Improved Low-Count Quantitative PET Reconstruction With an Iterative Neural Network

Hongki Lim, *Graduate Student Member, IEEE*, Il Yong Chun, *Member, IEEE*,
Yuni K. Dewaraja, *Member, IEEE*, and Jeffrey A. Fessler, *Fellow, IEEE*

Abstract—**Image reconstruction in low-count PET is particularly challenging because gammas from natural radioactivity in Lu-based crystals cause high random fractions that lower the measurement signal-to-noise-ratio (SNR). In model-based image reconstruction (MBIR), using more iterations of an unregularized method may increase the noise, so incorporating regularization into the image reconstruction is desirable to control the noise. New regularization methods based on learned convolutional operators are emerging in MBIR. We modify the architecture of an iterative neural network, *BCD-Net*, for PET MBIR, and demonstrate the efficacy of the trained BCD-Net using XCAT phantom data that simulates the low true coincidence count-rates with high random fractions typical for Y-90 PET patient imaging after Y-90 microsphere radioembolization. Numerical results show that the proposed BCD-Net significantly improves CNR and RMSE of the reconstructed images compared to MBIR methods using non-trained regularizers, total variation (TV) and non-local means (NLM). Moreover, BCD-Net successfully generalizes to test data that differs from the training data. Improvements were also demonstrated for the clinically relevant phantom measurement data where we used training and testing datasets having very different activity distributions and count-levels.**

Index Terms—**Iterative neural network, regularized model-based image reconstruction, low-count quantitative PET, Y-90.**

I. Introduction

IMAGE reconstruction in low-count PET is particularly challenging because dominant gammas from natural radioactivity in Lu-based crystals cause high random fractions, lowering the measurement signal-to-noise-ratio (SNR) [1].

To accurately reconstruct images in low-count PET, regularized model-based image reconstruction (MBIR) solves the following optimization problem consisting of *1*) a data fidelity term $f(x)$ that models the physical PET imaging system, and *2*) a regularization term $\mathsf{R}(x)$ that penalizes image roughness and controls noise [2]:

$$\hat{x} = \arg\min_{x \geq 0} f(x) + \mathsf{R}(x)$$
$$f(x) := \mathbf{1}^T(Ax + \bar{r}) - y^T \log(Ax + \bar{r}). \quad (1)$$

Here, $f(x)$ is the Poisson negative log-likelihood for measurement y and estimated measurement means $\bar{y}(x) = Ax + \bar{r}$, the matrix A denotes the system model, and $\bar{r}$ denotes the mean background events such as scatter and random coincidences. Recently, applying learned regularizers to $\mathsf{R}(x)$ is emerging for MBIR [3].

While there is much ongoing research on machine learning or deep-learning techniques applied to CT [4]–[8] and MRI [9]–[13] reconstruction problems, fewer studies have applied these techniques to PET. Most past PET studies used deep learning in image space without exploiting the physical imaging model in (1). For example, [14] applied a deep neural network (NN) mapping between reconstructed PET images with normal dose and reduced dose and [15] applied a multi-layer perceptron mapping between reconstructed images using maximum a posteriori algorithm and a reference (true) image, and their framework uses the acquisition data only to form the initial image. Recently, [16] trained a NN to reconstruct a 2D image directly from PET sinogram and [17], [18] proposed a PET MBIR framework using a deep-learning based regularizer. Our proposed MBIR framework, *BCD-Net*, also uses a regularizer that penalizes differences between the unknown image and "denoised" images given by a regression neural network in an iterative manner. In particular, whereas [17], [18] trained only a single image denoising NN, the proposed method is an iterative framework that includes multiple trained NNs. This iterative framework enables the NNs in the later stages to learn how to recover fine details. Our proposed BCD-Net also differs from [17], [18] in that our denoising NNs are defined by an optimization formulation with a mathematical motivation (whereas, for the trained regularizer, [17], [18] brought U-Net [19] and DnCNN that were [20] developed for medical image segmentation and general Gaussian denoising,

respectively) and characterized by fewer parameters, thereby avoiding over-fitting and generalizing well to unseen data especially when training samples are limited.

Iterative NNs [8]–[11], [21]–[24] are a broad family of methods that originate from an unrolling algorithm for solving an optimization problem and BCD-Net [25] is a specific example of an iterative NN. BCD-Net is constructed by unfolding a block coordinate descent (BCD) MBIR algorithm using "learned" convolutional analysis operators [26]–[28], leading to significantly improved image recovery accuracy in extreme imaging applications, e.g., low-dose CT [29], dual-energy CT [30], highly undersampled MRI [25], denoising low-SNR images [25], etc. A preliminary version of this paper was presented at the 2018 Nuclear Science Symposium and Medical Imaging Conference [31]. We significantly extended this work by applying our proposed method to measured PET data with newly developed techniques. We also added detailed analysis of our proposed method as well as comparisons to related works.

To show the efficacy of our proposed BCD-Net method in low-count PET imaging, we performed both digital phantom simulation and experimental measurement studies with activity distributions and count-rates that are relevant to clinical Y-90 PET imaging after liver radioembolization. Novel therapeutic applications have sparked growing interest in quantitative imaging of Y-90, an almost pure beta emitter that is widely used in internal radionuclide therapy. In addition to the FDA approved Y-90 microsphere radioembolization and Y-90 ibritumomab radioimmunotherapy, there are 50 active clinical trials for Y-90 labeled therapies (www.clinicaltrials.gov). However, the lack of gamma photons complicates imaging of Y-90; it involves SPECT via bremsstrahlung photons produced by the betas [32] or PET via a very low abundance positron in the presence of bremsstrahlung that leads to low signal-to-noise [33]. This paper applies a BCD-Net that is trained for realistic low-count PET imaging environments and compares its performance with those of non-trained regularizers. Our proposed BCD-Net applies to PET imaging in general, particularly in other imaging situations that also have low counts. Using shorter scan times and lower tracer activity in diagnostic PET has cost benefits and reduces radiation exposure, but at the expense of reduced counts that makes traditional iterative reconstruction challenging.

Section II develops the proposed BCD-Net architecture for PET MBIR. Section II also explains the simulation studies in the setting of Y-90 radioembolization and provides details on how we perform the physical phantom measurement. Section III presents how the different reconstruction methods perform with the simulation and measurement data. Section IV discusses what training and imaging factors most affect generalization performance of BCD-Net. Section V concludes with future works.

II. METHODS

This section presents the problem formulation of the BCD-Net and gives a detailed derivation that inspires the final form of BCD-Net. We also provide several techniques for BCD-Net that we specifically devised for PET data

where each measurement has different count-level (and noise-level). Then we review the related works that we compare with BCD-Net such as MBIR methods using conventional non-trained regularizers. This section also describes the simulation setting and details on the measurement data and what evaluation metrics are used to assess the efficacy of each reconstruction algorithm.

A. BCD Algorithm for MBIR Using "Learned" Convolutional Regularization

Conventional PET regularizers penalize differences between neighboring pixels [34]. That approach is equivalent to assuming that convolving the image with the [1,-1] finite difference filter along different directions produces sparse outputs. Using such "hand-crafted" filters is unlikely to be the best approach. A recent trend is to use training data to learn filters c_k that produce sparse outputs when convolved with images of interest [26], [27], [35], [36]. Such learned filters can be used to define a regularizer that prefers images having sparse outputs, as follows [37]:

$$\mathsf{R}(\boldsymbol{x}) = \min_{\{z_k\}} \beta \left(\sum_{k=1}^{K} \frac{1}{2} \|\boldsymbol{c}_k * \boldsymbol{x} - z_k\|_2^2 + \alpha_k \|z_k\|_1 \right), \quad (2)$$

where β is regularization parameter, $\{\boldsymbol{c}_k \in \mathbb{R}^R : k = 1, \ldots, K\}$ is a set of convolutional filters, $\{z_k \in \mathbb{R}^{n_p} : k = 1, \ldots, K\}$ is a set of sparse codes, $\{\alpha_k \in \mathbb{R} : k = 1, \ldots, K\}$ is a set of thresholding parameters controlling the sparsity of $\{z_k\}$, n_p is the number of image voxels, and R and K denote the size and number of learned filters, respectively. BCD-Net is inspired by this type of "learned" regularizer. Ultimately, we hope that the learned regularizer can better separate true signal from noisy components compared to hand-crafted filters [29].

A natural BCD algorithm solves (1) with regularizer (2) by alternatively updating $\{z_k\}$ and $\boldsymbol{x}$:

$$\{z_k^{(n+1)}\} = \arg\min_{\{z_k\}} \frac{1}{2} \|\boldsymbol{c}_k * \boldsymbol{x}^{(n)} - z_k\|_2^2 + \alpha_k \|z_k\|_1$$

$$= \mathcal{T}(\boldsymbol{c}_k * \boldsymbol{x}^{(n)}, \alpha_k) \quad (3)$$

$$\boldsymbol{x}^{(n+1)} = \arg\min_{\boldsymbol{x} \geq 0} f(\boldsymbol{x}) + \frac{\beta}{2} \left(\sum_{k=1}^{K} \left\| \boldsymbol{c}_k * \boldsymbol{x} - z_k^{(n+1)} \right\|_2^2 \right),$$

$$(4)$$

where $\mathcal{T}(\cdot, \cdot)$ is the element-wise soft thresholding operator: $\mathcal{T}(\boldsymbol{t}, q)_j := \text{sign}(t_j) \max(|t_j| - q, 0)$.

Assuming that learned filters $\{\boldsymbol{c}_k\}$ satisfy the tight-frame condition, $\sum_{k=1}^{K} \|\boldsymbol{c}_k * \boldsymbol{x}\|_2^2 = \|\boldsymbol{x}\|_2^2 \ \forall \boldsymbol{x}$ [26], we rewrite the updates in (3)-(4) as follows:

$$\boldsymbol{u}^{(n+1)} = \sum_{k=1}^{K} \tilde{\boldsymbol{c}}_k * \left(\mathcal{T}\left(\boldsymbol{c}_k * \boldsymbol{x}^{(n)}, \alpha_k \right) \right) \quad (5)$$

$$\boldsymbol{x}^{(n+1)} = \arg\min_{\boldsymbol{x} \geq 0} f(\boldsymbol{x}) + \frac{\beta}{2} \left\| \boldsymbol{x} - \boldsymbol{u}^{(n+1)} \right\|_2^2, \quad (6)$$

where $\tilde{\boldsymbol{c}}_k$ denotes a rotated version of $\boldsymbol{c}_k$. The operations of convolution, soft thresholding and then filtering again with summation typically have the effect of denoising the image $\boldsymbol{x}^{(n)}$.

Algorithm 1 BCD-Net for PET MBIR

Require:
$\{c_k^{(n)}, d_k^{(n)}, \alpha_k^{(n)} : n = 1, \ldots, T\}, y, \bar{r}, A, c$
Initialize:
$x^{(0)}$ using EM algorithm
Calculate $a_j = \sum_{i=1}^{n_d} a_{ij}$
for $n = 0, \ldots, T-1$ **do**
$\quad u^{(n+1)} = \sum_{k=1}^{K} d_k^{(n+1)} * \left(\mathcal{T}\left(c_k^{(n+1)} * g_1\left(x^{(n)}\right), \alpha_k^{(n+1)} \right) \right)$
$\quad \beta^{(n+1)} = \dfrac{\left\| a_j - \sum_{i=1}^{n_d} a_{ij} \frac{y_i}{\bar{y}_i(x^{(n)})} \right\|_2}{\left\| x^{(n)} - g_2(u^{(n+1)}) \right\|_2} \cdot c$
$\quad$**for** $n' = 0, \ldots, T'-1$ **do**
$\quad\quad \lambda = \frac{1}{2}\left(a_j - \beta^{(n+1)} g_2\left(u_j^{(n+1)}\right) \right)$
$\quad\quad v = x_j^{(n')}\left(\sum_{i=1}^{n_d} a_{ij} \frac{y_i}{\bar{y}_i(x^{(n')})} \right)$
$\quad\quad x_j^{(n'+1)} = \begin{cases} \dfrac{\sqrt{\lambda^2 + \beta^{(n+1)}v} - \lambda}{\beta^{(n+1)}}, & \lambda < 0 \\[2ex] \dfrac{v}{\sqrt{\lambda^2 + \beta^{(n+1)}v} + \lambda}, & \lambda \geq 0 \end{cases}$
$\quad$**end for**
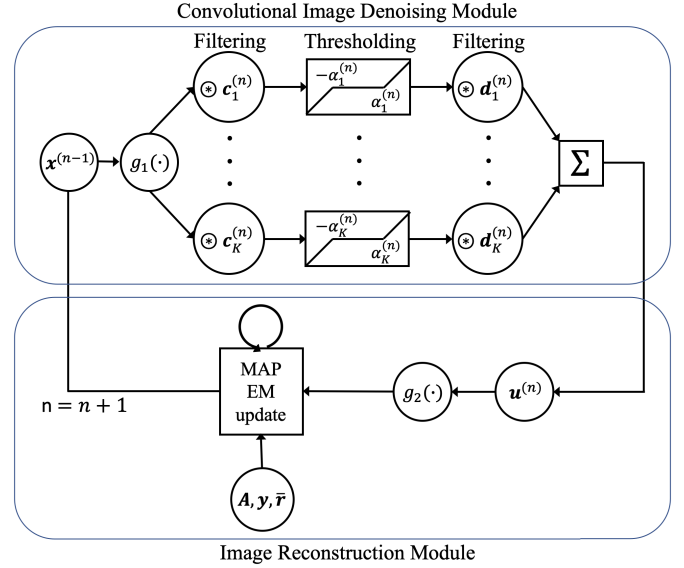$\quad x^{(n+1)} = x^{(T')}$
end for

Fig. 1. Architecture of the proposed BCD-Net for PET. The proposed BCD-Net has an iterative NN architecture: each BCD-Net iteration uses three inputs – fixed measurement and mean background $\{y, \bar{r}\}$, and the image $x^{(n-1)}$ reconstructed at the previous BCD-Net iteration – and provides the reconstructed image $x^{(n)}$. A circular arrow above MAP EM update indicates inner iterations. $g_1(\cdot)$ and $g_2(\cdot)$ are the normalization and scaling functions described in Section II-C.

For efficient image reconstruction (6) in PET, we use the standard EM-surrogate of Poisson log-likelihood function [38]:

$$f(x) + \frac{\beta}{2}\left\| x - u^{(n+1)} \right\|_2^2$$

$$= \sum_{i=1}^{n_d} [Ax]_i + \bar{r}_i - y_i \log([Ax]_i + \bar{r}_i) + \frac{\beta}{2}\sum_{j=1}^{n_p}(x_j - u_j^{(n+1)})^2$$

$$\leq \sum_{j=1}^{n_p}\left\{ -e_j(x^{(n')})(x_j^{(n')})\log(x_j) + a_j x_j + \frac{\beta}{2}(x_j - u_j^{(n+1)})^2 \right\}$$

$$= \sum_{j=1}^{n_p} Q_j(x_j)$$

where n' denotes n'th inner-iteration in (6), $e_j(x^{(n')}) = \sum_{i=1}^{n_d} a_{ij}\frac{y_i}{\bar{y}_i(x^{(n')})}$, a_{ij} denotes an element of the system model at ith row and jth column, and n_d is the number of rays. Equating $\frac{\partial Q_j(x_j)}{\partial x_j}$ to zero is equivalent to finding the root of the following quadratic formula:

$$\beta x_j^2 + \left(a_j - \beta u_j^{(n+1)} \right) x_j - e_j(x^{(n')}) x_j^{(n')} = 0,$$

and finding the root [39] leads to the minimizer:

$$x_j^{(n'+1)} = \begin{cases} \dfrac{\sqrt{\lambda^2 + \beta v} - \lambda}{\beta}, & \lambda < 0 \\[2ex] \dfrac{v}{\sqrt{\lambda^2 + \beta v} + \lambda}, & \lambda \geq 0, \end{cases}$$

where $\lambda = \frac{1}{2}(a_j - \beta u_j^{(n+1)})$, $v = e_j(x^{(n')})x_j^{(n')}$, $a_j = \sum_{i=1}^{n_d} a_{ij}$.

B. BCD-Net for PET MBIR and Training Its Denoising Module

To further improve denoising capability by providing more trainable parameters, we extend the convolutional image

denoiser (CID) in (5) [25], by replacing $\{\tilde{c}_k\}$ with separate decoding filters $\{d_k\}$. We *define* BCD-Net to use the following updates for each iteration:

$$u^{(n+1)} = \sum_{k=1}^{K} d_k^{(n+1)} * \left(\mathcal{T}\left(c_k^{(n+1)} * x^{(n)}, \alpha_k^{(n+1)} \right) \right) \quad (7)$$

$$x^{(n+1)} = \arg\min_{x \geq 0} f(x) + \frac{\beta}{2}\left\| x - u^{(n+1)} \right\|_2^2, \quad (8)$$

where separate encoding and decoding filters $\{c_k\}$ and $\{d_k\}$ are learned for each iteration. Fig. 1 shows the corresponding BCD-Net architecture. We refer to the u and x updates in (7)-(8) as two *modules*: 1) image denoising module and 2) image reconstruction module. The final output image is from the reconstruction module.

The image denoising module consists of encoding and decoding filters $\{c_k^{(n+1)}\}, \{d_k^{(n+1)}\}$ and thresholding values $\{\alpha_k^{(n+1)}\}$. We train these parameters to "best map" from noisy images into high-quality reference images (e.g., true images if available) in the sense of mean squared error:

$$\arg\min_{\{c_k\},\{d_k\},\{\alpha_k\}} \sum_{l=1}^{L} \left\| x_{\text{true},l} - \sum_{k=1}^{K} d_k * \left(\mathcal{T}\left(c_k * x_l^{(n)}, \alpha_k \right) \right) \right\|_2^2, \quad (9)$$

where L is the total number of training samples, $\{x_{\text{true},l} \in \mathbb{R}^{n_p} : l = 1, \ldots, L\}$ is a set of true images and $\{x_l^{(n)} \in \mathbb{R}^{n_p} : l = 1, \ldots, L\}$ is a set of images estimated by image reconstruction module in the nth iteration. We train the set of filters and thresholding values iteration-by-iteration and do not include the system matrix or sinograms for training as shown in (9). Moreover, we do not enforce the tight-frame condition when training the filters.

One can further extend the CID in (7) to a general regression NN, e.g., a deep U-Net [19]. We investigated if the iterative BCD-Net combined with U-Net denoisers (by replacing the denoising module in (7) with a U-Net) performs better than the proposed BCD-Net using CID (7). Section II-G.2 gives the details of the U-Net implementation.

C. Adaptive BCD-Net Generalizing to Various Count-Levels

1) Normalization and Scaling Scheme: Different PET images can have very different intensity values due to variations in scan time and activity, and it is important for trained methods to be able to generalize to a wide range of count levels. Towards this end, we implemented normalization and scaling techniques in BCD-Net. Reference [18] extended [17] by implementing "local linear fitting" to ensure that the denoising NN output has similar intensity as the input patch from the current estimated image. Our approach is different in that we normalize and scale the image with a global approach, not a patch-based approach. In particular, we modify the architecture in (7)-(8) as:

$$u^{(n+1)} = \sum_{k=1}^{K} d_k^{(n+1)} * \left(\mathcal{T} \left(c_k^{(n+1)} * g_1(x^{(n)}), a_k^{(n+1)} \right) \right) \quad (10)$$

$$x^{(n+1)} = \arg\min_{x \geq 0} f(x) + \frac{\beta}{2} \left\| x - g_2(u^{(n+1)}) \right\|_2^2, \quad (11)$$

where the normalization function $g_1(\cdot)$ is defined by $g_1(v) := \frac{1}{\sum_j v_j} v$ to ensure that $\mathbf{1}^T g_1(v) = 1$, and the scaling function $g_2(\cdot)$ is defined by $g_2(v) := \{\arg\min_s f(s \cdot v)\} v$. We solve the optimization problem over s using Newton's method:

$$
\begin{aligned}
s^{(n+1)} &= s^{(n)} - \frac{\nabla_s f(s^{(n)} \cdot v)}{\nabla_s^2 f(s^{(n)} \cdot v)} \\
&= s^{(n)} - \frac{\sum_{i=1}^{n_d} [Av]_i - y_i \frac{[Av]_i}{s^{(n)}[Av]_i + \bar{r}_i}}{\sum_{i=1}^{n_d} y_i \left(\frac{[Av]_i}{(s^{(n)}[Av]_i + \bar{r}_i)} \right)^2}.
\end{aligned} \quad (12)
$$

To be consistent with the modified CID in (10), we also apply this image-based normalization technique when training the convolutional filters and thresholding values:

$$\arg\min_{\{c_k\},\{d_k\},\{a_k\}} \sum_{l=1}^{L} \left\| g_1(x_{\text{true},l}) - \sum_{k=1}^{K} d_k * \left(\mathcal{T} \left(c_k * g_1(x_l^{(n)}), a_k \right) \right) \right\|_2^2.$$

2) Adaptive Regularization Parameter Scheme: The best regularization parameter value can also vary greatly between scans, depending on the count level. Therefore, instead of choosing one specific value for the regularization parameter, we set the β value for each iteration based on evaluation on current gradients of data-fidelity term and regularization term:

$$
\begin{aligned}
\beta^{(n+1)} &= \frac{\left\| \nabla_x f(x^{(n)}) \right\|_2}{\left\| \nabla_x \mathsf{R}(x^{(n)}) \right\|_2} \cdot c \\
&= \frac{\left\| a_j - e_j(x^{(n)}) \right\|_2}{\left\| x^{(n)} - g_2(u^{(n+1)}) \right\|_2} \cdot c, \quad n = 0, \ldots, T-1,
\end{aligned} \quad (13)
$$

where c is a constant specifying how we balance between the data-fidelity term and regularization term and n denotes nth outer-iteration. Algorithm (1) gives detailed pseudocode of the proposed method. T denotes the total number of outer-iterations and T' denotes the number of inner iterations used for (8). We use $x^{(n)}$ as the initial image when solving (11).

D. Conventional MBIR Methods: Non-Trained Regularizers

We compared the proposed BCD-Net with two MBIR methods that use standard non-trained regularizers.

1) Total-Variation (TV): TV regularization penalizes the sum of absolute value of differences between adjacent voxels:

$$\mathsf{R}(x) = \beta \left\| Cx \right\|_1,$$

where C is finite differencing matrix. Recent work [40] applied Primal-Dual Hybrid Gradient (PDHG) [41] for PET MBIR using TV regularization and demonstrated that PDHG-TV is superior than clinical reconstruction (e.g., OS-EM) for low-count datasets in terms of several image quality evaluation metrics such as contrast recovery and variability.

2) Non-Local Means (NLM): NLM regularization penalizes the differences between nearby patches in image:

$$\mathsf{R}(x) = \beta \sum_{i,j \in S_i} p \left(\left\| N_i x - N_j x \right\|_2^2 \right),$$

where $p(t)$ is a potential function of a scalar variable t, S_i is the search neighborhood around the ith voxel, and N_i is a patch extraction operator at the ith voxel. We used the Fair potential function for $p(t)$:

$$p(t) = \sigma_f^2 \left(\sqrt{\frac{t}{\sigma_f^2 N_f}} + \log \left(1 + \sqrt{\frac{t}{\sigma_f^2 N_f}} \right) \right),$$

where σ_f is a design parameter and N_f is the number of voxels in the patch $N_i x$. Unlike conventional local filters that assume similarity between only adjacent voxels, NLM filters can average image intensities over distant voxels. As in [42], we used ADMM to accelerate algorithmic convergence with an adaptive penalty parameter selection method [43].

E. Experimental Setup: Digital Phantom Simulation and Experimental Measurement

1) Y-90 PET/CT XCAT Simulations: We used the XCAT [44] phantom (Fig. 2) to simulate Y-90 PET following radioembolization. We set the image size to $128 \times 128 \times 100$ with a voxel size $4.0 \times 4.0 \times 4.0$ (mm^3) and chose 100 slices ranging from lung to liver. To simulate extremely low count scans with high random fractions, typical for Y-90 PET, we set total true coincidences and random fractions based on numbers from patient PET imaging performed after radioembolization [45]. To test the generalization capability of the trained BCD-Net, we changed all imaging factors between training and testing dataset. Here, imaging factors include activity distribution (shape and size of tumor and liver background, concentration ratio between hot and warm region) and count-level (total true coincidences and random fraction). Fig. 2 and Table I provide

TABLE I

DETAILS ON XCAT SIMULATION DATA: VARIATIONS BETWEEN TRAINING AND TESTING DATA

	Training data	Testing data
Concentration ratio (hot:warm)	9:1	4:1
Total net trues	200 K	500 K
Random fraction (%)	90.9	87.5

TABLE II

DETAILS ON PHANTOM MEASUREMENT DATA: ACTIVITY CONCENTRATION RATIO BETWEEN HOT AND WARM REGIONS AND RANDOMS FRACTIONS FOR TWO PHANTOM STUDIES

	Sphere	Liver-torso
Total activity (GBq)	0.65	1.9
Concentration ratio (hot:warm)	8.9:1	5.4:1
Total prompts	3.2 - 6.3 M	2.3 M
Total randoms	2.9 - 5.7 M	2.1 M
Total net trues	308 - 599 K	220 K
Random fraction(%)	90.3 - 90.5	90.7

TABLE III

DETAILS ON TYPICAL PATIENT MEASUREMENT DATA: TOTAL TRUES AND RANDOMS FRACTIONS

	Patient A
Total activity (GBq)	2.55
Total prompts	2.7 M
Total randoms	2.3 M
Total net trues	380 K
Random fraction(%)	85.8

details on how we changed the testing dataset from the training dataset. We trained BCD-Net using five pairs ($L = 5$) of 3D true images and estimated images at each iteration (1 true image, 5 realizations). We generated multiple (5) realizations to train the denoising NN to deal with the Poisson noise. We also generated 5 realizations (1 true image, 5 realizations) as a testing dataset to evaluate the noise across realizations.

2) Y90 PET/CT Physical Phantom Measurements and Patient Scan: For training BCD-Net, we used PET measurements of a sphere phantom (Fig. 4) where six 'hot' spheres (2,4,8,16,30 and 113 mL, 0.5 MBq/ml) are placed in a 'warm' background (0.057 MBq/ml) with total activity of 0.65 GBq. The phantom was scanned for 40 (3 acquisitions) - 80 (1 acquisition) ($L = 4$) minutes on a Siemens Biograph mCT PET/CT. For testing BCD-Net and other reconstruction algorithms, we used an anthropomorphic liver/lung torso phantom (Fig. 4) with total activity and distribution that is clinically realistic for imaging following radioembolization with Y-90 microspheres: 5% lung shunt, 1.17 MBq/mL in liver, 3 hepatic lesions (4 and 16 mL spheres, 29 mL ovoid) of 6.6 MBq/ml. The phantom with total activity of 1.9 GBq was scanned 5 times (each 30 minutes) on a Siemens Biograph mCT PET/CT. Fig. 4 and Table II provide details on the count-level (random fraction) and activity distribution differences between training (sphere phantom) and testing (liver phantom) dataset. We also tested BCD-Net with an actual Y-90 patient scan and Table III provides count-level information.

We acquired all measurement data with time of flight TOF information. The measurement data size is $200 \times 168 \times 621 \times 13$. The last dimension of measurement indicates the number of time bin. The reconstructed image size

is $200 \times 200 \times 112$ with a voxel size $4.07 \times 4.07 \times 2.03$ (mm^3). To reconstruct the image with measurement data, we used a SIEMENS TOF system model (A in (1)) along with manufacturer given attenuation/normalization correction, PSF modelling, and randoms/scatters estimation.

F. Evaluation Metrics

For the XCAT phantom simulation, we evaluated each reconstruction with contrast recovery (CR) (volume-of-interest (VOI): cold spot indicated in Fig. 2), noise across realizations, root mean squared error (RMSE), and contrast to noise ratio (CNR). For the physical phantom measurement, we used CR (VOI: hot spheres) and CNR averaged over multiple hot spheres. We define each VOI's mask based on attenuation map interpolated to PET voxel size. For the patient measurement, we used CNR and the field of view (FOV) activity bias since the total activity in FOV is known (equal to the injected activity because the microspheres are trapped) whearaes the activity distribution is unknown:

CR (VOI: cold spot)

$$= \left(1 - \frac{C_{\text{VOI}}}{C_{\text{BKG}}}\right) \times 100 \ (\%)$$

CR (VOI: hot sphere)

$$= \frac{\frac{C_{\text{VOI}}}{C_{\text{BKG}}} - 1}{R_{\text{True}} - 1} \times 100 \ (\%)$$

Noise

$$= \frac{\sqrt{\frac{1}{J_{\text{Liver}}} \sum_{j \in \text{Liver}} \left(\frac{1}{M-1} \sum_{m=1}^{M} (\hat{x}_m[j] - \frac{1}{M} \sum_{m'=1}^{M} \hat{x}_{m'}[j])^2\right)}}{\frac{1}{J_{\text{Liver}}} \sum_{j \in \text{Liver}} x_{\text{true}}[j]}$$

$$\times 100 \ \%$$

RMSE

$$= \sqrt{\frac{\sum_j (x_{\text{true}}[j] - \hat{x}[j])^2}{J_{\text{FOV}}}} \times 100 \ (\%)$$

CNR

$$= \frac{C_{\text{Lesion}} - C_{\text{BKG}}}{\text{STD}_{\text{BKG}}}$$

FOV bias

$$= \frac{\sum_j \hat{x}[j] - x_{\text{true}}[j]}{\sum_j x_{\text{true}}[j]} \times 100 \ (\%),$$

where C_{VOI} is mean counts in the VOI, R_{True} is true ratio between hot and warm region, $x[j]$ denotes the jth voxel of an image x, M is the number of realizations ($M = 5$ in both XCAT phantom simulation and physical phantom measurement) and J_{Liver} is the number of voxels in the volume of liver, STD_{BKG} is standard deviation between voxel values in uniform background liver (indicated in Fig. 2), and J_{FOV} is the total number of voxels in the FOV. As the background region when calculating the patient CNR, we used a part of liver region that has relatively uniform activity distribution.

G. Training Details

We trained the denoising network in each iteration with a stochastic gradient descent method using the PyTorch [46] deep-learning library.
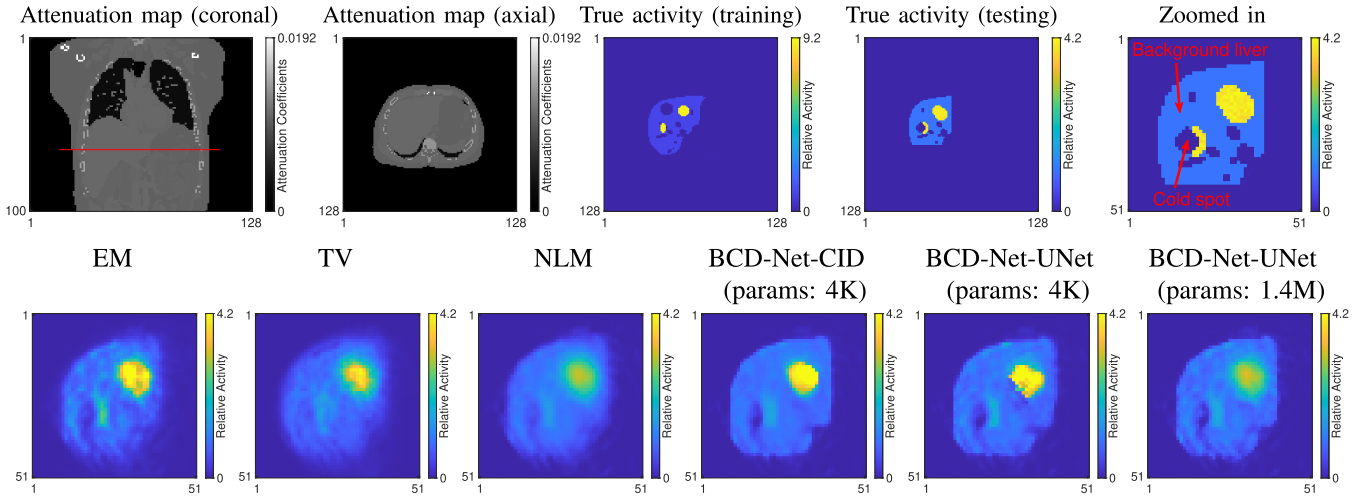
Fig. 2. XCAT phantom simulation: (First row) coronal and axial view of attenuation map and true relative activity distribution corresponding to axial attenuation map. (Second row) reconstructed images of one slice from different reconstruction methods. BCD-Net-CID/UNet is the BCD-Net with CID/UNet and params indicates the number of trainable parameters.
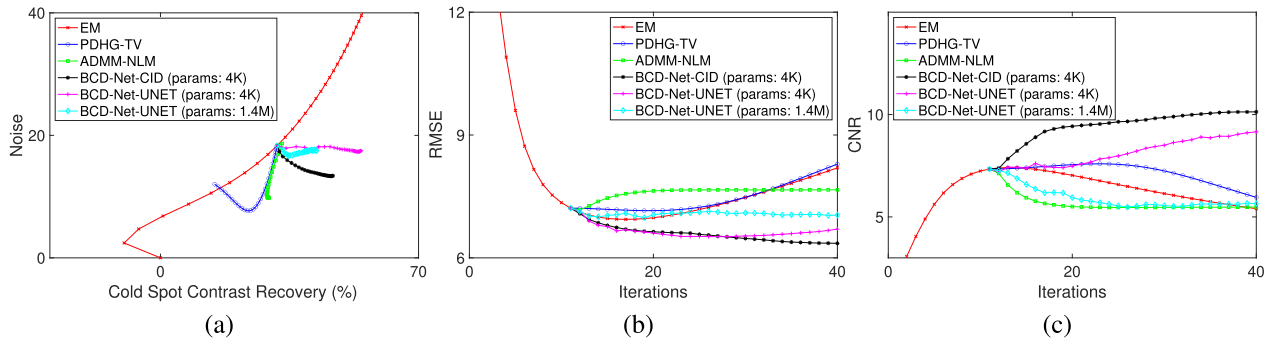

Fig. 3. (a) Plot of noise in background liver vs contrast recovery in cold spot. (b) RMSE vs iteration. (c) Contrast to noise ratio vs iteration. We initialized regularized methods with the 10th iterate of EM reconstruction.

1) BCD-Net With CID: We trained a set of CID for the denoising module in BCD-Net where each iteration has 78 sets of thresholding values and convolutional encoding/decoding filters ($K = 78$). We set the size of each filter as $3 \times 3 \times 3 (R = 3^3)$, and set the initial thresholding values by sorting the initial estimate of image and getting a 10% largest value of sorted initial image. We used the Adam optimization method [47] to train the NN. We applied the learning rate decay scheme. Due to the large size of 3D input, we set the batch size as 1.

2) BCD-Net With U-Net: We implemented a 3-D version of U-Net by modifying a shared code[1] (implemented for denoising 2-D MRI images) for fastMRI challenge [48]. We used a batch normalization layer instead of the instance normalization layer used in the baseline code. The 'encoder' part of U-Net consists of multiple sets of 1) max pooling layer, 2) $3 \times 3 \times 3$ convolutional layer, 3) batch normalization (BN) layer, 4) ReLU layer and the 'decoder' part of U-Net consists of multiple sets of 1) upsampling with trilinear interpolation [17], 2) $3 \times 3 \times 3$ convolutional layer, 3) BN layer, 4) ReLU layer. For training the U-Net, we used the same training dataset that we used for training the CID. We also used the Adam optimization method and identical settings (number of

[1]https://github.com/facebookresearch/fastMRI

epochs, learning rate decay, batch size) as those of the CID. We trained and tested two different U-Nets sizes. At each BCD-Net iteration, the U-Net has either about 4 K (similar size to the CID) or 1.4 M trainable parameters. We set the number of convolutional filter channels of the first encoder layer as 12 with 4 times of contraction/expansion for the U-Net with 1.4 M parameters and 5 with 1 time of contraction/expansion for the U-Net with 4 K parameters.

III. RESULTS

A. Reconstruction Setup

We compared the proposed BCD-Net method to the standard EM (1 subset), TV-based MBIR with PDHG algorithm (PDHG-TV), and NLM-based MBIR with ADMM algorithm (ADMM-NLM). For regularized MBIR methods including BCD-Net, we used 10 EM algorithm iterations to get the initial image $x^{(0)}$. For each regularization method, we finely tuned the regularization parameter β (within range $[2^{-15}, 2^{15}]$) by considering the recovery accuracy and noise. For NLM, we additionally tuned the window and search sizes. For the XCAT simulation data, we used 40 iterations for EM and 30 iterations ($T = 30$) for PDHG-TV, ADMM-NLM, and BCD-Net. We used 1 inner-iteration ($T' = 1$) for the reconstruction module (11) for each outer-iteration of BCD-Net. For the measured data, we used 20 iterations for
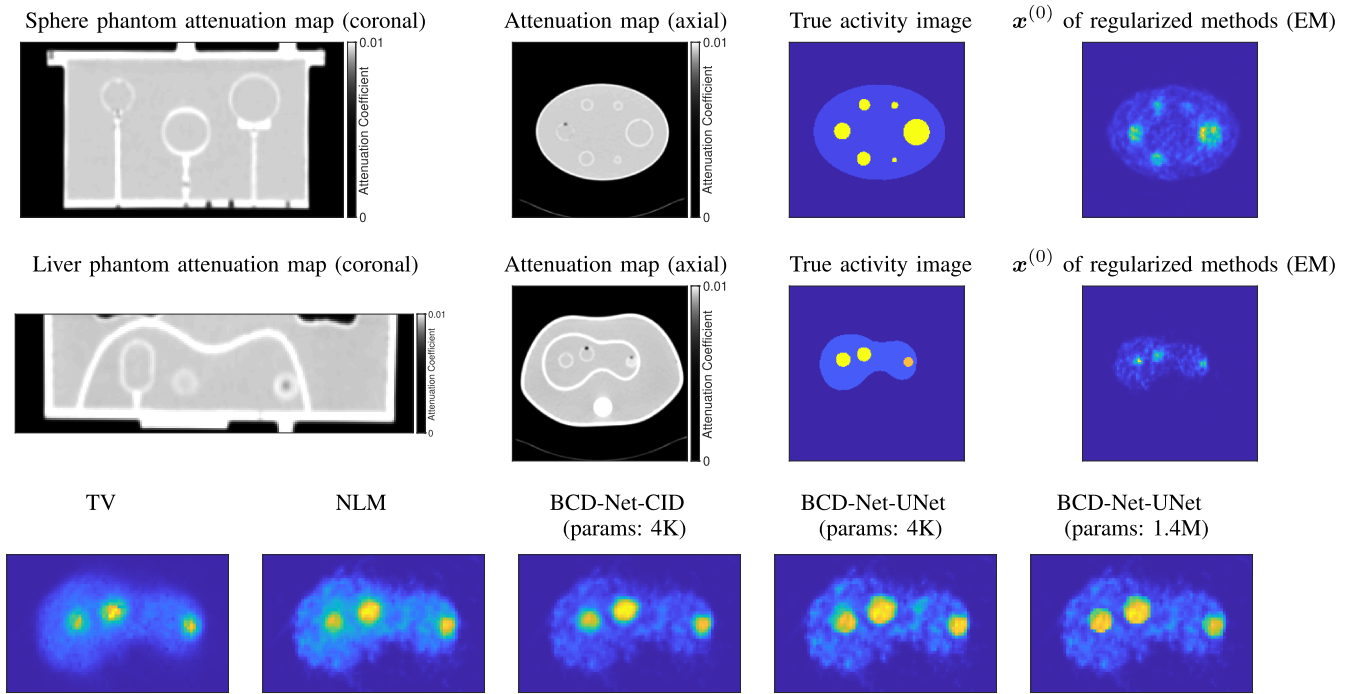
Fig. 4. Y90 PET/CT physical phantom measurement: (First row: training data, Second row: testing data) Attenuation map, true activity, and $x^{(0)}$ of regularized methods of sphere and liver phantom used for training and testing BCD-Net. (Third row) Reconstructed images of one slice from different reconstruction methods.
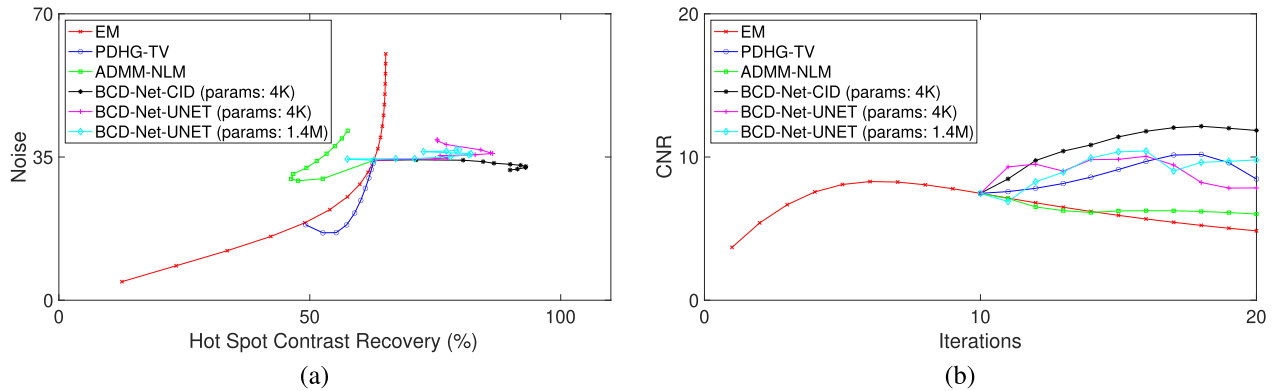

Fig. 5. Liver phantom measurement: (a) Plot of noise in background liver vs contrast recovery in hot spheres. (b) Contrast to noise ratio vs iteration. We initialized regularized methods with the 10th iterate of EM reconstruction.

EM and 10 iterations ($T = 10$) for PDHG-TV, ADMM-NLM, and BCD-Net. We used 1 inner-iteration ($T' = 1$) for the reconstruction module (11). We set $c = 0.01$ in (13) in the XCAT simulation study and $c = 0.005$ in both the phantom measurement and patient studies.

B. Results: Reconstruction (Testing) on Simulation Data

Fig. 2-3 shows that the proposed iterative NN, BCD-Net, significantly improves overall reconstruction performance over the other non-trained regularized MBIR methods. Fig. 3 reports averaged evaluation metrics over realizations. Fig. 3 shows that BCD-Net with a trained CID achieves the best results in most evaluation metrics. In particular, BCD-Net with a CID improves CNR and RMSE compared to PDHG-TV and ADMM-NLM. BCD-Net also improved contrast recovery in the cold region while not increasing noise

compared to the initial EM reconstruction, whereas PHDG-TV and ADMM-NLM improved noise while degrading the CR. For Fig. 2, we selected the iteration number for EM to obtain the highest CNR and the last iteration number for other methods. Fig. 2 shows that BCD-Net's reconstructed image with a CID is closest to the true image whereas PHDG-TV and ADMM-NLM exceedingly blur the cold region. BCD-Net with the U-Net denoiser shows good recovery for the cold region, however, it blurs the hot region. Moreover, the larger sized U-Net (params: 1.4 M) denoiser worsens the performance of BCD-Net possibly due to over-fitting the training dataset.

C. Results: Reconstruction (Testing) on Measurement Data

1) Phantom Study: Similar to the simulation results, Fig. 4-5 shows that, BCD-Net improved overall reconstruction
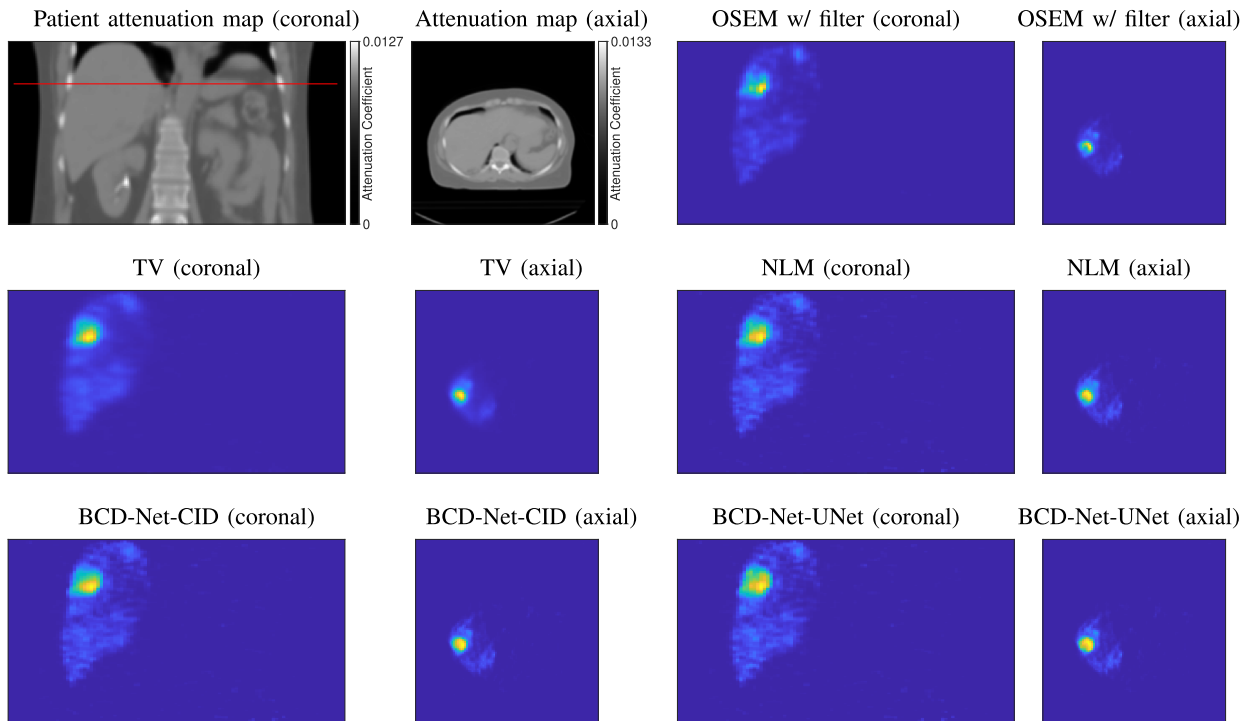
Fig. 6. Y90 PET/CT patient measurement: Attenuation map and reconstructed images of one slice (coronal and axial view) using OSEM, TV, NLM, and BCD-Net. We visualized the reconstructed image of BCD-Net-UNet with 4 K parameters.
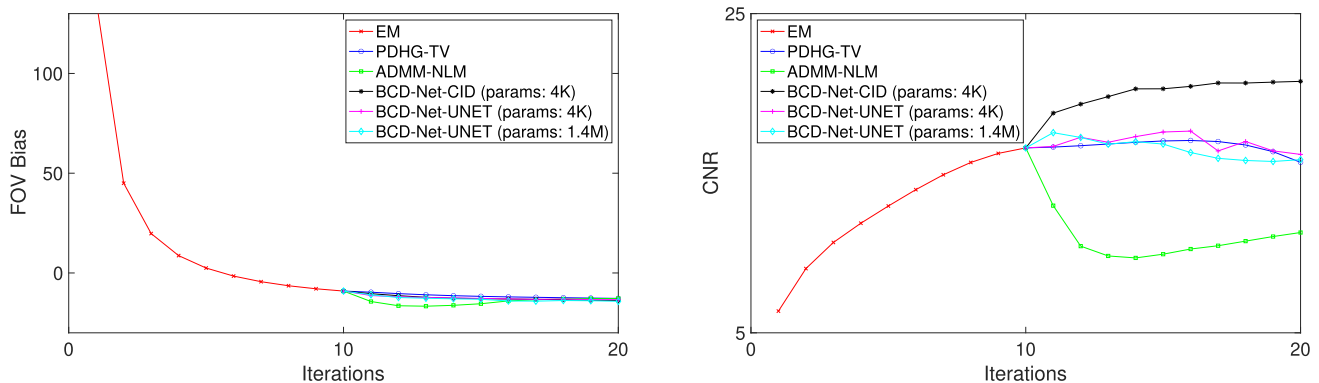

Fig. 7. Patient scan: (a) Field of view bias vs iteration. BCD-Net shows similar results compared to other methods. (b) Contrast to noise ratio vs iteration.

performance over the other reconstruction methods. Fig. 4 shows that reconstructed images using PHDG-TV and ADMM-NLM show uniform texture in background liver compared to EM, however, those exceedingly blur around hot spheres. The blurred hot region is more evident in the quantification results in Fig. 5. BCD-Net gives more visibility for hot spheres with noisier texture in uniform liver region. Fig. 5 shows that BCD-Net with a CID improves CNR compared to PDHG-TV and ADMM-NLM. BCD-Net with CID also improved contrast recovery in hot spheres while slightly increasing noise compared to the initial EM reconstruction. In Fig. 5 (a), BCD-Net with U-Net denoiser shows a fluctuation with iterations, however, the plot trend is similar to that of BCD-Net with CID.

2) Patient Study: Because of the unknown true activity distribution, we quantitatively evaluated each reconstruction method with FOV activity bias. In this quantitative evaluation,

BCD-Net showed similar results compared to other methods. See Fig. 6-7. Fig. 6 shows that the quality of image using different methods in patient study is similar to that of phantom measurement study shown in Fig. 4. Fig. 7 (b) shows that the CNR trend in the patient study is similar to that of the XCAT simulation and the liver phantom measurement.

IV. DISCUSSION

In this study we showed the efficacy of trained BCD-Net on both qualitative and quantitative Y-90 PET/CT imaging and compared between conventional non-trained regularizers. The proposed approach uses learned denoising NNs to lift estimated signals and thresholding operations to remove unwanted signals. In particular, the iterative framework of BCD-Net enables one to train the filters and thresholding values to deal with the different image roughness at its each iteration. We experimentally demonstrate its generalization capabilities

TABLE IV
IMPACT OF IMAGING VARIABLE ON GENERALIZATION
CAPABILITY OF BCD-NET-CID

Changed imaging variable	Training	Testing	RMSE	Drop (%)
Identical	-		4.74	-
Shape and size	See Fig. 2		5.49	15.9
Concentration ratio	9:1	4:1	5.55	17.1
Concentration ratio	1.7:1	4:1	5.81	22.5
Trues Count-level	2×10^5	5×10^5	5.01	5.7
Trues Count-level	11×10^5	5×10^5	5.71	20.5

with simulation and measurement data. In the XCAT PET/CT simulation with activity distributions and count-rates mimicking Y-90 PET imaging, total counts in the cold spot were overestimated with standard reconstruction and other MBIR methods using non-trained regularization, yet approached the true value with the proposed approach. Improvements were also demonstrated for the measurement data where we used training and testing datasets having very different activity distribution and count-levels. The architecture and size of denoising NN significantly affect the performance of BCD-Net. In both simulation and measurement experiments, the CID outperformed the U-Net architectures. Using a U-Net with more trainable parameters degraded the performance, especially in the simulation study, due to the small size of dataset. Size of the denoising NN should be set with consideration of training dataset size.

We tested which imaging variable most affects the generalization performance of the proposed BCD-Net. Table IV shows how BCD-Net performs when training and testing data had the same activity distribution and count-level (only difference is Poisson noise) and how the performance of BCD-Net is degraded when each imaging variable is changed between training and testing dataset. We changed one of three factors (shape and size of tumor and liver, concentration ratio, count-level) in training dataset compared to testing dataset. The result shows that generalization performance of the proposed BCD-Net depends largely on all imaging variables. However, training with higher contrast and lower count-level dataset (compared to testing dataset) gave less degradation of performance compared to the opposite cases. This result suggests that it is better to have noisier data in training dataset than testing dataset. In other words, training for extra noise reduction than needed is better than less noise reduction than needed.

We also investigated how each factor in training of denoising module (7) impacts the generalization capability of BCD-Net. Fig. 8(a)-(b) show the impact of number and size of filters on performance. Plots show that the proposed BCD-Net achieved lower training RMSE when using larger number and size of filters; however, it did not decrease testing RMSE compared to smaller number and size of filters and BCD-Net with larger size of filter exceedingly blurs image thereby resulting in higher RMSE. See Fig. 8(e). We also tested $l1$ training loss to see if it improves the performance over the $l2$ loss (MSE) in (9). However, it led to unnaturally piece-wise constant images and details in small cold regions were ignored.

(a) Impact of number of filter (Fixed $R = 3^3$) (b) Impact of size of filter (Fixed $K = 2^5$)

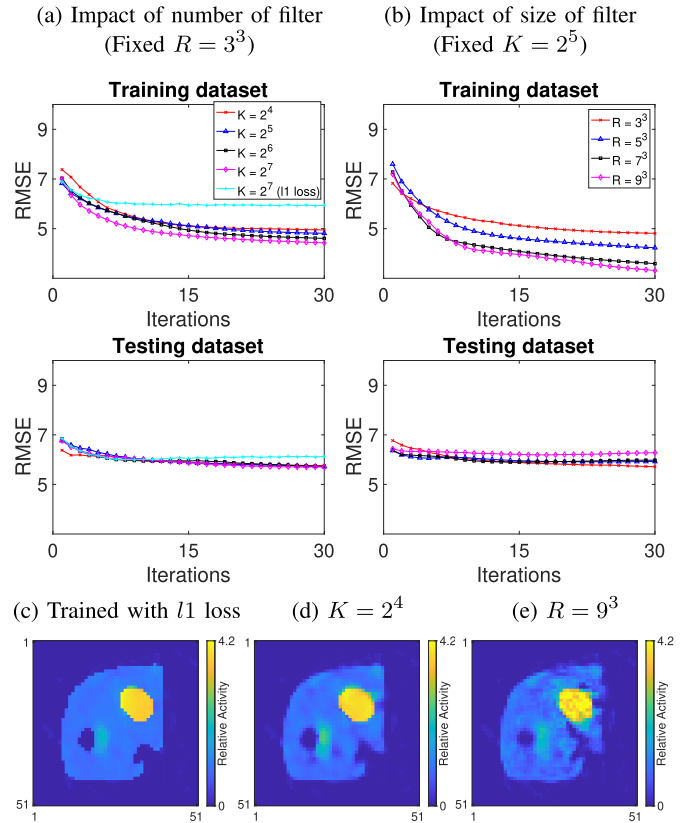(c) Trained with $l1$ loss   (d) $K = 2^4$ (e) $R = 9^3$

Fig. 8. (a)-(b) Impact of number/size of filter and training loss on testing dataset RMSE. (c) Reconstructed image from BCD-Net-CID with filters and thresholding values trained with $l1$-loss.

Fig. 9 shows how the regularization parameter β in (13) changes with iterations in training and testing datasets. The β value in each iteration converges to different limits in training and testing cases. The adaptive scheme automatically increases the β value when the count-level decreases. This behavior concurs with the general knowledge that more regularization is needed when the noise-level increases. These empirical results underscore the importance of such adaptive regularization parameter selection schemes proposed in Section II-C.2 in PET imaging.

Many related works [5], [14], [15], [49] use single image denoising (deep) NN (e.g., U-Net) as a post-reconstruction processing and we investigated how the denoising NN detached from the data-fit term performs compared to iterative NN. Fig. 10 illustrates $u^{(1)}$ generated by CID and U-Net. As in the iterative NN, using more trainable parameters degraded the visual quality and RMSE value in U-Net case and CID achieved better result than U-Net. In all cases, iterative NN achieved lower RMSE compared to those post-reconstruction processed images as shown in Table V.

BCD-Net is trained for a specific number of iterations and its practical use would be akin to how ML-EM is used with a fixed number of iterations in clinical systems. If one is interested in convergence guarantees with running more iterations, then one can extend the sequence convergence guarantee of BCD-Net in [23] by setting the nth "adaptive" denoiser as $\widetilde{\mathcal{D}}^{(n)} = g_2(\mathcal{D}^{(n)}(g_1(x^{(n-1)})))$ with some nth denoiser $\mathcal{D}^{(n)}$ (e.g., CID (7) and U-Net), $\forall n$, using sufficient T' (so MAP
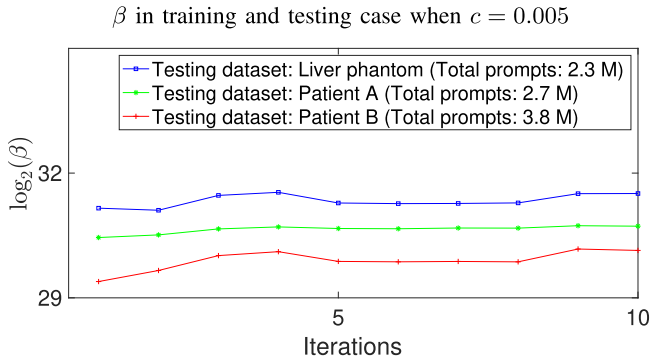
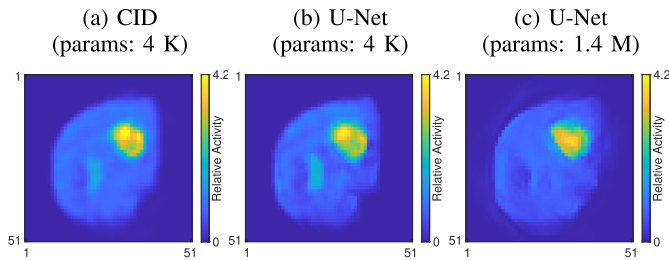Fig. 9. Efficacy of adaptive selection of regularization parameter β.

(a) CID	(b) U-Net	(c) U-Net
(params: 4 K)	(params: 4 K)	(params: 1.4 M)

Fig. 10. $g_2(\boldsymbol{u}^{(1)})$ generated by (a) CID and (b)-(c) U-Net. Using more parameters degraded the visual quality and RMSE value as in the iterative NN.

TABLE V

COMPARISON BETWEEN POST-RECONSTRUCTION PROCESSING AND ITERATIVE NN

Method	$g_2\left(\boldsymbol{u}^{(1)}\right)$ RMSE	$\boldsymbol{x}^{(30)}$ RMSE	Δ (%)
CID (params: 4 k)	6.87	6.36	7.52
U-Net (params: 4 k)	7.39	6.70	9.30
U-Net (params: 1.4 M)	7.67	7.04	8.15

EM finds a critical point), and additionally assuming that $\beta^{(n)}$ converges. We empirically observed that the $\beta^{(n)}$ tends to converge to some constant in this Y-90 PET as well as another application of Lu-177 SPECT.

To more practically guarantee the convergence, one could use training and testing dataset having similar count-level and a fixed regularization parameter value across iterations using an initial estimated image and a corresponding denoised image as follows:

$$\beta = \frac{\left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(0)})\right\|_2}{\left\|\nabla_{\boldsymbol{x}} \mathsf{R}(\boldsymbol{x}^{(0)})\right\|_2} \cdot c = \frac{\left\|a_j - e_j(\boldsymbol{x}^{(0)})\right\|_2}{\left\|\boldsymbol{x}^{(0)} - g_2(\boldsymbol{u}^{(1)})\right\|_2} \cdot c.$$

The convergence properties depend on additional technical assumptions detailed in [23].

V. CONCLUSION

It is important for a "learned" regularizer to have generalization capability to help ensure good performance when applying it to an unseen dataset. For low-count PET reconstruction, the proposed iterative NN, BCD-Net, showed reliable generalization capability even when the training dataset is small. The proposed BCD-Net achieved significant qualitative and quantitative improvements over the conventional MBIR methods using "hand-crafted" non-trained regularizers: TV and NLM. In particular, these conventional

MBIR methods have a trade-off between noise and recovery accuracy, whereas the proposed BCD-Net improves CR for hot regions while not increasing the noise when the regularization parameter is appropriately set. Visual comparisons of the reconstructed images also show that the proposed BCD-Net significantly improves PET image reconstruction performance compared to MBIR methods using non-trained regularizers.

Future work includes investigating performance of BCD-Net trained with end-to-end training principles and adaptive selection of trainable parameter numbers depending on the size of training dataset.

REFERENCES

[1] T. Carlier, K. P. Willowson, E. Fourkal, D. L. Bailey, M. Doss, and M. Conti, "^{90}Y -PET imaging: Exploring limitations and accuracy under conditions of low counts and high random fraction," *Med. Phys.*, vol. 42, no. 7, pp. 4295–4309, Jun. 2015.

[2] S. Ahn *et al.*, "Quantitative comparison of OSEM and penalized likelihood image reconstruction using relative difference penalties for clinical PET," *Phys. Med. Biol.*, vol. 60, no. 15, p. 5733, 2015.

[3] G. Wang, J. C. Ye, K. Mueller, and J. A. Fessler, "Image reconstruction is a new frontier of machine learning," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1289–1296, Jun. 2018.

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

[5] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017.

[6] J. C. Ye, Y. Han, and E. Cha, "Deep convolutional framelets: A general deep learning framework for inverse problems," *SIAM J. Imag. Sci.*, vol. 11, no. 2, pp. 991–1048, Jan. 2018.

[7] H. Gupta, K. H. Jin, H. Q. Nguyen, M. T. McCann, and M. Unser, "CNN-based projected gradient descent for consistent CT image reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1440–1453, Jun. 2018.

[8] I. Y. Chun, H. Lim, Z. Huang, and J. A. Fessler, "Fast and convergent iterative image recovery using trained convolutional neural networks," in *Proc. 56th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Monticello, IL, USA, Oct. 2018, pp. 155–159.

[9] H. K. Aggarwal, M. P. Mani, and M. Jacob, "MoDL: Model-based deep learning architecture for inverse problems," *IEEE Trans. Med. Imag.*, vol. 38, no. 2, pp. 394–405, Feb. 2019.

[10] K. Hammernik *et al.*, "Learning a variational network for reconstruction of accelerated MRI data," *Magn. Reson. Med.*, vol. 79, no. 6, pp. 3055–3071, Jun. 2018.

[11] J. Sun *et al.*, "Deep ADMM-Net for compressive sensing MRI," in *Proc. NIPS*, 2016, pp. 10–18.

[12] M. Mardani *et al.*, "Deep generative adversarial neural networks for compressive sensing MRI," *IEEE Trans. Med. Imag.*, vol. 38, no. 1, pp. 167–179, Jan. 2019.

[13] G. Yang *et al.*, "DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1310–1321, Jun. 2018.

[14] J. Xu, E. Gong, J. Pauly, and G. Zaharchuk, "200x low-dose PET reconstruction using deep learning," 2017, *arXiv:1712.04119*. [Online]. Available: http://arxiv.org/abs/1712.04119

[15] B. Yang, L. Ying, and J. Tang, "Artificial neural network enhanced Bayesian PET image reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1297–1309, Jun. 2018.

[16] I. Haggstrom, C. R. Schmidtlein, G. Campanella, and T. J. Fuchs, "DeepPET: A deep encoder–decoder network for directly solving the PET image reconstruction inverse problem," *Med. Image Anal.*, vol. 54, pp. 253–262, May 2019.

[17] K. Gong *et al.*, "Iterative PET image reconstruction using convolutional neural network representation," *IEEE Trans. Med. Imag.*, vol. 38, no. 3, pp. 675–685, Mar. 2019.

[18] K. Kim *et al.*, "Penalized PET reconstruction using deep learning prior and local linear fitting," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1478–1487, Jun. 2018.

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

[20] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[21] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn.*, 2010, pp. 399–406.

[22] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1256–1272, Jun. 2017.

[23] I. Yong Chun, Z. Huang, H. Lim, and J. A. Fessler, "Momentum-net: Fast and convergent iterative neural network for inverse problems," 2019, *arXiv:1907.11818*. [Online]. Available: http://arxiv.org/abs/1907.11818

[24] S. Ye, Y. Long, and I. Yong Chun, "Momentum-net for low-dose CT image reconstruction," 2020, *arXiv:2002.12018*. [Online]. Available: http://arxiv.org/abs/2002.12018

[25] Y. Chun and J. A. Fessler, "Deep BCD-net using identical encoding-decoding CNN structures for iterative image recovery," in *Proc. IEEE 13th Image, Video, Multidimensional Signal Process. Workshop (IVMSP)*, Zagori, Greece, Jun. 2018, pp. 1–5.

[26] I. Y. Chun and J. A. Fessler, "Convolutional analysis operator learning: Acceleration and convergence," *IEEE Trans. Image Process.*, vol. 29, pp. 2108–2122, 2020.

[27] I. Y. Chun, D. Hong, B. Adcock, and J. A. Fessler, "Convolutional analysis operator learning: Dependence on training data," *IEEE Signal Process. Lett.*, vol. 26, no. 8, pp. 1137–1141, Aug. 2019.

[28] C. Crockett, D. Hong, I. Y. Chun, and J. A. Fessler, "Incorporating handcrafted filters in convolutional analysis operator learning for ill-posed inverse problems," in *Proc. IEEE 8th Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, Guadeloupe, West Indies, Dec. 2019, pp. 316–320.

[29] I. Y. Chun, X. Zheng, Y. Long, and J. A. Fessler, "BCD-Net for low-dose CT reconstruction: Acceleration, convergence, and generalization," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, Shenzhen, China, Oct. 2019, pp. 31–40.

[30] Z. Li, I. Y. Chun, and Y. Long, "Image-domain material decomposition using an iterative neural network for dual-energy CT," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Iowa City, IA, USA, Apr. 2020, pp. 651–655.

[31] H. Lim, Z. Huang, J. A. Fessler, Y. K. Dewaraja, and I. Y. Chun, "Application of trained deep BCD-net to iterative low-count PET image reconstruction," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, Sydney, NSW, Australia, Nov. 2018, pp. 1–4.

[32] M. Elschot, M. G. E. H. Lam, M. A. A. J. van den Bosch, M. A. Viergever, and H. W. A. M. de Jong, "Quantitative Monte Carlo-based 90Y SPECT reconstruction," *J. Nucl. Med.*, vol. 54, no. 9, pp. 1557–1563, Sep. 2013.

[33] A. S. Pasciak *et al.*, "Radioembolization and the dynamic role of 90Y PET/CT," *Frontiers Oncol.*, vol. 4, p. 38, Feb. 2014.

[34] J. Nuyts, D. Beque, P. Dupont, and L. Mortelmans, "A concave prior penalizing relative differences for maximum-a-posteriori reconstruction in emission tomography," *IEEE Trans. Nucl. Sci.*, vol. 49, no. 1, pp. 56–60, Feb. 2002.

[35] L. Pfister and Y. Bresler, "Learning sparsifying filter banks," *Proc. SPIE*, vol. 9597, Aug. 2015, Art. no. 959703.

[36] J.-F. Cai, H. Ji, Z. Shen, and G.-B. Ye, "Data-driven tight frame construction and image denoising," *Appl. Comput. Harmon. Anal.*, vol. 37, no. 1, pp. 89–105, Jul. 2014.

[37] I. Y. Chun and J. A. Fessler, "Convolutional analysis operator learning: Application to sparse-view CT : (Invited Paper)," in *Proc. 52nd Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, USA, Oct. 2018, pp. 1631–1635.

[38] A. R. De Pierro, "A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography," *IEEE Trans. Med. Imag.*, vol. 14, no. 1, pp. 132–137, Mar. 1995.

[39] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*. New York, NY, USA: Cambridge Univ. Press, 1988.

[40] Z. Zhang *et al.*, "Optimization-based image reconstruction from low-count, list-mode TOF-PET data," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 4, pp. 936–946, Apr. 2018.

[41] A. Chambolle and T. Pock, "An introduction to continuous optimization for imaging," *Acta Numerica*, vol. 25, pp. 161–319, May 2016.

[42] S. Y. Chun, Y. K. Dewaraja, and J. A. Fessler, "Alternating direction method of multiplier for tomography with nonlocal regularizers," *IEEE Trans. Med. Imag.*, vol. 33, no. 10, pp. 1960–1968, Oct. 2014.

[43] S. Boyd, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.

[44] W. P. Segars, G. Sturgeon, S. Mendonca, J. Grimes, and B. M. W. Tsui, "4D XCAT phantom for multimodality imaging research," *Med. Phys.*, vol. 37, no. 9, pp. 4902–4915, Aug. 2010.

[45] H. Lim, Y. K. Dewaraja, and J. A. Fessler, "A PET reconstruction formulation that enforces non-negativity in projection space for bias reduction in Y-90 imaging," *Phys. Med. Biol.*, vol. 63, no. 3, Feb. 2018, Art. no. 035042.

[46] A. Paszke *et al.*, "Automatic differentiation in PyTorch," in *Proc. NIPS-W*, 2017, pp. 1–4.

[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[48] J. Zbontar *et al.*, "FastMRI: An open dataset and benchmarks for accelerated MRI," 2018, *arXiv:1811.08839*. [Online]. Available: http://arxiv.org/abs/1811.08839

[49] E. Kang, J. Min, and J. C. Ye, "A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction," *Med. Phys.*, vol. 44, no. 10, pp. e360–e375, Oct. 2017.