Application of trained Deep BCD-Net to iterative low-count PET image reconstruction

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

***Abstract*—Image reconstruction in low-count PET is 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, 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 iterative reconstruction. We apply the newly developed block coordinate descent network (BCD-Net) to PET reconstruction by modifying the image reconstruction module to incorporate PET physics. Using the XCAT phantom we simulated the low true coincidence count-rates with high random fractions typical for Y-90 PET patient imaging after Y-90 microsphere radioembolization. We trained a 10 layer BCD-Net where each layer has 200 convolutional filters that encode/decode an input image. Numerical results show that deep BCD-Net significantly improves PET reconstruction performance compared to iterative image reconstruction using non-trained regularizers (total variation (TV) and non-local means (NLM)). We selected the regularization parameter for each method to obtain the highest contrast to noise ratio (CNR). BCD-Net improved activity recovery for a hot sphere significantly and reduced noise, whereas non-trained regularizers had a trade-off between noise and quantification. BCD-Net improved CNR and activity recovery by 96.6% (150.0%) and 41.5% (35.9%) compared to TV (NLM) regularized reconstruction.**

I. Introduction

Image reconstruction in low-count PET is challenging because dominant gammas from natural radioactivity in Lu-based crystals cause low signal-to-noise-ratio (SNR) measurements [1]. To accurately reconstruct images in low-count PET, iterative image reconstruction is most widely used along with a regularization term $\mathsf{R}(\boldsymbol{x})$ that penalizes image roughness and controls noise:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x} \geq \boldsymbol{0}} f(\boldsymbol{x}; \boldsymbol{y}) + \mathsf{R}(\boldsymbol{x}). \quad (1)$$

Here, $f(\boldsymbol{x}; \boldsymbol{y})$ is the Poisson negative log-likelihood between measurement $\boldsymbol{y}$ and estimated measurement means $\bar{\boldsymbol{y}}(\boldsymbol{x})$, $\bar{\boldsymbol{y}}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \bar{\boldsymbol{r}}$, the matrix $\boldsymbol{A}$ denotes the system model, and $\bar{\boldsymbol{r}}$ denotes the mean background events such as scatter and random coincidences. Recently, applying learned regularizers to $\mathsf{R}(\boldsymbol{x})$ is emerging for iterative image reconstruction [2].

While there is much ongoing research on machine learning or deep-learning techniques applied to CT [3] and MRI [4] reconstruction problems, only a few studies have applied these techniques to PET. The past PET studies mostly use deep learning in image space without exploiting the physical imaging model. For example, [5] applied deep-learning network to train a network to enhance low dose FDG PET reconstruction, however, its framework uses the acquisition data only to form the initial image. Therefore, the reconstruction quality depends greatly on the training data set and information from atypical imaging situations and variable noise levels (that are not part of the training set) may not be recovered well. Recently [6] proposed an iterative PET reconstruction framework using a deep-learning based regularizer. Our proposed *BCD-Net* also uses a regularizer that penalizes differences between the unknown image and the "denoised" image given by the network. However, whereas [6] trained only a single image denoising network, our proposed method is a recurrent framework that is composed of multiple trained networks. This recurrent framework enables networks in the later stages to learn how to recover fine details. BCD-Net is constructed by unfolding block coordinate descent (BCD) image recovery algorithm using learned convolutional analysis operators [7], and significantly improved image recovery accuracy in some extreme imaging applications [8]. This paper *1)* modifies the architecture of BCD-Net for PET image reconstruction with an efficient image reconstruction module, *2)* applies a deep BCD-Net that is trained for realistic low-count PET imaging environments and compares its performance with those of non-trained regularizes. Our proposed deep learning-based reconstruction 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.

II. Methods

A. Trained BCD-Net for iterative PET image reconstruction

BCD-Net is inspired by signal recovery using a "learned" regularizer that consists of trained convolutional filters and thresholding values in an analysis form [7]:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \min_{\boldsymbol{z}} f(\boldsymbol{x}; \boldsymbol{y}) + \beta \big(\sum_{k=1}^{K} ||\boldsymbol{c}_k * \boldsymbol{x} - \boldsymbol{z}_k||_2^2 + \alpha_k ||\boldsymbol{z}_k||_1 \big), \quad (2)$$

where β is regularization parameter, $\{\boldsymbol{c}_k \in \mathbb{R}^R : k = 1, \ldots, K\}$ is a set of convolutional filters, $\{\boldsymbol{z}_k \in \mathbb{R}^{n_p} : k = 1, \ldots, K\}$ 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 $\{\boldsymbol{z}_k\}$, n_p is the number of image voxels, and R and K is the

Asterisk (*) indicates corresponding author.

Hongki Lim, Zhengyu Huang, Jeffrey A. Fessler, and Il Yong Chun are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (email: {hongki, zyhuang, fessler, iychun}@umich.edu).

Yuni K. Dewaraja is with the Department of Radiology, University of Michigan, Ann Arbor, MI 48109 USA (yuni@med.umich.edu).

978-1-5386-8494-8/18/$31.00 ©2018 IEEE

size and number of learned filters, respectively. Ultimately, we hope that the learned regularizer can better separate true signal from noisy components by lifting estimated signals with trained convolutional kernels and removing unwanted signals with thresholding operations.

Block Coordinate Descent (BCD) algorithm solves (2) by alternatively updating $\{\boldsymbol{z}_k\}$ and $\boldsymbol{x}$:

$$\begin{aligned}\{\boldsymbol{z}_k^{(n+1)}\} &= \underset{\{\boldsymbol{z}_k\}}{\operatorname{argmin}} ||\boldsymbol{c}_k * \boldsymbol{x}^{(n)} - \boldsymbol{z}_k||_2^2 + \alpha_k||\boldsymbol{z}_k||_1 \\ &= \mathcal{T}(\boldsymbol{c}_k * \boldsymbol{x}^{(n)}, \alpha_k) \qquad (3)\end{aligned}$$

$$\boldsymbol{x}^{(n+1)} = \underset{\boldsymbol{x}}{\operatorname{argmin}}\, f(\boldsymbol{x}) + \beta(\sum_{k=1}^{K} ||\boldsymbol{c}_k * \boldsymbol{x} - \boldsymbol{z}_k^{(n+1)}||_2^2), \qquad (4)$$

where $\mathcal{T}(\cdot,\cdot)$ is the element-wise soft thresholding operator:

$$\mathcal{T}(\boldsymbol{t}, q)_j := \begin{cases} t_j - q \cdot \operatorname{sign}(t_j), & |t_j| > q \\ 0, & |t_j| \le q. \end{cases}$$

Assuming that learned filters $\{\boldsymbol{c}_k\}$ satisfy the tight-frame condition, $\sum_{k=1}^{K} ||\boldsymbol{c}_k * \boldsymbol{x}||_2^2 = ||\boldsymbol{x}||_2^2 \ \forall \boldsymbol{x}$ [7], we rewrite the updates in (3)-(4) as follows:

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

$$\boldsymbol{x}^{(n+1)} = \underset{\boldsymbol{x}}{\operatorname{argmin}}\, f(\boldsymbol{x}) + \beta||\boldsymbol{x} - \boldsymbol{u}^{(n+1)}||_2^2, \qquad (6)$$

where $\tilde{\boldsymbol{c}}_k$ is a rotated version of $\boldsymbol{c}_k$. In this work, we train separate decoding convolutional filters $\{\boldsymbol{d}_k\}$ instead of using $\{\tilde{\boldsymbol{c}}_k\}$ to have better denoising capability. We train the image denoising module in the form of (5) – specifically, consisting of encoding and decoding filters, and thresholding values $\{\boldsymbol{c}_k^{(n+1)}, \boldsymbol{d}_k^{(n+1)}, \alpha_k^{(n+1)} : \forall k, n\}$ – that "best" map between high-quality images (e.g., true images if available) and noisy images in the sense of mean squared error:

$$\underset{\{\boldsymbol{c}_k\},\{\boldsymbol{d}_k\},\{\alpha_k\}}{\operatorname{argmin}} \sum_{l=1}^{L} \left|\left| \boldsymbol{x}_{\text{true},l} - \sum_{k=1}^{K} \boldsymbol{d}_k * \left(\mathcal{T}(\boldsymbol{c}_k * \boldsymbol{x}_l^{(n)}, \alpha_k)\right) \right|\right|_2^2,$$

where $\{\boldsymbol{x}_{\text{true},l} \in \mathbb{R}^{n_p} : l = 1, \ldots, L\}$ is a set of true images and $\{\boldsymbol{x}_l^{(n)} \in \mathbb{R}^{n_p} : l = 1, \ldots, L\}$ is a set of images estimated by image reconstruction module in the nth layer. With trained convolutional filters and soft-thresholding values, we define the following updates for each layer:

$$\boldsymbol{u}^{(n+1)} = \sum_{k=1}^{K} \boldsymbol{d}_k^{(n+1)} * \left(\mathcal{T}(\boldsymbol{c}_k^{(n+1)} * \boldsymbol{x}^{(n)}, \alpha_k^{(n+1)})\right) \qquad (7)$$

$$\boldsymbol{x}^{(n+1)} = \underset{\boldsymbol{x}}{\operatorname{argmin}}\, f(\boldsymbol{x}) + \beta||\boldsymbol{x} - \boldsymbol{u}^{(n+1)}||_2^2. \qquad (8)$$

Fig. 1 shows the corresponding BCD-Net architecture. We name the $\boldsymbol{u}$ and $\boldsymbol{x}$ updates in (7)-(8) as following two *modules*: 1) image denoising module and 2) image reconstruction module. A *layer* refers to one loop of denoising and reconstruction module.

For efficient image reconstruction module (8) in PET, we use EM-surrogate of Poisson log-likelihood function [9].

B. Conventional non-trained regularizers

We compared the proposed BCD-Net with two standard non-trained regularizers.

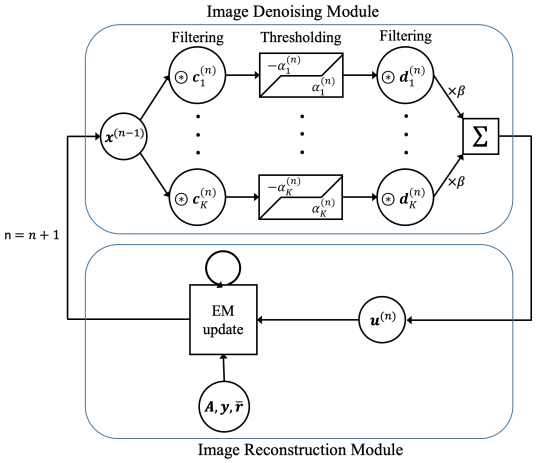

Fig. 1. Architecture of modified BCD-Net. Due to the feedback, BCD-Net can be categorized as a recurrent network.

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

$$\mathsf{R}(\boldsymbol{x}) = \beta||\boldsymbol{C}\boldsymbol{x}||_1,$$

where $\boldsymbol{C}$ is finite differencing matrix. Recent work [10] applied Primal-Dual Hybrid Gradient (PDHG) [11] for TV-regularized reconstruction 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 minimizes the differences between nearby patches in image:

$$\mathsf{R}(\boldsymbol{x}) = \beta \sum_{i,j \in S_i} p(||\boldsymbol{N}_i\boldsymbol{x} - \boldsymbol{N}_j\boldsymbol{x}||_2^2),$$

where $p(t)$ is a potential function of a scalar variable t, S_i is the search neighborhood around the ith voxel, and $\boldsymbol{N}_i$ is a patch extraction operator at the ith voxel. We implemented Fair potential function for $p(t)$. Unlike conventional local filters which assumes similarity between only adjacent voxels, NLM filters can average image intensities over distant voxels. As in [12], we used ADMM to accelerate algorithmic convergence with adaptive penalty parameter selection method [13].

III. Results and Discussion

A. Experimental setup

Y-90 PET imaging is challenging because of the low true coincidence rate. We used XCAT [14] phantom (Fig. 2) to simulate Y-90 PET following radioembolization. To simulate the extremely low count scan, typical for Y-90 PET, we set total true coincidences to 3×10^5 and random fraction to 90% based on patient PET imaging performed after radiomebolization [15]. We placed a 42ml hot sphere (lesion) and cold sphere inside the warm background (liver) and the activity concentration ratio between lesion and liver was 5:1. We evaluated each reconstruction with contrast to noise ratio (CNR), activity recovery (AR) at volume of interest (VOI: hot

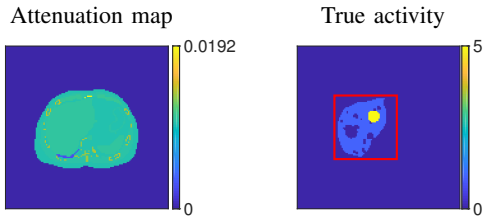

Fig. 2. Attenuation map and true relative activity distribution.

TABLE I
EVALUATION METRICS FOR RECONSTRUCTED IMAGES FROM DIFFERENT REGULARIZATION METHODS

	AR Lesion	AR Liver	CR Cold Spot	CNR	RMSE
EM	89.4	86.9	71.9	5.0	12.9
PDHG-TV	68.2	86.0	66.5	8.8	7.7
ADMM-NLM	71.0	84.7	70.2	7.0	9.2
BCD-Net	**96.5**	**88.8**	**78.9**	**17.5**	**5.9**

sphere, background), contrast recovery (CR) (VOI: cold spot), and root mean squared error (RMSE):

$$\text{AR } (\%) = \frac{\text{Estimated } C_{\text{VOI}}}{\text{True } C_{\text{VOI}}} \times 100(\%)$$

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

$$\text{CNR} = \frac{C_{\text{lesion}} - C_{\text{bkg}}}{STD_{\text{bkg}}}$$

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

where C_{VOI} is mean counts in the volume of interest (VOI), STD_{bkg} is standard deviation between voxel values in uniform background liver, and J_{FOV} is the total number of voxels in field of view (FOV).

B. Training BCD-Net

We trained BCD-Net using six pairs of true image and realization. We generated two activity distributions with varying the location of hot and cold spot. For each activity distribution, we generated three realizations to train the denoising network to deal with the Poisson noise. We used 20 EM algorithm iterations to get the initial image. We trained 3D convolutional filters and thresholding values in each layer with a stochastic gradient descent method using PyTorch [16] deep-learning library. We trained a ten-layer BCD-Net where each layer has 200 sets (K) of thresholding value and convolutional encoding/decoding filters. We set the size of each filter as $3 \times 3 \times 3$ ($R = 3^3$), and the initial thresholding values by sorting the initial estimate of image and getting a 10% largest value of sorted initial image. We used Adam optimization method to train the network with learning rate of 10^{-3} for filters and 10^{-1} for thresholding values. We applied the learning rate decay scheme (e.g., decreasing the learning rate as a factor of 0.9 per 10 epochs.)

C. Reconstruction results and discussion

We trained BCD-Net using an image having a different lesion and cold spot than in the training images. We used 200 iterations for EM, PDHG-TV, ADMM-NLM, and 20 iterations

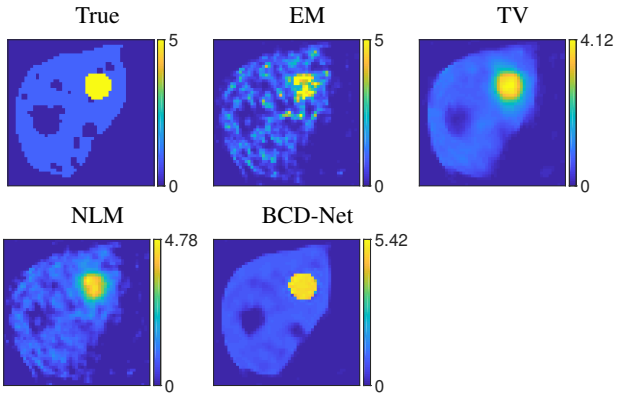

Fig. 3. Reconstructed images of one slice from different reconstruction methods.

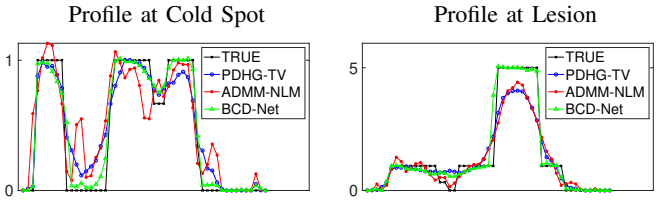

Fig. 4. Line profile where true image contains hot, warm and cold region.

for the reconstruction module (8) at each layer of BCD-Net. For each regularizer, we finely tuned the regularization parameter β (within range $[2^{-15}, 2^{15}]$) to achieve highest CNR and lowest RMSE values. For NLM, we additionally tuned the window and search sizes.

Deep BCD-Net significantly improves overall reconstruction performance over the other non-trained regularization methods. See Table I and Fig. 3. Table I shows that BCD-Net achieves best results in all evaluation metrics. In particular, BCD-Net improved CNR and activity recovery by 96.6% (150.0%) and 41.5% (35.9%) compared to PDHG-TV (ADMM-NLM) regularized reconstruction. Fig. 3 shows that reconstructed image using BCD-Net is closest to the true image whereas PHDG-TV exceedingly blurs in cold region and ADMM-NLM is noisy in uniform region. The profiles in Fig. 4 also illustrate that PDHG-TV does not correctly recover the cold region surrounded by warm region, whereas profile of BCD-Net is very close to true value in both hot and cold region.

In this work, we empirically show the convergence of BCD-Net. In Fig. 5, outputs of denoising module and reconstruction module ($\boldsymbol{u}^{(n)}$ and $\boldsymbol{x}^{(n)}$) are visually converged after 4-5 layers. Fig. 6 also demonstrates that the relative difference between $\boldsymbol{x}$-updates is negligible after 5-6 layers.

To analyze the performance of BCD-Net, Fig. 7–8 visualize each step in denoising network. When visualizing, we sorted the thresholding values in the ascending order, therefore the upper ones are least shrunk outputs and the lower ones are most shrunk outputs. In both Fig. 7 and Fig. 8, some filters and thresholding values are trained to smooth the image and detect edges with small gradients (see upper ones) and some filters and thresholding values are trained to detect edges with large gradients (see lower ones).

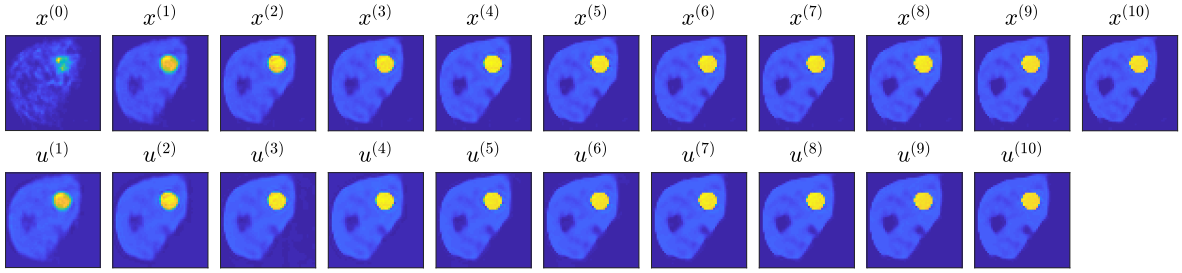

Fig. 5. Outputs of image denoising module and image reconstruction module at each layer. $\boldsymbol{u}^{(n)}$ and $\boldsymbol{x}^{(n)}$ are visually converging.

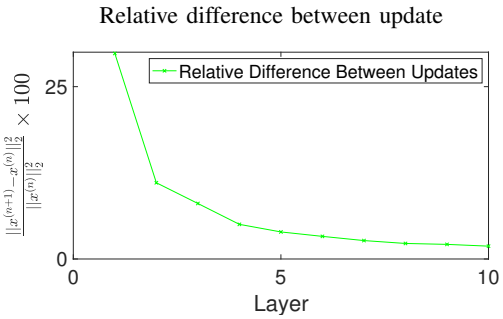

Fig. 6. Empirical demonstration of convergence: relative difference between updates $\left(\frac{||x^{(n+1)}-x^{(n)}||_2^2}{||x^{(n)}||_2^2}\right)$ diminishes as the algorithm iterates.

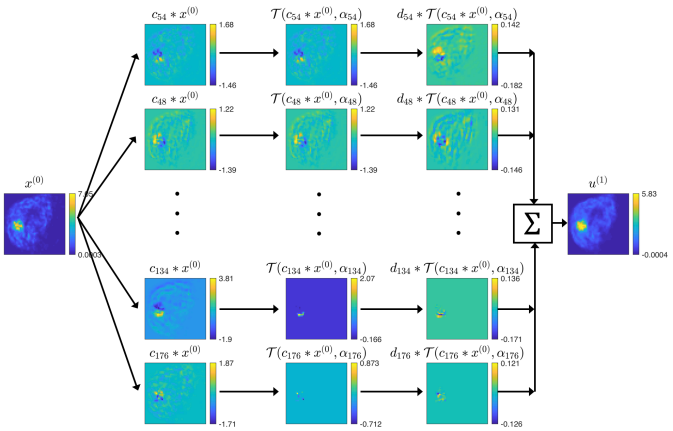

Fig. 7. Visualization of each step in denoising network (filtering-thresholding-filtering) at early layer (n = 0).

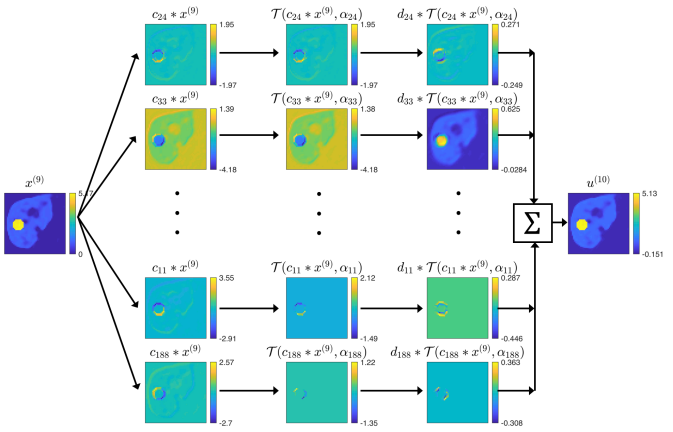

Fig. 8. Visualization of each step in denoising network (filtering-thresholding-filtering) at later layer (n = 9).

IV. Conclusion

Non-trained regularizers had a trade-off between noise and recovery accuracy, whereas BCD-Net improved activity recovery for a hot sphere and reduced noise at the same time. BCD-Net significantly improved CNR and AR compared to EM, TV, and NLM reconstruction.

Future work includes investigating performance of BCD-Net trained with end-to-end training principles, adaptive regularization parameter selection, and testing trained BCD-Net with measurement datasets having various count-levels and activity distributions.

V. Acknowledgment

We acknowledge Se Young Chun (UNIST) for providing NLM regularization codes. This work is supported by NIH-NIBIB grant R01EB022075.

References

[1] T. Carlier, K. P. Willowson, E. Fourkal, D. L. Bailey, M. Doss, and M. Conti, "Y90-PET imaging: exploring limitations and accuracy under conditions of low counts and high random fraction," *Med. Phys.*, vol. 42, no. 7, pp. 4295–309, Jun. 2015.

[2] G. Wang, "A perspective on deep imaging," *IEEE Access*, vol. 4, pp. 8914–24, Nov. 2016.

[3] I. Y. Chun, H. Lim, Z. Huang, and J. A. Fessler, "Fast and convergent iterative signal recovery using trained convolutional neural networks," to appear in *Proc. Allerton*, Oct 2018.

[4] H. K. Aggarwal, M. P. Mani, and M. Jacob, "MoDL: Model Based Deep Learning Architecture for Inverse Problems," *IEEE transactions on medical imaging*, 2018.

[5] J. Xu, E. Gong, J. Pauly, and G. Zaharchuk, "200x low-dose PET reconstruction using deep learning," *arXiv preprint arXiv:1712.04119*, 2017.

[6] K. Kim, D. Wu, K. Gong, J. Dutta, J. H. Kim, Y. D. Son, H. K. Kim, G. El Fakhri, and Q. Li, "Penalized PET reconstruction using deep learning prior and local linear fitting," *IEEE transactions on medical imaging*, vol. 37, no. 6, pp. 1478–1487, 2018.

[7] I. Y. Chun and J. A. Fessler, "Convolutional analysis operator learning: Acceleration and convergence," *arXiv preprint arXiv:1802.05584*, Jan 2018.

[8] ——, "Deep BCD-Net using identical encoding-decoding CNN structures for iterative image recovery," *Proc. IEEE IVMSP Workshop*, Feb 2018.

[9] 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. 132–7, Mar. 1995.

[10] Z. Zhang, S. Rose, J. Ye, A. E. Perkins, B. Chen, C.-M. Kao, E. Y. Sidky, C.-H. Tung, and X. Pan, "Optimization-based image reconstruction from low-count, list-mode TOF-PET data," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 4, pp. 936–946, 2018.

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

[12] S. Y. Chun, Y. K. Dewaraja, and J. A. Fessler, "Alternating direction method of multiplier for tomography with nonlocal regularizers," *IEEE transactions on medical imaging*, vol. 33, no. 10, pp. 1960–1968, 2014.

[13] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. & Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

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

[15] 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," *Physics in medicine and biology*, 2018.

[16] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NIPS-W*, 2017.