (M - k)/s + 1$. Thus,

$$\text{Cov}(\mathbf{y}_{\text{ap}}) = \mathbf{A}\text{Cov}(\mathbf{x})\mathbf{A}^{\text{T}}. \tag{11}$$

2.2.5 Up-sampling layer

For an up-sampling layer, each element of the input is duplicated and can be expressed as a linear combination of the input:

$$\mathbf{y}_{\text{up}} = \mathbf{U}\mathbf{x}, \tag{12}$$

where the upsampling matrix $\mathbf{U} \in \mathbb{R}^{N \times M}$ is a sparse matrix with only one element in each row and $N = 2M$. The covariance estimated from an up-sampling layer is

$$\text{Cov}(\mathbf{y}_{\text{up}}) = \mathbf{U}\text{Cov}(\mathbf{x})\mathbf{U}^{\text{T}}. \tag{13}$$

2.3 Example: U-net

We adopt a commonly used U-net structure to denoise the projection, followed by reconstruction with the filter back projection (FBP) method:

$$\boldsymbol{\mu} = \Phi(\mathbf{p}) = \mathcal{O}(\varphi_L(\varphi_{L-1}(\cdots \varphi_l \cdots (\varphi_2(\varphi_1(\mathbf{p})))))). \tag{14}$$

Here, $\mathcal{O}$ represents the linear FBP operator. $\mathbf{p} \in \mathbb{R}^{M \times 1}$ denotes an input projection, and $\boldsymbol{\mu} \in \mathbb{R}^{N \times 1}$ represents the corresponding reconstruction. The reconstruction

Fig. 1 (Color online) U-net structure for CT reconstruction. The projection is first filtered by U-net and then reconstructed by FBP. Two activation functions: ① LeakyReLU and ② Swish, are applied to U-net
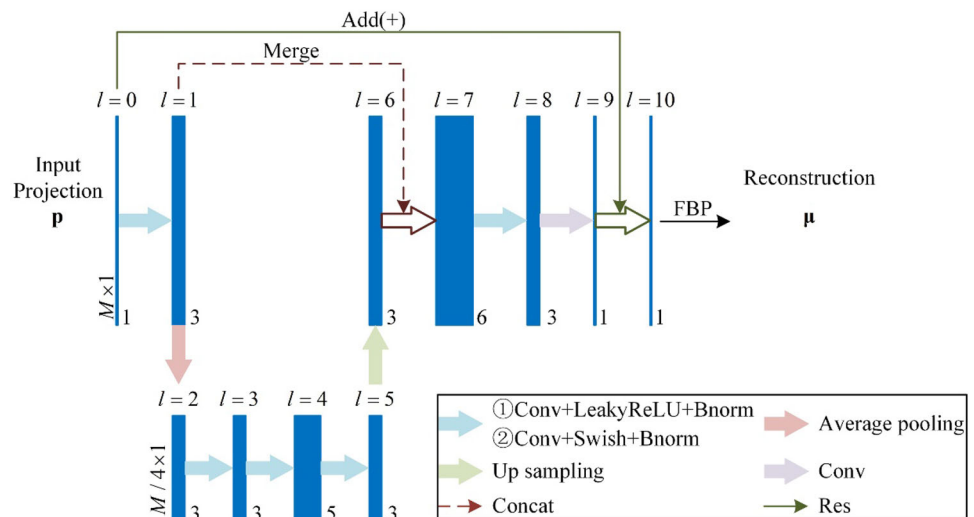
flowchart is illustrated in Fig. 1. A concatenate layer and residual layer are included in U-net. The concatenated layer merges the feature map in the 1st layer to the 6th layer, and the residual layer adds the input projection to the 9th layer to obtain the output projection.

We iteratively estimate the covariance of reconstruction predicted from the trained U-net:

$$\mathrm{Cov}(\mathbf{z}_i^l,\mathbf{z}_j^l) \simeq \begin{cases} \eta_i^l \mathbf{F}_i^l \left[\sum_{j'=1}^{C^{l-1}} \sum_{i'=1}^{C^{l-1}} \mathbf{W}_{i',i}^l \mathrm{Cov}(\mathbf{z}_{i'}^{l-1},\mathbf{z}_{j'}^{l-1})(\mathbf{W}_{j',j}^l)^{\mathrm{T}} \right] (\eta_j^l \mathbf{F}_j^l)^{\mathrm{T}}, & l=1,3,4,5,8,9 \\ \mathbf{A}_i \mathrm{Cov}(\mathbf{z}_i^{l-1},\mathbf{z}_j^{l-1})\mathbf{A}_j^{\mathrm{T}}, & l=2 \\ \mathbf{U}_i \mathrm{Cov}(\mathbf{z}_i^{l-1},\mathbf{z}_j^{l-1})\mathbf{U}_j^{\mathrm{T}}, & l=6 \end{cases}$$

$$(15)$$

where $\mathbf{z}$ represents the latent variable in the hidden layer of U-net. $\mathbf{z}^0 = \mathbf{p}$ when $l=1$ and $[\mathrm{Cov}(\mathbf{p})]_{n,n} = exp(\bar{\mathbf{p}})/I_0$. The subscript $\mathbf{W}_{i',i}^l$ denotes a convolutional weighting matrix from the i'th channel input of the lth layer to its ith channel output, where C^l represents the total number of channels in the lth layer.

With a concatenation operation, the 7th layer contains feature maps from the 1st and 6th layers. Thus, three types of covariances for the 7th layer must be present: (1) covariance between channels in the 6th layer, (2) covariance between channels in the 1st layer, and (3) covariance between channels in the 1st layer and 6th layer. The covariance of cases (1) and (2) has already been estimated using Eq. (15), and the covariance of case (3) can be estimated as

$$\mathrm{Cov}(\mathbf{z}_i^6,\mathbf{z}_j^1) = \mathrm{Cov}(\mathbf{z}_i^1,\mathbf{z}_j^6)^T = \mathbf{U}_i \mathrm{Cov}(\mathbf{z}_i^5,\mathbf{z}_j^1)$$

where $\mathrm{Cov}(\mathbf{z}_i^l,\mathbf{z}_j^1) = \mathbf{A}_i \mathrm{Cov}(\mathbf{z}_i^{l-1},\mathbf{z}_j^1),\ l=2$

$$\mathrm{Cov}(\mathbf{z}_i^l,\mathbf{z}_j^1) \simeq \eta_i^l \mathbf{F}_i^l \left[\sum_{i'=1}^{C^{l-1}} \mathbf{W}_{i',i}^l Cov(\mathbf{z}_{i'}^{l-1},\mathbf{z}_j^1) \right],\ l=3,4,5$$

$$(16)$$

Therefore, the covariance estimation of the 7th layer is

$$\mathrm{Cov}(\mathbf{z}_i^7,\mathbf{z}_j^7) = \begin{cases} \mathrm{Cov}(\mathbf{z}_i^6,\mathbf{z}_j^6), & i,j=1,2,3 \\ \mathrm{Cov}(\mathbf{z}_i^6,\mathbf{z}_j^1), & i=1,2,3, j=4,5,6 \\ \mathrm{Cov}(\mathbf{z}_i^1,\mathbf{z}_j^6), & i=4,5,6, j=1,2,3 \\ \mathrm{Cov}(\mathbf{z}_i^1,\mathbf{z}_j^1), & i,j=4,5,6 \end{cases}$$

$$(17)$$

With a residual operation, the output projection represents the sum of the input projection and output residue of the 9th layer. Thus, the covariance of the output projection also consists of three parts: (1) covariance of the input projection, (2) covariance of the output residue of the 9th layer, and (3) covariance between the input projection and output residue. Only the covariance estimation of case (3) should be calculated because cases (1) and (2) are estimated using Eq. (15):

$$\mathrm{Cov}(\mathbf{z}^9,\mathbf{p}) = \mathrm{Cov}(\mathbf{p},\mathbf{z}^9)^T = \sum_{i=1}^{C^8} \mathbf{W}_i^9 \mathrm{Cov}(\mathbf{z}_i^8,\mathbf{p})$$

where $\mathrm{Cov}(\mathbf{z}_i^l,\mathbf{p}) \simeq \eta_i^l \mathbf{F}_i^l \left[\sum_{i'=1}^{C^{l-1}} \mathbf{W}_{i',i}^l \mathrm{Cov}(\mathbf{z}_{i'}^{l-1},\mathbf{p}) \right],\ l=1,3,4,5,8$

$$\mathrm{Cov}(\mathbf{z}_i^l,\mathbf{p}) = \mathbf{A}_i Cov(\mathbf{z}_i^{l-1},\mathbf{p}),\ l=2$$
$$\mathrm{Cov}(\mathbf{z}_i^l,\mathbf{p}) = \mathbf{U}_i Cov(\mathbf{z}_i^{l-1},\mathbf{p}),\ l=6$$
$$\mathrm{Cov}(\mathbf{z}_i^l,\mathbf{p}) = \begin{cases} \mathrm{Cov}(\mathbf{z}_i^6,\mathbf{p}), & i=1,2,3 \\ \mathrm{Cov}(\mathbf{z}_i^1,\mathbf{p}), & i=4,5,6 \end{cases},\ l=7$$

$$(18)$$

The covariance estimation of the 10th layer is then calculated as

$$\mathrm{Cov}(\mathbf{z}^{10}) = \mathrm{Cov}(\mathbf{p}) + \mathrm{Cov}(\mathbf{z}^9) + \mathrm{Cov}(\mathbf{z}^9,\mathbf{p}) + \mathrm{Cov}(\mathbf{p},\mathbf{z}^9).$$

$$(19)$$

Combining Eqs. (15)–(19), we obtain the final covariance estimation of the reconstruction:

$$\text{Cov}(\boldsymbol{\mu}) = \mathcal{O}\text{Cov}(\mathbf{z}^{10})\mathcal{O}^{\text{T}}. \tag{20}$$

A gradient discontinuous function, LeakyReLU, and a gradient continuous function, Swish, are chosen as activation functions to detect the influence of activation functions on the covariance estimation of the CNN.

2.3.1 LeakyReLU activation function

The LeakyReLU function and its corresponding gradient are expressed as

$$f_{\text{LR}}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha x, & \text{if } x \le 0 \end{cases}$$
$$f'_{\text{LR}}(x) = \begin{cases} 1, & \text{if } x > 0 \\ \alpha, & \text{if } x \le 0 \end{cases} \tag{21}$$

The Taylor-based coefficient matrix in Eq. (6) is $[\mathbf{F}_{\text{LR}}]_{m,m} = f'_{\text{LR}}([\overline{\mathbf{x}}]_m)$. Plugging $\mathbf{F}_{\text{LR}}$ into Eqs. (15)–(20), we obtain the covariance estimation from U-net with LeakyReLU.

2.3.2 Swish activation function

The Swish function is another commonly used activation function:

$$f_{\text{Sw}}(x) = \frac{x}{1 + \exp(-\lambda x)}. \tag{22}$$

Its gradient is

$$f'_{\text{Sw}}(\overline{x}) = \lambda f_{\text{Sw}}(\overline{x}) + \frac{1 - \lambda f_{\text{Sw}}(\overline{x})}{1 + \exp(-\lambda \overline{x})}. \tag{23}$$

Thus, the Taylor-based coefficient matrix is $[\mathbf{F}_{\text{Sw}}]_{m,m} = f'_{\text{Sw}}([\overline{\mathbf{x}}]_m)$. Replacing $\mathbf{F}$ in Eqs. (15)–(18) with $\mathbf{F}_{\text{Sw}}$, we obtain the covariance estimation of the projection by U-net with Swish using Eq. (19) and the covariance estimation of its reconstruction by FBP using Eq. (20).

3 Experiments

The projection data used for training are generated from the Grand Challenge dataset of the Mayo Clinic. We randomly choose a reconstruction dataset of one patient and select various ROIs with sizes of 128×128 pixels as phantoms. Geometrical parameters of the simulated system are listed in Table 1. By setting the number of incident photons to $I_0 = 10^4$, we add Poisson noise to the noise-free projections simulated from phantoms to obtain noisy projections.

Using noise-free projections as labels and noisy projections as inputs, we train U-net by minimizing an L2 norm loss function between labels and outputs. In addition,

Table 1 System geometry of simulation experiments

Parameter	Value
Source to origin distance (mm)	200
Origin to detector distance (mm)	200
Scanning angle	2π
Detector pixel size (mm)	0.5
Detector pixel number	240
Image pixel size (mm)	0.3
Image pixel number	128

we fix the hyper-parameter $\alpha=0.1$ for LeakyReLU and $\lambda=0.5$ for Swish. We simulate 1792 noisy projections for the study. The dataset is randomly split into a training set and a validation set, where 80% of the dataset represents the training set and 20% represents the validation set. We train the network with Keras on a GPU RTX8000 of 48G. The loss function of U-net with LeakyReLU and Swish decreases to approximately 10^{-4} during converging. Furthermore, we randomly split the dataset into 5 folds to run a fivefold cross validation on the trained network. The average loss in the fivefold cross validation is also approximately 10^{-4}, which is similar to the loss of the trained network. Thus, the dataset is sufficient for training the small-size U-net in Fig. 1, and the trained network is stable.

Noisy projections used for inference are generated from another patient dataset in the same manner. We generate noisy projections of different phantoms and noise levels to validate the proposed analytical noise propagation method and analyze the performance of U-net using the analytically estimated covariance. Information on the noisy projections generated for prediction is presented in Table 2. Note that the number of incident photons increases linearly from 10^3 to 5.05×10^4. The reconstruction of both phantoms using $I_0 = 10^4$ is illustrated in Fig. 2.

In addition, we conduct a practical experiment to validate our proposed method. The experimental platform and phantom are presented in Fig. 3. The scanning parameters are presented in Table 3. We repeat the scan 450 times at each angle and acquire projection data of 360 views using 2π. Considering the computational cost, every four pixels of the detector are binned into one to obtain a projection with a smaller size.

Covariance estimation from a statistical method is used as a reference in this study.

$$\text{Cov}^*(\hat{\mathbf{p}}) = \frac{1}{K-1} \sum_{k=1}^{K} (\hat{\mathbf{p}}_k - \overline{\mathbf{p}})(\hat{\mathbf{p}}_k - \overline{\mathbf{p}})^{\text{T}}$$
$$\text{Cov}^*(\hat{\mu}) = \frac{1}{K-1} \sum_{k=1}^{K} (\hat{\mu}_k - \overline{\mu})(\hat{\mu}_k - \overline{\mu})^{\text{T}} \tag{24}$$

Table 2 Information on the noisy projections generated for inference

Number of generated noise realizations	$I_0 = 10^3$	$I_0 = 5.5 \times 10^3$	$I_0 = 10^4$	$I_0 = 3.25 \times 10^4$	$I_0 = 5.05 \times 10^4$
Test phantom A	–	–	1000	–	–
Test phantom B	1000	1000	1000	1000	1000

Fig. 2 Denoised projections of **a** test phantom A and **b** test phantom B for $I_0 = 10^4$ obtained from U-net and their reconstructions obtained from FBP
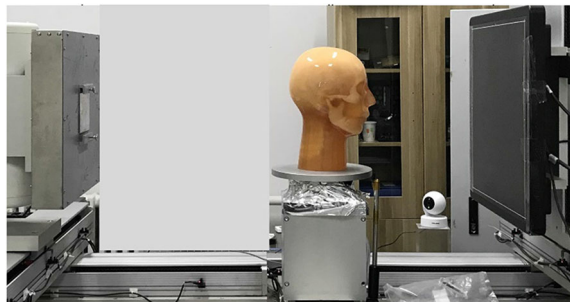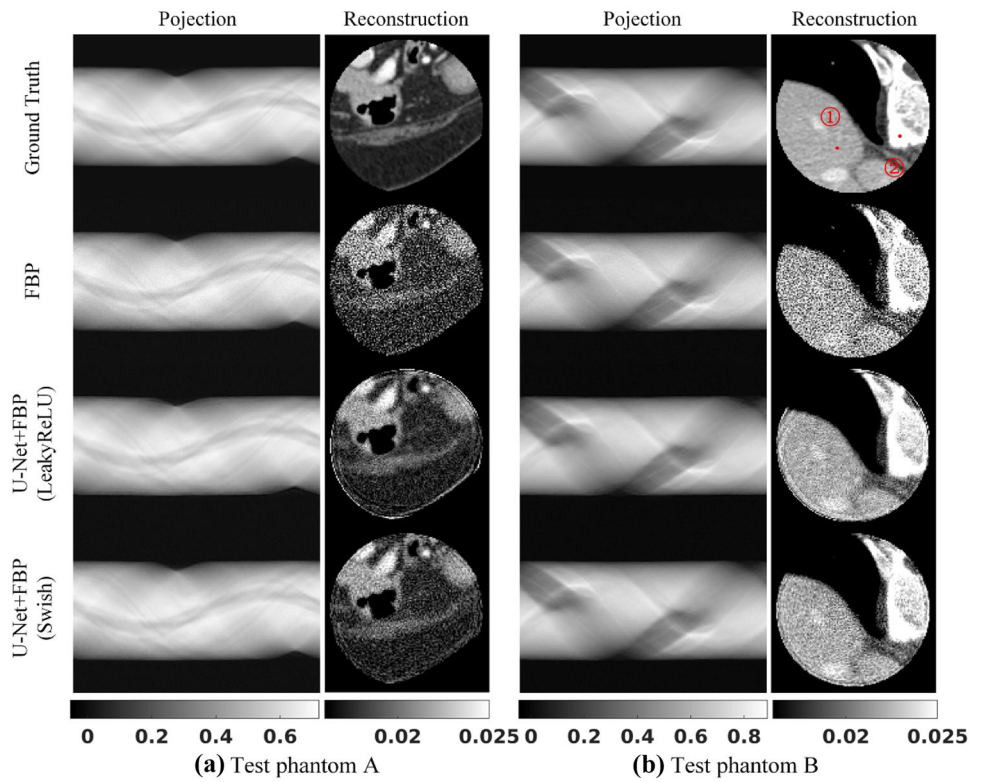
(a) Test phantom A

(b) Test phantom B

Fig. 3 (Color online) Experimental platform and phantom

Table 3 Scanning parameters of the practical experiment

Parameters	Before binning	After binning
Source to origin distance (mm)	750.2	750.2
Origin to detector distance (mm)	433.84	433.84
Detector pixel size (mm)	0.45	1.80
Detector pixel number	896	224
Image pixel size (mm)	0.48	1.92
Image pixel number	512	128
Tube voltage/current	110 kV/ 2.5 mA	110 kV/ 2.5 mA

where the total number of noise realizations is $K = 1000$.

The generalization error (GE) [33] represents the sum of the bias and variance and measures the generalization ability of a neural network:

$$[GE]_n = ([\hat{\mu}]_n - [\mu^*]_n)^2 + [Cov(\hat{\mu})]_{n,n}, \tag{25}$$

where μ^* represents noise-free reconstruction.

A pixel-wise noise reduction percentage (NPR) is calculated to analyze the denoising performance of U-net.

$$[NRP]_n = \left| \frac{\sqrt{[Cov(\hat{\mu})]_{n,n}} - \sqrt{[Cov(\mu)]_{n,n}}}{\sqrt{[Cov(\mu)]_{n,n}}} \right| \times 100\% \tag{26}$$

Here, $\boldsymbol{\mu} = \mathcal{O}(\mathbf{p})$ and $\text{Cov}(\boldsymbol{\mu}) = \mathcal{O}\text{Cov}(\mathbf{p})\mathcal{O}^{\mathrm{T}}$, according to Eq. (20). We choose one low-attenuation point and one high-attenuation point in test phantom B to present the trend of GE with the noise level; the two points are marked by red dots in Fig. 2b.

We also calculate the NPS to analyze the noise spatial correlation of U-net in the Fourier domain:

$$\text{NPS} = \mathcal{F}[\text{Cov}(\hat{\mu})], \tag{27}$$

where $\mathcal{F}$ denotes the Fourier transform operator.

4 Experimental results

Because the linear approximation of nonlinear activation functions requires the mean of the input in Eq. (6), we only use one noise realization as the mean of the input to analytically estimate the covariance of the projection acquired by U-net and its corresponding reconstruction by FBP.

4.1 Validation of the proposed analytical covariance estimation method for U-net

For test phantom A, the variance of the projections obtained by U-net is illustrated in Fig. 4a, and its covariance at the center of the projections is presented in Fig. 4b. The results reveal that the variance estimation is in agreement with the reference for both the activation functions. The error between the variance estimation and reference is not significant compared with the variance itself. We observe that the variance obtained from LeakyReLU varies sharply when approaching the boundary, whereas that from Swish changes smoothly. Meanwhile, the covariance estimation also agrees with the reference, where the error is primarily statistical. The shape of the covariance from the two activation functions is quite different; it is circular for LeakyReLU and elliptical for Swish. For test phantom B, good agreement can still be observed between the variance estimation and reference (as shown in Fig. 4c). Sharp changes also occur near the boundary in the projection variance from LeakyReLU. As shown in Fig. 4d, the covariance estimation for both activation functions is in agreement with the reference.

The variance and covariance of the reconstructions obtained by FBP from projections denoised by U-net are presented in Fig. 5. For both phantoms and activation functions, the variance estimation of the reconstructions agrees with the reference because FBP is linear. Meanwhile, the error that propagates through the FBP is not a concern. We discover that the central areas of variance from the two activation functions have different appearances. The central area appears dark for LeakyReLU and bright for Swish in the same display window. The covariance estimation is yet again in agreement with the reference, leaving only statistical noise in the error map.

The profiles of the variance and covariance for the projections and reconstructions are plotted in Fig. 6. For both phantoms and activation functions, the profiles of the variance and covariance estimations match those of the references. As shown in Fig. 6a1, c1, the profile of the projection variance from Swish appears smooth, whereas that from LeakyReLU appears sharp. The profile of covariance (as shown in Fig. 6b1 and d1) from Swish demonstrates a larger spread, whereas that from LeakyReLU exhibits sharper changes. For the profiles of the reconstruction variance, displayed in Fig. 6a2 and c2, we discover that the noise from LeakyReLU is lower than that from Swish. The variance gradually decreases from the edge of the field of view (FOV) to its center for LeakyReLU, whereas it demonstrates an opposite behavior for Swish. Although the values of the projection covariance are close, the absolute value of the reconstruction covariance from LeakyReLU is much smaller than that from Swish; this demonstrates that the covariance from Swish is more structurally related and difficult to deal with.

In addition, we estimate the variance of the projections obtained by U-net with LeakyReLU under different noise levels for test phantom B. As shown in Fig. 7, the variance estimation agrees with the reference for different noise levels. Although the error between the variance estimation and reference increases with the noise level, it is still insignificant compared with its corresponding variance.
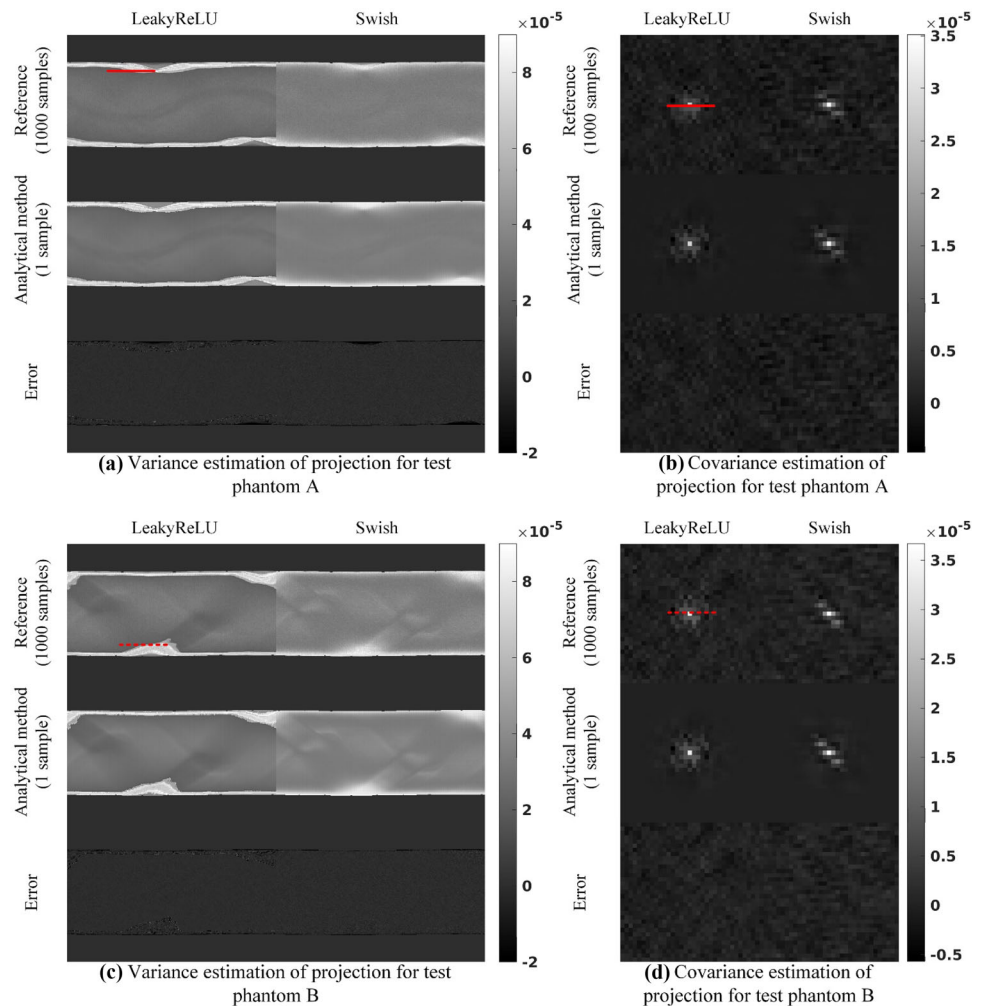
The noise estimation of U-net with LeakyReLU in the practical experiment is illustrated in Fig. 8. It is apparent that both the variance and covariance estimations from the analytical method agree well with the references, which strongly demonstrates the feasibility of the proposed analytical noise propagation method in practical usage.

4.2 Performance analysis with analytical covariance

Pixel-wise GE maps for test phantom B are illustrated in Fig. 9. For both activation functions, GE increases for each pixel with increasing noise levels, which indicates that U-net is inapplicable for highly noisy projections. When I_0 increases to a certain number, the decrease in GE is not significant. The GE in the smooth region is smaller than that at the edge when the number of incident photons increases to 3.25×10^4, indicating that U-net is more effective in smooth regions. Compared with the GE for LeakyReLU, the GE for Swish is relatively large in smooth regions, whereas it is almost the same at the edge.

Further, noise reduction percentage (NRPs) are listed in Table 4. For both activation functions, the increase in NRP

Fig. 4 Variance of the projections obtained from U-net and its covariance at the center of projections when $I_0 = 10^4$. **a** and **c** projection variance of phantom A and B used for testing, respectively. **b** and **d** corresponding covariance at the center of the projection

(a) Variance estimation of projection for test phantom A

(b) Covariance estimation of projection for test phantom A

(c) Variance estimation of projection for test phantom B

(d) Covariance estimation of projection for test phantom B

from $I_0 = 10^3$ to $I_0 = 5.5 \times 10^3$ is approximately 20%, and this increase quickly slows to approximately 5% or less. For both low- and high-attenuation points, the noise reduction effect of LeakyReLU is stronger than that of Swish at various noise levels. The NRP of LeakyReLU is approximately 10% higher than that of Swish, particularly for the low-attenuation point at $I_0 = 10^3$, and the difference decreases to approximately 1% for a low noise level with $I_0 = 5.05 \times 10^4$. For the high-attenuation point, the difference in the NRP for both activation functions is smaller than 3% and becomes even smaller when the number of incident photons increases. The NRPs at both points for LeakyReLU are comparable, which suggests that LeakyReLU treats low- and high-attenuation areas equally during noise suppression. The NRP at the low-attenuation point for Swish is slightly smaller than that at the high-attenuation point; however, the difference between the NRPs at the low- and high-attenuation points gradually reduces to approximately 1% with decreasing noise levels.
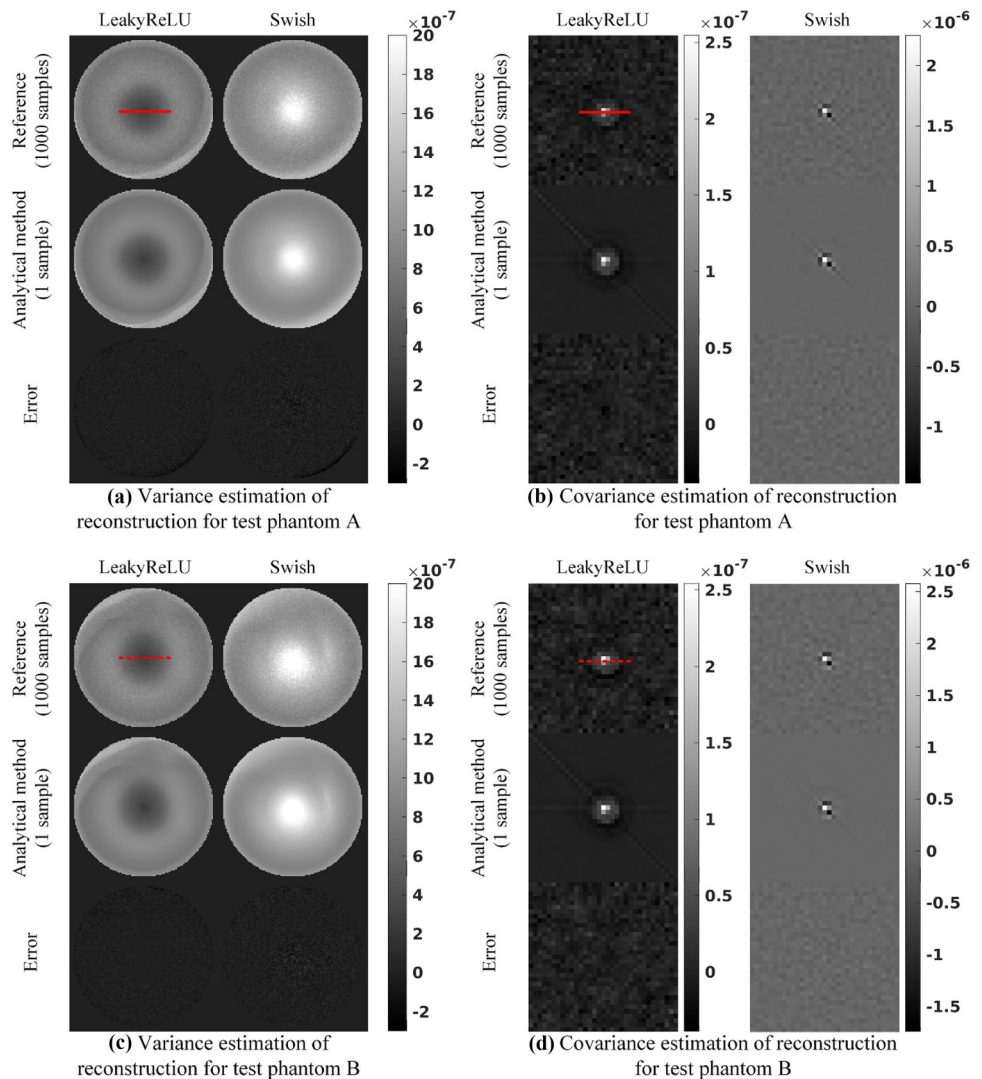
The NPS at the center of the reconstruction is illustrated in Fig. 10. For each noise level, the NPS at the center of the

reconstruction from both activation functions decreases as 10^3 increases to 5.05×10^4 and drops by approximately an order of magnitude from 10^3 to 5.5×10^3. The NPS from LeakyReLU first increases to a maximum and then decreases as the frequency increases, whereas that from Swish continues to increase with increasing frequency. Both LeakyReLU and Swish exhibit similar NPS shapes at low frequencies, indicating that their performance in dealing with low-frequency noise is comparable. The high-frequency noise in the NPS from LeakyReLU gradually reduces; however, it increases considerably for Swish, suggesting that more structures are present in the reconstruction noise propagated through U-net with Swish.

5 Discussion and conclusion

In this study, an analytical noise propagation method for CNNs in CT imaging is proposed. The five basic layers that comprise a typical CNN include the convolution, nonlinear activation, batch normalization, average pooling, and up-

Fig. 5 Variance of the reconstructions by FBP from projections obtained from U-net and its corresponding covariance at the center of the reconstructions for $I_0 = 10^4$. **a** the reconstruction variance of test phantom A and **b** its covariance at the center of reconstructions. **c** the reconstruction variance of test phantom B, and **d** its covariance at the center of reconstructions

(a) Variance estimation of reconstruction for test phantom A

(b) Covariance estimation of reconstruction for test phantom A

(c) Variance estimation of reconstruction for test phantom B

(d) Covariance estimation of reconstruction for test phantom B

sampling layers. Except for the nonlinear activation layer, the other four layers are all linear, which simplifies the estimation of the covariance of their output by linear propagation. The 1st order Taylor expansion is used to obtain the linear approximation of the nonlinear activation layer for linear propagation of noise. By integrating the noise propagation of both linear and nonlinear layers in the CNN, we can estimate the covariance of reconstruction from the projection in a step-by-step manner.

The results indicate that the covariance estimated by the proposed analytical method agrees well with that estimated by the statistical method, regardless of phantoms, noise levels, and activation functions. We demonstrate that it is feasible to propagate noise from only a single projection to image reconstructed from CNN. The covariance of the projection obtained from U-net with the gradient continuous activation function Swish is smooth, whereas that with the discontinuous gradient activation function LeakyReLU exhibits sharp changes near the boundary. The noise in the reconstruction from LeakyReLU is smaller than that in Swish. They demonstrate opposite performances, where the variance from Swish gradually decreases from the FOV edge to its center. Therefore, LeakyReLU and Swish are completely different in terms of noise suppression. The covariance for Swish spreads wider than that for LeakyReLU, which indicates that Swish uses the information of more neighborhood pixels in denoising.

We further qualitatively and quantitatively evaluate network performance from three aspects. The GE, which contains bias and variance, is a tradeoff between the accuracy and noise of the network output and measures the generalization of the neural network. Trained with data under the condition of $I_0 = 10^4$, the network fails to reduce the GE for projections with lower incident photons, which renders it unacceptable for application in projection denoising with high noise levels. This also limits its application to projections with lower noise when I_0 increases to a certain number because the improvement in

Fig. 6 (Color online) Profiles of the projection variance and covariance obtained from U-net, and profiles of its reconstruction variance and covariance obtained from FBP for $I_0 = 10^4$. **a1–d1** profiles of the projection variance and covariance: **a** profiles of the variance marked by a red solid line in Fig. 3a for test phantom A, and **b1** profiles of its covariance marked by a red solid line in Fig. 3b, c profiles of the variance marked by a red dashed line in Fig. 3c for test phantom B, and **d1** profiles of its covariance marked by a red dashed line in Fig. 3d. **a2–d2** profiles of the reconstruction variance and covariance: **a2** profiles of the variance marked by a red solid line in Fig. 4a for test phantom A, and **b2** profiles of its covariance marked by a red solid line in Fig. 4b, c2 profiles of the variance marked by a red dashed line in Fig. 4c, and d2 profiles of its covariance marked by a red dashed line in Fig. 4d
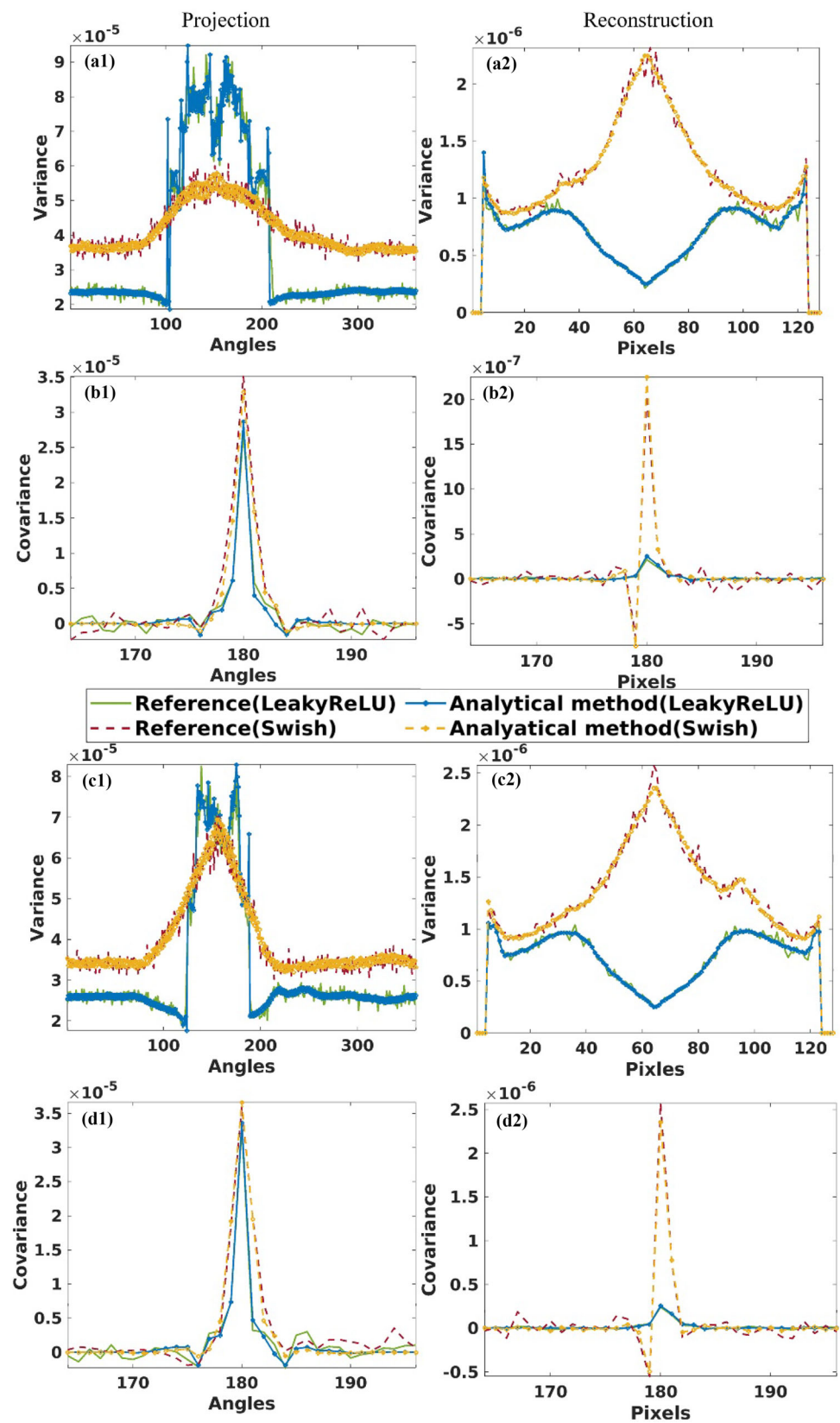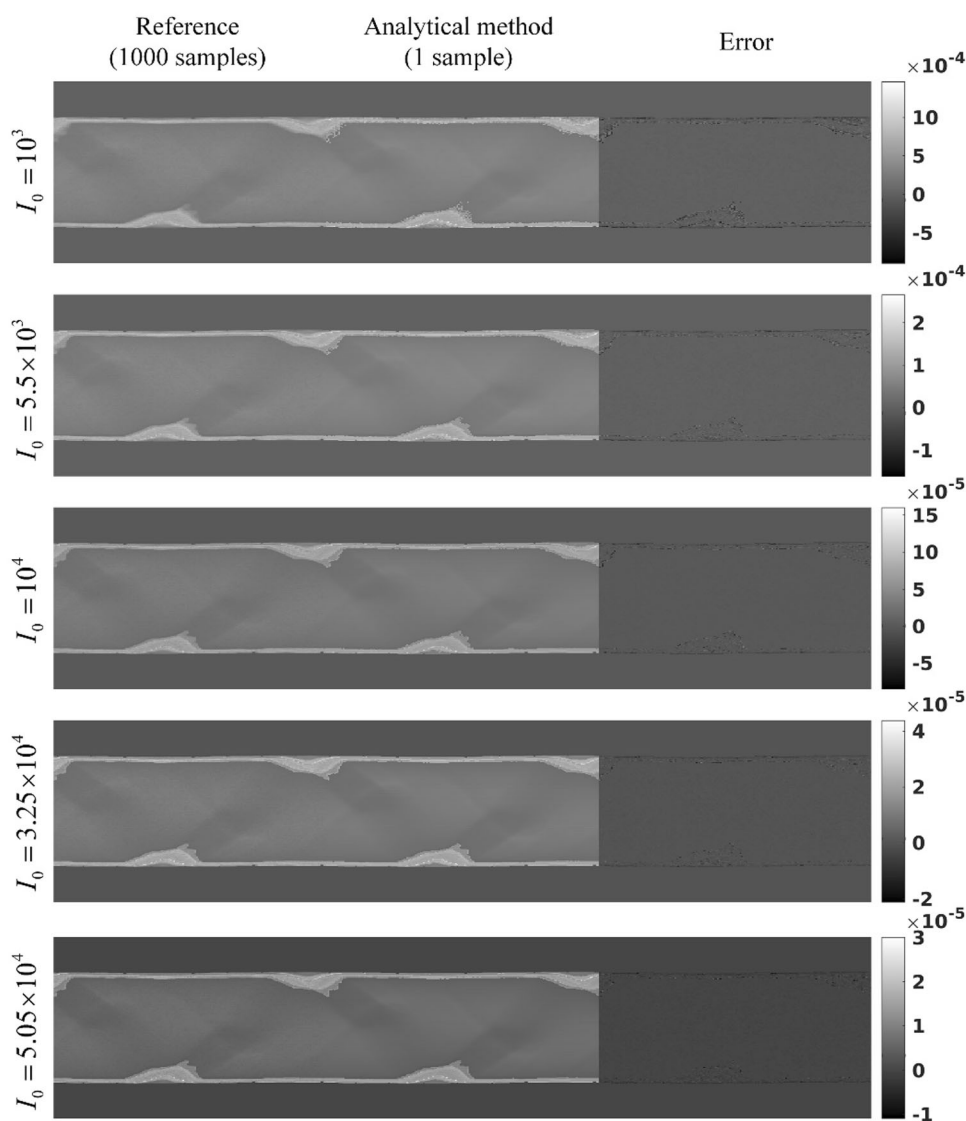
Fig. 7 Variance of projections obtained from U-net with LeakyReLU under different noise levels for test phantom B

GE is trivial. A pixel-wise NRP is defined to measure the denoising ability of the network. The effect of noise suppression is strong only when the noise level is sufficiently high; otherwise, it quickly weakens as the noise level decreases. An evident drop in GE can be observed in smooth regions but not at the edge when the number of incident photons increases, although the NPRs for smooth regions and the edge are the same. Therefore, the accuracy of the smooth regions is higher than that of the edges. In addition, the spatial correlation of noise is analyzed using the NPS. Consequently, it is discovered that there is no significant difference in NPS between LeakyReLU and Swish at low frequencies. However, the NPS at high frequencies is completely opposite for these two activation functions, where it weakens with increasing frequencies for LeakyReLU. The variance in projection denoised by the two activation functions is comparable; however, the NRP of the reconstruction from LeakyReLU is larger than that of Swish, and the NPS of the reconstruction from Swish increases as the frequency increases. Thus, both activation functions demonstrate comparable performance in projection denoising; however, their different noise distributions lead to different effects of noise suppression in reconstruction. Swish utilizes information from more adjacent pixels in noise reduction; therefore, its noise is too structural to be handled by FBP. The noise correlation of LeakyReLU is lower and easier to process. Therefore, the image quality of the reconstructions from LeakyReLU is better than that of Swish in terms of noise suppression.

In summary, the proposed analytical noise propagation method is capable of providing a reasonable pixel-wise noise property estimation from only a single sample, whereas other noise estimation methods cannot present comparable performance under the same conditions. Our

Fig. 8 (Color online) Noise properties of reconstruction from the projection produced by U-net with LeakyReLU for the practical experiment. **a** The variance and profiles are marked by the red solid line, and **b** the covariance at the center of the reconstruction and the profiles are marked by the red solid line
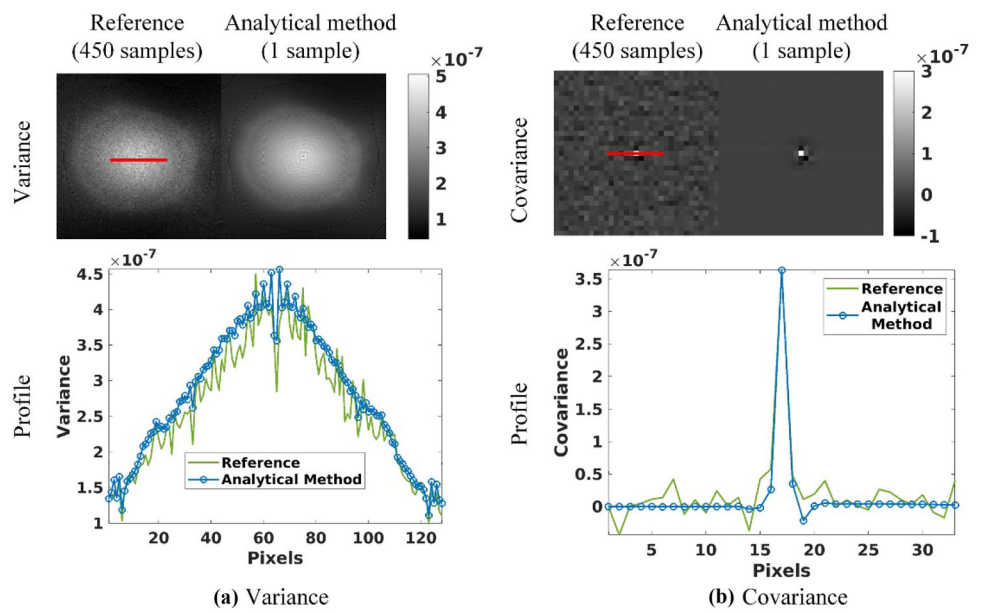

(a) Variance

(b) Covariance

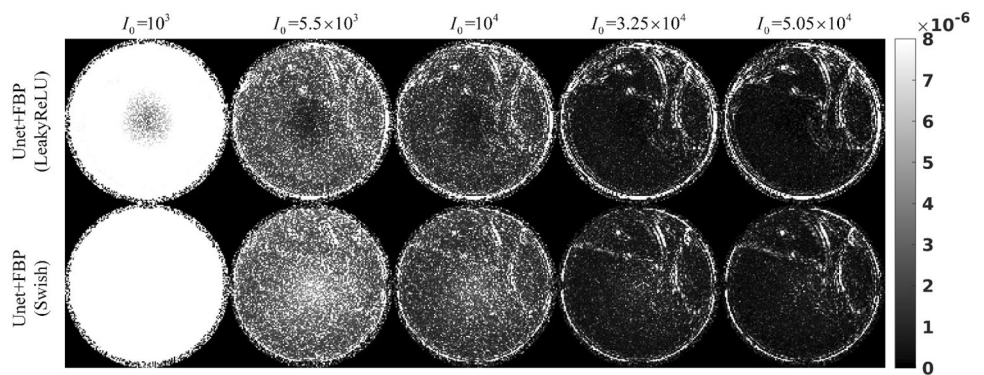**Fig. 9** GE maps of U-net with varying noise levels for test phantom B

Table 4 NRPs of U-net with varying noise levels. Point ① represents the low-attenuating point, and point ② indicates the high-attenuating point. Both points are marked by red dots in Fig. 2(b)

NRP (%)	Activation function	$I_0 = 10^3$	$I_0 = 5.5 \times 10^3$	$I_0 = 10^4$	$I_0 = 3.25 \times 10^4$	$I_0 = 5.05 \times 10^4$
Point ①	LeakyReLU	64.78	84.99	88.87	93.82	95.04
	Swish	55.66	81.16	86.02	92.25	93.78
Point ②	LeakyReLU	63.93	84.61	88.59	93.67	94.93
	Swish	60.42	83.11	87.48	93.05	94.43

proposed method can be applied to any inference-ready CNN with a fixed structure and weight for noise estimation. Because the convolution, batch normalization, average pooling, and up-sampling operations are all linear, and the nonlinear activation function is linearly approximated, the noise of the network input propagates linearly in the network. Evidently, the error in the noise estimation of the network output results from the linear approximation of the nonlinear activation function. Two activation functions,

LeakyReLU and Swish, are validated in this study; hence, the proposed method is applicable to any network with these two activation functions, regardless of the network structure. Moreover, the noise property estimation of the network output can be used to evaluate the performance of the reconstruction methods. We can characterize noise features based on pixel-by-pixel noise estimation, which also enables us to analyze the spatial correlation and structural properties of noise. The experimental results
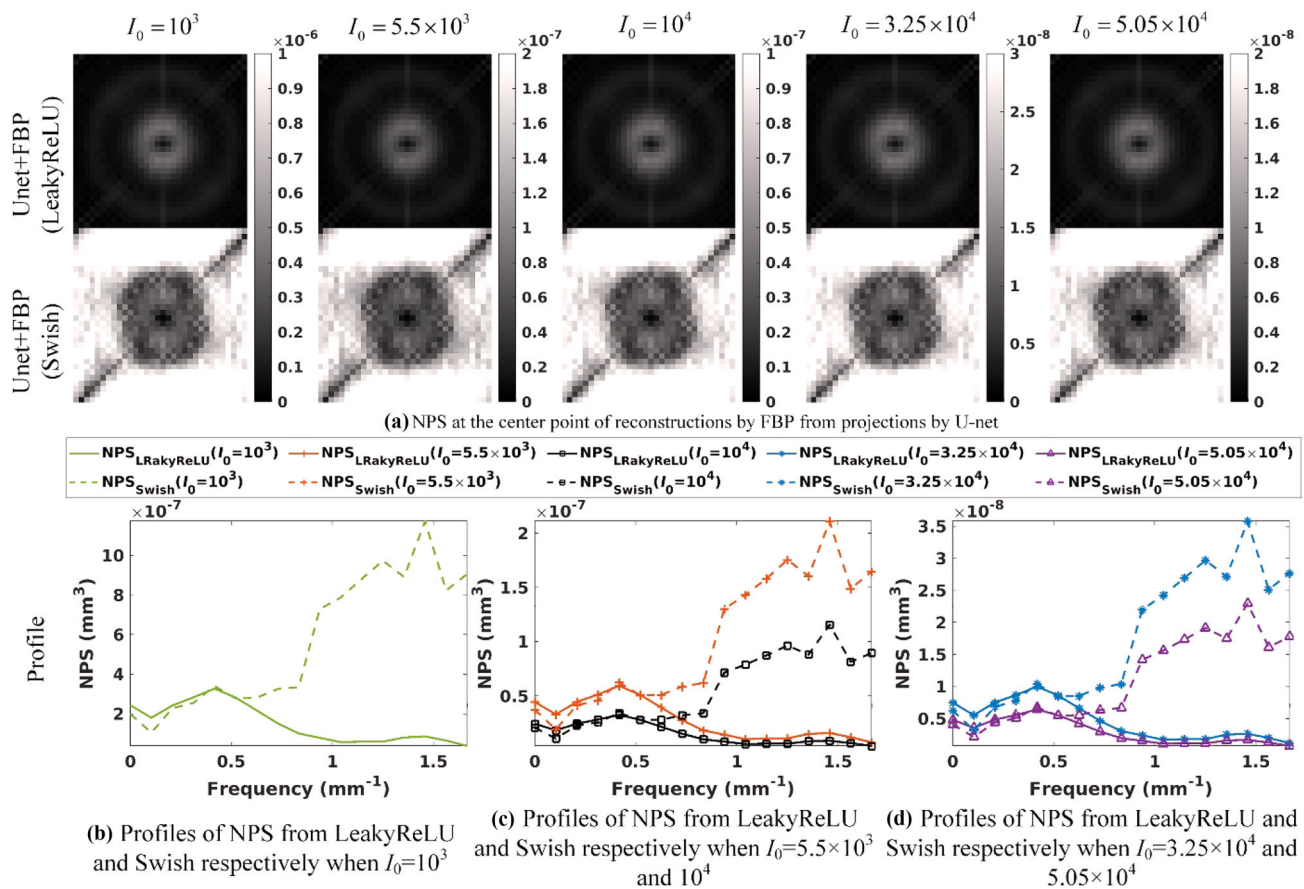
(a) NPS at the center point of reconstructions by FBP from projections by U-net

(b) Profiles of NPS from LeakyReLU and Swish respectively when $I_0 = 10^3$

(c) Profiles of NPS from LeakyReLU and Swish respectively when $I_0 = 5.5 \times 10^3$ and 10^4

(d) Profiles of NPS from LeakyReLU and Swish respectively when $I_0 = 3.25 \times 10^4$ and 5.05×10^4

Fig. 10 (Color online) NPS at the center of reconstructions varying with the number of incident photons. **a** NPS of reconstructions with varying noise levels. **b–d** NPS profiles of different noise levels: **b** NPS profiles at $I_0 = 10^3$, **c** NPS profiles at $I_0 = 5.5 \times 10^3$ and 10^4, and **d** NPS profiles at $I_0 = 3.25 \times 10^4$ and 5.05×10^4

reveal the significant value of this method in evaluating the output from CNN methods. In future studies, we aim to study the application of covariance estimation to a model observer for subjective image quality assessment. However, the computational cost is expected to increase with increasing network complexity and dimensions. Hence, efficient noise propagation methods for complex and high-dimensional networks must be studied.

References

1. M.H. Touch, D.P. Clark, W. Barber et al., A neural network-based method for spectral distortion correction in photon counting X-ray CT. Phys. Med. Biol. **61**(16), 6132–6153 (2016). https://doi.org/10.1088/0031-9155/61/16/6132

2. M.D. Holbrook, D.P. Clark, C.T. Badea, Deep learning based spectral distortion correction and decomposition for photon counting CT using calibration provided by an energy integrated detector, in *SPIE Medical Imaging 2021: Physics of Medical Imaging* (2021). https://doi.org/10.1117/12.2581124

3. K.C. Liang, L. Zhang, H.K. Yang et al., A model-based unsupervised deep learning method for low-dose CT reconstruction. IEEE Access **8**, 159260–159273 (2020). https://doi.org/10.1109/ACCESS.2020.3020406

4. Y.K. Zhang, D.L. Hu, Q.L. Zhao et al., CLEAR: comprehensive learning enabled adversarial reconstruction for subtle structure enhanced low-Dose CT imaging. IEEE Trans. Med. Imaging **40**(11), 3089–3101 (2021). https://doi.org/10.1109/TMI.2021.3097808

5. H.K. Yang, K.C. Liang, K.J. Kang et al., Slice-wise reconstruction for low-dose cone-beam CT using a deep residual convolutional neural network. Nucl. Sci. Tech. **30**, 59 (2019). https://doi.org/10.1007/s41365-019-0581-7

6. X.R. Yin, Q.L. Zhao, J. Liu et al., Domain progressive 3D residual convolution network to improve low-dose CT imaging. IEEE Trans. Med. Imaging **38**(12), 2903–2913 (2019). https://doi.org/10.1109/TMI.2019.2917258

7. J. Liu, Y. Zhang, Q.L. Zhao et al., Deep iterative reconstruction estimation (DIRE): approximate iterative reconstruction estimation for low dose CT imaging. Phys. Med. Biol. **64**(13), 135007 (2019). https://doi.org/10.1088/1361-6560/ab18db

8. D.L. Hu, J. Liu, T.L. Lv et al., Hybrid-domain neural network processing for sparse-view CT reconstruction. IEEE Trans. Radiat. Plasma. Med. Sci. **5**(1), 88–98 (2021). https://doi.org/10.1109/TRPMS.2020.3011413

9. D. Hu, Y. Zhang, J. Liu et al., DIOR: deep iterative optimization-based residual-learning for limited-angle CT reconstruction. IEEE Trans. Med. Imaging **41**, 1778–1790 (2022). https://doi.org/10.1109/TMI.2022.3148110

10. Y.J. Ma, Y. Ren, P. Feng et al., Sinogram denoising via attention residual dense convolutional neural network for low-dose computed tomography. Nucl. Sci. Tech. **32**, 41 (2021). https://doi.org/10.1007/s41365-021-00874-2

11. W. Fang, D.F. Wu, K. Kim et al., Iterative material decomposition for spectral CT using self-supervised Noise2Noise prior. Phys. Med. Biol. **66**(15), 1–17 (2021). https://doi.org/10.1088/1361-6560/ac0afd

12. T.L. Lyu, W. Zhao, Y.S. Zhu et al., Estimating dual-energy CT imaging from single-energy CT data with material decomposition convolutional neural network. Med. Image Anal. **70**, 102001 (2021). https://doi.org/10.1016/j.media.2021.102001

13. A. Zheng, H.K. Yang, L. Zhang et al., Interweaving network: a novel monochromatic image synthesis method for a photon-counting detector CT system. IEEE Access **8**, 217710 (2020). https://doi.org/10.1109/ACCESS.2020.3041078

14. K.C. Liang, L. Zhang, H.K. Yang et al., Metal artifact reduction for practical dental computed tomography by improving interpolation-based reconstruction with deep learning. Med. Phys. **46**(12), e823–e834 (2019). https://doi.org/10.1002/mp.13644

15. W. Fang, L. Li, Z.Q. Chen, Removing ring artefacts for photon-counting detectors using neural networks in different domains. IEEE Access **8**, 42447–42457 (2020). https://doi.org/10.1109/ACCESS.2020.2977096

16. P.J. Liu, M. Wang, Y.N. Wang et al., Impact of deep learning-based optimization algorithm on image quality of low-dose coronary CT angiography with noise reduction: a prospective study. Acad. Radiol. **27**(9), 1241–1248 (2020). https://doi.org/10.1016/j.acra.2019.11.010

17. A. Steuwe, M. Weber, O.T. Bethge et al., Influence of a novel deep-learning based reconstruction software on the objective and subjective image quality in low-dose abdominal computed tomography. Br. J. Radiol. **94**, 20200677 (2021). https://doi.org/10.1259/bjr.20200677

18. C. Park, K.S. Choo, Y. Jung et al., CT iterative vs deep learning reconstruction: comparison of noise and sharpness. Eur. Radiol. **31**(5), 3156–3164 (2021). https://doi.org/10.1007/s00330-020-07358-8

19. J. Greffier, A. Hamard, F. Pereira et al., Image quality and dose reduction opportunity of deep learning image reconstruction algorithm for CT: a phantom study. Eur. Radiol. **30**(7), 3951–3959 (2020). https://doi.org/10.1007/s00330-020-06724-w

20. C.T. Jensen, X.M. Liu, E.P. Tamm et al., Image quality assessment of abdominal CT by use of new deep learning image reconstruction: initial experience. AJR **215**(1), 50–57 (2020). https://doi.org/10.2214/ajr.19.22332

21. R. Singh, S.R. Digumarthy, V.V. Muse et al., Image quality and lesion detection on deep learning reconstruction and iterative reconstruction of submillisievert chest and abdominal CT. AJR **214**(3), 566–573 (2020). https://doi.org/10.2214/AJR.19.21809

22. X. He, S. Park, Model observers in medical imaging research. Theranostics **3**(10), 774–786 (2013). https://doi.org/10.7150/thno.5138

23. S. Leng, L.Y. Yu, Y. Zhang et al., Correlation between model observer and human observer performance in CT imaging when lesion location is uncertain. Med. Phys. **40**(8), 081908 (2013). https://doi.org/10.1118/1.4812430

24. L.Y. Yu, B.Y. Chen, J.M. Kofler et al., Correlation between a 2D channelized Hotelling observer and human observers in a low-contrast detection task with multislice reading in CT. Med. Phys. **44**(8), 3990–3999 (2017). https://doi.org/10.1002/mp.12380

25. G. Kim, M. Han, H. Shim et al., A convolutional neural network-based model observer for breast CT images. Med. Phys. **47**(4), 1619–1632 (2020). https://doi.org/10.1002/mp.14072

26. D. Piccini, R. Demesmaeker, J. Heerfordt et al., Deep learning to automate reference-free image quality assessment of whole-heart MR images. Radiol. Artif. Intell. **2**(3), e190123 (2020). https://doi.org/10.1148/ryai.2020190123

27. H. Gong, L.Y. Yu, S. Leng et al., A deep learning- and partial least square regression-based model observer for a low-contrast lesion detection task in CT. Med. Phys. **46**(5), 2052–2063 (2019). https://doi.org/10.1002/mp.13500

28. H. Gong, Q. Hu, A. Walther et al., Deep-learning-based model observer for a lung nodule detection task in computed tomography. J. Med. Imaging **7**(4), 042807 (2020). https://doi.org/10.1117/1.JMI.7.4.042807

29. H. Gong, J.G. Fletcher, J.P. Heiken et al., Deep-learning model observer for a low-contrast hepatic metastases localization task in computed tomography. Med. Phys. **49**(1), 70–83 (2021). https://doi.org/10.1002/mp.15362

30. A.H. Abdelaziz, S. Watanabe, J. Hershey et al., Uncertainty propagation through deep neural networks, in *InterSpeech* (2015). https://hal.inria.fr/hal-01162550

31. J. Lee, Y Bahri, R. Novak et al., Deep neural networks as Gaussian processes, in *the 6th International Conference on Learning Representations (ICRL 2018)* (2018). arXiv:1711.00165

32. R. Tanno, D.E. Worrall, E. Kaden et al., Uncertainty modelling in deep learning for safer neuroimage enhancement: demonstration in diffusion MRI. Neuroimage **225**, 117366 (2021). https://doi.org/10.1016/j.neuroimage.2020.117366

33. N. Ueda, R. Nakano, Generalization error of ensemble estimators, in *Proceedings of International Conference on Neural Networks (ICNN'96)* (1996), pp. 90–95. https://doi.org/10.1109/ICNN.1996.548872