Title

PET Image Reconstruction Using Deep Image Prior

Journal

IEEE Transactions on Medical Imaging, 38(7)

ISSN

0278-0062

Authors

Gong, Kuang
Catana, Ciprian
Qi, Jinyi
et al.

Publication Date

2019-07-01

DOI

10.1109/tmi.2018.2888491

Peer reviewed

HHS Public Access

Author manuscript

IEEE Trans Med Imaging. Author manuscript; available in PMC 2020 July 01.

Published in final edited form as:

IEEE Trans Med Imaging. 2019 July ; 38(7): 1655–1665. doi:10.1109/TMI.2018.2888491.

PET Image Reconstruction Using Deep Image Prior

Kuang Gong,
Gordon Center for Medical Imaging, Massachusetts General Hospital and Harvard Medical School, Boston, MA 02114 USA, and with the Department of Biomedical Engineering, University of California, Davis, CA 95616 USA.

Ciprian Catana,
Martinos Center for Biomedical Imaging, Massachusetts General Hospital and Harvard Medical School, Boston, MA 02114 USA

Jinyi Qi, and
Department of Biomedical Engineering, University of California, Davis, CA 95616 USA

Quanzheng Li*
Gordon Center for Medical Imaging, Massachusetts General Hospital and Harvard Medical School, Boston, MA 02114 USA

Abstract

Recently deep neural networks have been widely and successfully applied in computer vision tasks and attracted growing interests in medical imaging. One barrier for the application of deep neural networks to medical imaging is the need of large amounts of prior training pairs, which is not always feasible in clinical practice. This is especially true for medical image reconstruction problems, where raw data are needed. Inspired by the deep image prior framework, in this work we proposed a personalized network training method where no prior training pairs are needed, but only the patient' own prior information. The network is updated during the iterative reconstruction process using the patient specific prior information and measured data. We formulated the maximum likelihood estimation as a constrained optimization problem and solved it using the alternating direction method of multipliers (ADMM) algorithm. Magnetic resonance imaging (MRI) guided Positron emission tomography (PET) reconstruction was employed as an example to demonstrate the effectiveness of the proposed framework. Quantification results based on simulation and real data show that the proposed reconstruction framework can outperform Gaussian post-smoothing and anatomically-guided reconstructions using the kernel method or the neural network penalty.

Keywords

Medical image reconstruction; deep neural network; unsupervised learning; positron emission tomography

*(li.quanzhengdmgh.harvard.edu).

I. Introduction

Over the past several years, deep neural networks have been widely and successfully applied to various imaging tasks such as segmentation [1], object detection [2] and image synethesis [3], by demonstrating better performance than state-of-the-art methods when large amounts of data sets are available. For medical imaging tasks such as lesion detection and region-of-interest (ROI) quantification, obtaining high quality diagnostic images is essential. Recently the neural network method has been applied to transform low-quality images into the images with improved signal-to-noise ratio (SNR).

During network training, images reconstructed from high dose or long-scanned duration are required to be used as the training labels. In some cases collecting large amounts of training labels is easy, such as static magnetic resonance (MR) reconstruction. However, this is not an easy task for some other cases: high-dose computed tomography (CT) has potential safety concerns; long-scanned dynamic positron emission tomography (PET) is not employed in routine clinical practice; in cardiac MR applications, it is impossible to acquire breath-hold fully sampled 3D images. With limited amounts of high-quality patient data sets available, overfitting can be a potential pitfall: if new patient data does not lie in the training space due to population difference, the trained network cannot accurately recover unseen structures. In addition, low-quality images are often simulated by artificially downsampling the full-dose/high-count data, which may not reflect the real physical condition of low-dose imaging. This mismatch between training and the real clinical environment can reduce the network performance.

Apart from using training pairs to perform supervised learning, a lot of prior arts focus on exploiting prior images acquired from the same patient to improve the image quality. The priors can come from temporal information [4], [5], different physics settings [6], or even other imaging modalities [7]. They are included into the maximum posterior estimation or sparse representation framework using pre-defined analytical expressions or pre-learning steps. The pre-defined expressions might not be able to extract all the useful information, and the pre-learnt model might not be optimal for the later reconstruction as no data consistency constraint is enforced during pre-learning. Ideally the learning process should be included inside the reconstruction framework.

Recently, the deep image prior (DIP) framework proposed in [8] shows that convolutional neural networks (CNNs) have the intrinsic ability to regularize a variety of ill-posed inverse problems without pre-training. No prior training pairs are needed and random noise can be employed as the network input to generate denoised images. The ability of CNN to learn the structure information is also revealed in the deep convolutional generative adversarial network (GAN) [9] where the generator is fully convolutional, and the network can generate various distributions based on random noise input. Furthermore, it has been presented in conditional GAN works [10]–[12] that when the input is not random noise, but the associated prior information, the prediction results can be improved. This shows that for the DIP framework, its conditional version using prior information as input, instead of random noise, may generate better results.

Inspired by the prior arts, we proposed a personalized image reconstruction framework based on the conditional DIP method, called DIPRecon. The input is not random noise, but prior images of the same patient. Instead of calculating the mean squared error (MSE) between the network output and the original corrupted image, we formulated the training objective function based on maximum likelihood estimation derived from imaging physics. Modified 3D U-net [13] was employed as the neural network structure. To stabilize the network training, the limited memory BFGS (L-BFGS) algorithm [14] was used instead of adaptive stochastic gradient descent (SGD) methods.

PET is a molecular imaging modality widely used in neurology studies. The image resolution of current PET scanners is still limited by various physical degradation factors [15]. Improving PET image resolution is essential for a lot of applications, such as dopamine neurotransmitter imaging, brain tumor staging and early diagnosis of Alzheimer's disease. For the past decades, various efforts are focusing on using MR or CT to improve PET image quality [16]–[23]. In this work, the anatomically-aided PET image reconstruction problem was presented as an example to demonstrate the effectiveness of the proposed DIPRecon framework. Compared with the state-of-the-art kernel-based method and the penalized reconstruction based on a neural network penalty, DIPRecon shows superior performance both visually and quantitatively in simulation and real data experiments.

This paper is organized as follows. Section 2 describes the related works. Section 3 introduces the DIPRecon framework and implementation details. Section 4 describes the simulations and real data used in the evaluation. Experimental results are shown in section 5, followed by discussions in section 6. Finally, conclusions are drawn in Section 7.

II. Related Works

A. Deep neural network with training pairs

When large amounts of training pairs are available, a neural network can be trained by

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_i \left\| \boldsymbol{x}^i_{\text{label}} - f\left(\boldsymbol{\theta} \mid \boldsymbol{z}^i\right) \right\|, \quad (1)$$

where $f : \mathbb{R} \rightarrow \mathbb{R}$ represents the neural network, $\boldsymbol{\theta} \in \mathbb{R}^L$ indicates the trainable variables, $\boldsymbol{z}^i \in \mathbb{R}^N$ is the network input for the ith training pair, and $\boldsymbol{x}^i_{\text{label}} \in \mathbb{R}^N$ denotes the ith training label. For CNN, $\boldsymbol{\theta}$ contains the convolutional filters and the bias items from all layers. The trained network can be directly applied to image denoising [24]–[27]. Apart from image denoising, some efforts focus on including the trained neural network into an iterative reconstruction framework, using penalty design or variable reparameterization methods [24], [28]–[30]. Unrolled neural network methods have also been proposed, which unfold the iterative update steps and use neural networks to implicitly represent some modules [31]–[36]. Compared to denoising approaches, combining the neural network with iterative reconstruction framework takes the data consistency into consideration and can recover more image details. All of the above approaches require a large number of training pairs to

learn the network parameters $\boldsymbol{\theta}$. Even for GAN, high-quality reference images are still needed in the discriminative network.

B. Train-data-free approaches

In this category, the high-quality training labels are not needed. The model is learned from the measurement data or prior images from other resources.

1) Adaptive dictionary learning—Dictionary learning is an effective method in image denoising [37]. It constructs an overcomplete dictionary and uses the estimated dictionary to approximate the distorted image. Similar to the pre-mentioned neural network method, high-quality reference images can be used to train a global dictionary. When high-quality reference images do not exist, adaptive dictionary learning can be applied where the training and denoising are implemented in an alternating iterative process. Apart from denoising, adaptive dictionary learning has also been applied to image reconstruction by combining the data fidelity item with the constraint from sparse representations [38]–[40].

2) Deep image prior—In [8], the authors proposed the DIP method, where no prior-learning was performed before applied to image restoration/denoising. The general idea is similar to the adaptive dictionary learning approach. Supposing $\boldsymbol{x}_0 \in \mathbb{R}^N$ is the distorted image, the training process is characterized as

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \left\| \boldsymbol{x}_0 - f(\boldsymbol{\theta}|z) \right\|, \hat{\boldsymbol{x}} = f(\hat{\boldsymbol{\theta}}|z), \quad (2)$$

where $\hat{\boldsymbol{x}} \in \mathbb{R}^N$ is the final corrected image output, and the network input $z \in \mathbb{R}^N$ is random noise. No training pairs are needed and $f(\boldsymbol{\theta}|z)$ is updated from scratch. Results from this work indicate that the network structure of a CNN can function as a regularizer and the network weights are the parameters representing $\hat{\boldsymbol{x}}$.

3) Kernel method—The kernel method is a method developed for PET image reconstruction using temporal or anatomical priors [5]. It is based on the assumption that the unknown image $\boldsymbol{x} \in \mathbb{R}^N$ can be represented as

$$\boldsymbol{x} = \boldsymbol{K}_z \boldsymbol{\theta}, \quad (3)$$

where $\boldsymbol{K}_z \in \mathbb{R}^{N \times N}$ is the kernel matrix calculated from the prior image $z \in \mathbb{R}^N$ with a pre-defined kernel basis function. $\boldsymbol{\theta} \in \mathbb{R}^N$ is the kernel coefficients. The kernel representation shown in (3) is included in the iterative reconstruction framework to estimate the kernel coefficients $\boldsymbol{\theta}$. From the network point of view, the kernel method can be considered as a twolayer network: the prior image $\boldsymbol{z}$ is the network input, $\boldsymbol{K}_\boldsymbol{z}$ is a non-local feature extraction layer, $\boldsymbol{\theta}$ is the convolutional filter with spatial size 1 × 1 and N feature channels, and $\boldsymbol{x}$ is the network output. The concept of non-local feature extraction layer is similar to the recently

proposed non-local neural networks [41]. Thus, the kernel based iterative reconstruction can be treated as a two-layer neural network training process.

III. Methodology

A. Background

In inverse problems, such as image deblurring and image reconstruction, the measured data $\boldsymbol{y} \in \mathbb{R}^M$ can be assumed as a collection of independent random variables and its mean $\bar{\boldsymbol{y}} \in \mathbb{R}^M$ is assumed to be related to the original image $\boldsymbol{x} \in \mathbb{R}^N$ through an affine transform

$$\bar{\boldsymbol{y}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{s}, \quad (4)$$

where $\boldsymbol{A} \in \mathbb{R}^{M \times N}$ is the transformation matrix and $\boldsymbol{s} \in \mathbb{R}^M$ is a known additive term. Supposing the measured random variable y_i follows a distribution of $p(y_i|\boldsymbol{x})$, the log likelihood for the measured data $\boldsymbol{y}$ can be written as

$$L(\boldsymbol{y}|\boldsymbol{x}) = \sum_{i=1}^{M} \log p\left(y_i|\boldsymbol{x}\right). \quad (5)$$

B. Proposed framework

In this proposed DIPRecon framework, the unknown image $\boldsymbol{x}$ is represented as

$$\boldsymbol{x} = f(\boldsymbol{\theta}|z), \quad (6)$$

where f represents the neural network, $\boldsymbol{\theta}$ are the unknown parameters of the neural network, z denotes the prior image and is the input to the neural network. When substituting x with the neural network representation in (6), the original data model shown in (4) can be rewritten as

$$\bar{\boldsymbol{y}} = \boldsymbol{A} f(\boldsymbol{\theta}|z) + \boldsymbol{s}. \quad (7)$$

Replacing $\boldsymbol{x}$ by (6), we can express the log likelihood using $\boldsymbol{\theta}$ as

$$L(\boldsymbol{y}|\boldsymbol{\theta}) = \sum_{i=1}^{M} \log p\left(y_i|f(\boldsymbol{\theta}|z)\right). \quad (8)$$

The maximum likelihood estimate of the unknown image $\boldsymbol{x}$ can be calculated in two steps as

$$\hat{\boldsymbol{\theta}} = \arg\max L(\boldsymbol{y}|\boldsymbol{\theta}), \quad (9)$$

$$\hat{\boldsymbol{x}} = f(\hat{\boldsymbol{\theta}}|z). \quad (10)$$

C. Optimization

The objective function in (9) is difficult to solve due to the coupling between the likelihood function and the neural network. Here we transfer it to the constrained format as below

$$\begin{aligned} &\max \quad L(\boldsymbol{y}|\boldsymbol{x}) \\ &\text{s.t.} \quad \boldsymbol{x} = f(\boldsymbol{\theta}|z). \end{aligned} \quad (11)$$

We use the augmented Lagrangian format for the constrained optimization problem in (11) as

$$L_\rho = L(\boldsymbol{y}|\boldsymbol{x}) - \frac{\rho}{2}\left\|\boldsymbol{x} - f(\boldsymbol{\theta}|z) + \boldsymbol{\mu}\right\|^2 + \frac{\rho}{2}\left\|\boldsymbol{\mu}\right\|^2, \quad (12)$$

which can be solved by the ADMM algorithm iteratively in three steps

$$\boldsymbol{x}^{n+1} = \arg\max_{\boldsymbol{x}} L(\boldsymbol{y}|\boldsymbol{x}) - \frac{\rho}{2}\left\|\boldsymbol{x} - f(\boldsymbol{\theta}^n|z) + \boldsymbol{\mu}^n\right\|^2, \quad (13)$$

$$\boldsymbol{\theta}^{n+1} = \arg\min_{\boldsymbol{\theta}}\left\|f(\boldsymbol{\theta}|z) - \left(\boldsymbol{x}^{n+1} + \boldsymbol{\mu}^n\right)\right\|^2, \quad (14)$$

$$\boldsymbol{\mu}^{n+1} = \boldsymbol{\mu}^n + \boldsymbol{x}^{n+1} - f\left(\boldsymbol{\theta}^{n+1}|z\right). \quad (15)$$

1) Solving subproblem (13)—In this work, we use PET image reconstruction as an example. In PET image reconstruction, A is the detection probability matrix, with A_{ij} denoting the probability of photons originating from voxel j being detected by detector i [42]. $s \in \mathbb{R}^M$ denotes the expectation of scattered and random events. M is the number of lines of response (LOR). Assuming the measured photon coincidences follow Poisson distribution, the log-likelihood function $L(\boldsymbol{y}|\boldsymbol{x})$ can be explicitly written as

$$L(\boldsymbol{y}|\boldsymbol{x}) = \sum_{i=1}^{M} \log p\left(y_i|\boldsymbol{x}\right) = \sum_{i=1}^{M} y_i \log \bar{y}_i - \bar{y}_i - \log y_i!. \quad (16)$$

Though the measurement data may follow different distributions for other inverse problems, only subproblem (13) needs to be reformulated and the whole DIPRecon framework keeps unchanged. Here subproblem (13) is a penalized image reconstruction problem, and the optimization transfer method [43] was chosen to solve it. As $\boldsymbol{x}$ in $L(\boldsymbol{y}|\boldsymbol{x})$ is coupled together, we first construct a surrogate function $Q_L(\boldsymbol{x}|\boldsymbol{x}^n)$ for $L(\boldsymbol{y}|\boldsymbol{x})$ to decouple the image pixels so that each pixel can be optimized independently. $Q_L(\boldsymbol{x}|\boldsymbol{x}^n)$ is constructed as follows

$$Q_L(\boldsymbol{x}|\boldsymbol{x}^n) = \sum_{j=1}^{n_j} A_{\cdot j}(\hat{x}_{j,\mathrm{EM}}^{n+1}\log x_j - x_j), \quad (17)$$

where $A_{\cdot j} = \sum_{i=1}^{M} A_{ij}$ and $\hat{x}_{j,\mathrm{EM}}^{n+1}$ is calculated by

$$\hat{x}_{j,\mathrm{EM}}^{n+1} = \frac{x_j^n}{A_{\cdot j}} \sum_{i=1}^{M} A_{ij} \frac{y_i}{[\boldsymbol{A}\boldsymbol{x}^n]_i + s_i}. \quad (18)$$

It can be verified that the constructed surrogate function $Q_L(\boldsymbol{x}|\boldsymbol{x}^n)$ fulfills the following two conditions:

$$Q_L(\boldsymbol{x}|\boldsymbol{x}^n) - Q_L(\boldsymbol{x}^n|\boldsymbol{x}^n) \le L(\boldsymbol{y}|\boldsymbol{x}) - L(\boldsymbol{y}|\boldsymbol{x}^n), \quad (19)$$

$$\nabla Q_L(\boldsymbol{x}^n|\boldsymbol{x}^n) = \nabla L(\boldsymbol{y}|\boldsymbol{x}^n). \quad (20)$$

After getting this surrogate function, subproblem (13) can be optimized pixel by pixel. For pixel *j*, the surrogate objective function for subproblem (13) is

$$P\left(x_j|\boldsymbol{x}^n\right) = A_{\cdot j}\left(\hat{x}_{j,\mathrm{EM}}^{n+1}\log x_j - x_j\right) - \frac{\rho}{2}\left[x_j - f(\boldsymbol{\theta}|z)_j^n + \mu_j^n\right]^2. \quad (21)$$

The final update equation for pixel *j* after maximizing (21) is

$$x_j^{n+1} = \frac{1}{2}\left[f(\boldsymbol{\theta}^n|z)_j - \mu_j^n - A_{\cdot j}/\rho\right] + \frac{1}{2}\sqrt{\left[f(\boldsymbol{\theta}^n|z)_j - \mu_j^n - A_{\cdot j}/\rho\right]^2 + 4\hat{x}_{j,\mathrm{EM}}^{n+1}A_{\cdot j}/\rho}. \quad (22)$$

2) Solving subproblem (14)—Subproblem (14) is a nonlinear least square problem and it has the same format as the network training problem shown in (1), with $\boldsymbol{x}_{\mathrm{label}}$ replaced by

$\boldsymbol{x}^{n+1} + \boldsymbol{\mu}^n$. Currently network training is mostly based on first order methods, such as the Adam algorithm [44] and the Nesterov's accelerated gradient (NAG) algorithm [45]. The L-BFGS algorithm is a quasi-newton method, combining a history of updates to approximate the Hessian matrix. It is not widely used in network training as it requires large batch size to accurately calculate the descent direction, which is less effective than first order methods for large-scale applications. In this proposed framework, as only the patient's own prior images are employed as the network input, the data size is much smaller than the traditional network training. In our case, the L-BFGS method is preferred to solve subproblem (14) due to its stability and better performance observed

Algorithm 1

Algorithm for the proposed DIPRecon method.

Input: Maximum iteration number MaxIt, sub-iteration number SubIt1, sub-iteration number SubIt2, network initialization $\boldsymbol{\theta}^0$, Prior image $\boldsymbol{z}$

1: $\boldsymbol{x}^0$,SubIt1 = $f(\boldsymbol{\theta}^0|\boldsymbol{z})$

2: $\boldsymbol{\mu}^0 = \mathbf{0}$

3: **for** $n = 1$ to MaxIt **do**

4: $\boldsymbol{x}^{n,0} = \boldsymbol{x}^{n-1,\text{SubIt1}}$

5: **for** $m = 1$ to SubIt1 **do**

6: $\hat{x}_{j,\text{EM}}^{n,m} = \left[x_j^{n,m-1}/A_{\cdot j}\right]\sum_{i=1}^{M} A_{ij}\frac{y_i}{\left[\boldsymbol{A}\boldsymbol{x}^{n,m-1}\right]_i + s_i}$, where $A_{\cdot j} = \sum_{i=1}^{M} A_{ij}$

7: $x_j^{n,m} = \frac{1}{2}\left[f\left(\boldsymbol{\theta}^{n-1}|z\right)_j - \mu_j^{n-1} - A_{\cdot j}/\rho\right] + \frac{1}{2}\sqrt{\left[f\left(\boldsymbol{\theta}^{n-1}|z\right)_j - \mu_j^n - A_{\cdot j}/\rho\right]^2 + 4\hat{x}_{j,\text{EM}}^{n,m}A_{\cdot j}/\rho}$

8: **end for**

9: $\boldsymbol{x}_{\text{label}}^n = \boldsymbol{x}^{n,\text{SubIt1}} + \boldsymbol{\mu}^{n-1}$

10: Running L-BFGS algorithm SubIt2 iterations to train the network, get $\boldsymbol{\theta}^n = \arg\min_{\boldsymbol{\theta}}\left\|f(\boldsymbol{\theta}|z) - \boldsymbol{x}_{\text{label}}^n\right\|^2$

11: $\boldsymbol{\mu}^n = \boldsymbol{\mu}^{n-1} + \boldsymbol{x}^{n,\text{SubIt1}} - f(\boldsymbol{\theta}^n|\boldsymbol{z})$

12: **end for**

13: **return** $\hat{\boldsymbol{x}} = f\left(\boldsymbol{\theta}^{\text{MaxIt}}|z\right)$

in the experiments. The comparing results are presented in Section. V-A.

In the iterative framework, we run two iterations to solve subproblem (13) and ten iterations to solve subproblem (14). The overall algorithm flowchart is presented in Algorithm 1.

D. Network structure

The network structure employed in this work is based on the modified 3D U-net [13]. The overall network architecture is summarized in Fig. 1. It consists of repetitive applications of 1) 3×3×3 3D convolutional layer, 2) batch normalization (BN) layer, 3) Leaky rectified linear unit (LReLU) layer, 4) 3 × 3 × 3 3D convolutional layer with stride 2 × 2 × 2 for down-sampling, 5) 2 × 2 × 2 bilinear interpolation layer for up-sampling, and 6) identity

mapping layer that adds features from left-side encoder path to the right-side decoder path. In our implementation, there are several modifications compared to the original 3D U-net:

1. using convolutional layer with stride 2 to down-sample the image instead of using max pooling, to construct a fully convolutional network;
2. using skip connections to link the encoder path and decoder path instead of concatenating, to reduce the number of training parameters;
3. using the bi-linear interpolation instead of the deconvolution upsampling, to reduce the checkboard artifact;
4. using leaky ReLU instead of ReLU.

Through the experiments it was found that using 3D convolution for 3D image reconstruction is better than using 2D convolution with multiple axial slices in the input channels. To enable the non-negative constraint on the reconstructed image, a ReLU layer was added before the output. In addition, we found that compared to the network without encoder and decoder path, the U-net structure could save more GPU memory because the spatial size is reduced due to the encoder path.

E. Implementation details and reference methods

To stabilize the network training, before we run the networking training, the intensity of PET images were scaled to the range [0,1]. Due to the non-convexity of neural network training, it is better to assign good initials to the network parameters before trained inside the iterative reconstruction loop. Comparisons between the results with and without pretraining are shown in the supplemental materials. In our implementation, we first ran ML EM algorithm for 60 iterations, and then used it as x_{label} to train the network based on (1) with MR images as network input. This pre-training was run 300 epochs using the L-BFGS algorithm based on minimizing (1). As the penalty parameter has a large impact on the convergence speed, we examined the convergence of the log-likelihood $L(\boldsymbol{y}|f(\hat{\boldsymbol{\theta}}|z))$ to determine the penalty parameter used in practice. As an example, Fig. 2 shows the log-likelihood curve for different penalty parameter ρ for the simulation study mentioned in Section. IV-A. Considering the convergence speed and stability of the likelihood, $\rho = 3\times10^{-3}$ was chosen.

We compared the DIPRecon method with the Gaussian postfiltering method, the penalized reconstruction method based on including a pre-trained neural network in the penalty [24], and the kernel method [5]. One popular and intuitive way of incorporating the trained network into iterative reconstruction framework is include it in the penalty item as first shown in [24]. The objective function is

$$\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{x}} L(\boldsymbol{y}|\boldsymbol{x}) - \beta\left\|\boldsymbol{x} - f(\boldsymbol{\theta}_0|z)\right\|^2. \quad (23)$$

Here $f(\boldsymbol{\theta}_0|\boldsymbol{z})$ is the pre-trained network using noisy PET images as labels and the pre-trained network parameters $\boldsymbol{\theta}_0$ keeps fixed during the reconstruction process. This method is denoted as CNNPenalty method. Interestingly, the objective function shown in (23) becomes

the same as subproblem (13) by replacing $f(\boldsymbol{\theta}_0|\boldsymbol{z})$ with $f(\boldsymbol{\theta}^n|\boldsymbol{z})+\boldsymbol{\mu}^n$. In our implementation, the difference between the CNNPenalty method and the proposed DIPRecon method is that $\boldsymbol{\mu}$ was fixed to 0 and the network updating step (14) was skipped. By this setting, we can understand the effects of updating the network parameters inside the reconstruction loop. For the kernel method, the (i, j)th element of the kernel matrix $\boldsymbol{K}_{\boldsymbol{z}}$ is

$$k_{ij} = \exp\left(-\frac{\left\|\boldsymbol{f}_i - \boldsymbol{f}_j\right\|^2}{2N_f\sigma^2}\right), \quad (24)$$

where $\boldsymbol{f}_i \in \mathbb{R}^{N_f}$ and $\boldsymbol{f}_j \in \mathbb{R}^{N_f}$ represents the feature vectors of voxel i and voxel j from the MR prior image $\boldsymbol{z}$, σ^2 is the variance of the prior image and N_f is the number of voxels in a feature vector. For efficient computation, the kernel matrix was constructed using a K-Nearest-Neighbor (KNN) search in a 7×7×7 search window with $K = 50$. A 3×3×3 local patch was extracted for each voxel to form the feature vector. During image reconstruction, the linear coefficients $\boldsymbol{\theta}$ were computed using iterative update as [46]

$$\boldsymbol{\theta}^{n+1} = \frac{\boldsymbol{\theta}^n}{\left(\boldsymbol{A}\boldsymbol{K}_z\right)^T\mathbf{1}_M}\left[\left(\boldsymbol{A}\boldsymbol{K}_z\right)^T\frac{\boldsymbol{y}}{\left(\boldsymbol{A}\boldsymbol{K}_z\right)\boldsymbol{\theta}^n + \boldsymbol{s}}\right], \quad (25)$$

and the final output image is $\hat{\boldsymbol{x}} = \boldsymbol{K}_{\boldsymbol{z}}\hat{\boldsymbol{\theta}}$. The kernel method is denoted as KMRI in later comparisons.

Currently the code is not optimized and the network training was done in GPU (NVIDIA GTX 1080 Ti) and the image reconstruction was implemented in CPU with 16 threads (Intel Xeon E5–2630 v3). The neural network was implemented using TensorFlow 1.4. The LBFGS algorithm was implemented using the TensorFlow "ScipyOptimizerInterface" to call "L-BFGS-B" method from scipy libary running in CPU mode. 10 previous iterations were used in L-BFGS-B to approximate the Hessian matrix. For the simulation setting, the CPU memory usage for L-BFGS is 2GB and for the real data setting, the CPU memory usage is 7 GB. For the simulation study described in Section. IV-A, the running time is 20 seconds per iteration for the EM method, 23 seconds per iteration for the CNNPenalty method, 30 seconds per iteration for the KMRI method (averaging the kernel matrix calculation time), and 78 seconds per iteration for the DIPRecon method. If the L-BFGS is implemented in GPU mode, the computation time for DIPRecon can be further reduced.

IV. Experimental setup

A. Brain phantom simulation

A 3D brain phantom from BrainWeb [47] was used in the simulation. Corresponding T1 weighted MR image was used as the prior image. The voxel size is 2×2×2 mm^3 and the phantom image size is 128×128×105. The input to the network was cropped to 128×128×96 to reduce GPU memory usage. To simulate mismatches between the MR and PET images,

twelve hot spheres of diameter 16 mm were inserted into the PET image as tumor regions, which are not visible in the MR image. The time activity curves of blood, gray matter, white matter and tumor were simulated based on a two-tissue compartment model, with kinetic parameters the same as those used in [48]. The computer simulation modeled the geometry of a Siemens mCT scanner [49], and the system matrix was generated using the multi-ray tracing method [50]. In this experiment, the last 5 min frame of a one-hour FDG scan was used as the ground-truth image. Noise-free sinogram data were generated by forward-projecting the ground-truth image using the system matrix and the attenuation map. Poisson noise was then introduced to the noise-free data by setting the total count level to be equivalent to last 5 min scan with 5 mCi injection. Uniform random events were simulated and accounted for 30 percent of the noise-free data. Scatters were not included. For quantitative comparison, contrast recovery coefficient (CRC) vs. the standard deviation (STD) curves were plotted based on reconstructions of twenty independent and identically distributed (i.i.d) realizations. The CRC was computed between selected gray matter regions and white matter regions as

$$\mathrm{CRC} = \frac{1}{R}\sum_{r=1}^{R}\left(\frac{\bar{a}_r}{\bar{b}_r} - 1\right) \Big/ \left(\frac{a^{\mathrm{true}}}{b^{\mathrm{true}}} - 1\right). \quad (26)$$

Here R is the number of realizations and is set to 20, $\bar{a}_r = 1/K_a \sum_{k=1}^{K_a} a_{r,k}$ is the average uptake of the gray matter over K_a ROIs in realization r. The ROIs were drawn in both matched gray matter regions and the tumor regions. For the case of matched gray matter, K_a = 10. For the tumor regions, K_a = 12. When choosing the matched gray matter regions, only those pixels inside the predefined 20mm-diameter spheres and containing 80% of gray matter were included. $\bar{b}_r = 1/K_b \sum_{k=1}^{K_b} b_{r,k}$ is the average value of the background ROIs in realization r, and K_b = 37 is the total number of background ROIs. The background ROIs were drawn in the white matter. The background STD was computed as

$$\mathrm{STD} = \frac{1}{K_b}\sum_{k=1}^{K_b} \frac{\sqrt{\frac{1}{R-1}\sum_{r=1}^{R}\left(b_{r,k} - \bar{b}_k\right)^2}}{\bar{b}_k}, \quad (27)$$

where $\bar{b}_k = 1/R \sum_{r=1}^{R} b_{r,k}$ is the average of the background ROI means over realizations.

The network structure for the simulated data set is the same as the network structure shown in Fig. 1, except input and output size are 128 × 128 × 96.

B. Real brain data sets

A 70-minutes dynamic PET scan of a human subject acquired on a Siemens Brain MR-PET scanner after 5 mCi FDG injection was employed in the real data evaluation. The data were reconstructed with an image array of 256×256×153 and a voxel size of 1.25 × 1.25 × 1.25 mm^3. The input to the network was cropped to 192×192×128 to reduce GPU memory usage.

A simultaneous acquired T1-weighted MR image having the same image array and voxel size as the PET image was used as the prior image. Correction factors for randoms, scatters were estimated using the standard software provided by the manufacturer and included during reconstruction. The motion correction was performed in the line-of-response (LOR) space based on the simultaneously acquired MR navigator signal [51]. Attenuation was derived from T1-weighted MR image using the SPM based atlas method [52]. To generate multiple realizations for quantitative analysis, the last 40 minutes PET data were binned together and resampled with a 1/8 ratio to obtain 20 i.i.d. datasets that mimic 5-minutes frames. As the ground truth of the regional uptake is unknown, a hot sphere with diameter 12.5 mm, mimicking a tumor, was added to the PET data (invisible in the MRI image). It simulates the case where MRI and PET information does not match. The TAC of the hot sphere was set to the TAC of the gray matter, so the final TAC of the simulated tumor region is higher than that of the gray matter because of the superposition. The simulated tumor image of the last 40 minutes was forward-projected to generate a set of noise-free sinograms, including detector normalization and patient attenuation. Randoms and scatters from the inserted tumor were not simulated as they would be negligible compared with the scattered and random events from the patient background. Poisson noise was then introduced and finally the tumor sinograms were added to the original patient sinograms to generate the hybrid real data sets. For tumor quantification, images with and without the inserted tumor were reconstructed and the difference was taken to obtain the tumor only image and compared with the ground truth. The tumor contrast recovery (CR) was calculated as

$$\mathrm{CR} = \frac{1}{R}\sum_{r=1}^{R} \bar{l}_r / l_{\mathrm{true}}, \quad (28)$$

where $\bar{l}_r$ is the mean tumor uptake inside the tumor ROI, l_{true} is the ground truth of the tumor uptake, and R is the number of the realizations. For the background, 11 circular ROIs with a diameter of 12.5 mm were drawn in the white matter and the standard deviation was calculated according to (27). The network structure for the real data set is shown in Fig. 1.

V. Results

A. Effect of network settings

To test the effectiveness of the conditional DIP framework, an experiment was performed by using either the uniform random noise or the patient's MR prior image as the network input. ML EM reconstruction of the real brain data at 60^{th} iteration was treated as the label image. 300 epochs were run for network training using the L-BFGS algorithm. Fig. 3(a,b) shows one coronal view of the network output. When the input is random noise, the image is smooth, but some cortex structures cannot be recovered. When the input is MR image, more cortex structures show up. This comparison demonstrates the benefits of including the patient's prior image as the network input. To show the benefit of including the neural network into reconstruction framework, compared to image denoising as shown in the original DIP paper [8], we have also presented the reconstructed image using the proposed DIPRecon framework in Fig. 3(c). Clearly, DIPRecon can recover more cortex details and

generate higher contrast between the white matter and gray matter as compared to the original DIP framework.

We also compared the behaviors of different optimization algorithms under the conditional DIP framework. The Adam, NAG, and L-BFGS algorithms were compared regarding the pre-training process using the real brain data. When comparing different algorithms, we computed the normalized cost value, which is defined as

$$L_n = \frac{\phi^{\text{ref}}_{\text{Adam}} - \phi^n}{\phi^{\text{ref}}_{\text{Adam}} - \phi^1_{\text{Adam}}}, \quad (29)$$

where $\phi^{\text{ref}}_{\text{Adam}}$ and ϕ^1_{Adam} is the cost value defined in (1) after running Adam for 700 iterations and 1 iteration, respectively. Fig. 4 plots the normalized cost value curves for different algorithms. The L-BFGS algorithm is monotonic decreasing while the Adam algorithm is not due to the adaptive learning rate implemented. The NAG algorithm is slower than the other two algorithms. The reason why L-BFGS is faster is due to its using the approximated Hessian matrix, which makes it closer to a second-order optimization algorithm, while both the Adam and NAG methods are first-order methods. The monotonic property is another good advantage of L-BFGS algorithm. Due to the monotonic property, the network output using the L-BFGS algorithm is more stable and less influenced by the image noise when running multiple realizations. Faster convergence speed and better quantitative results are the reasons we chose L-BFGS algorithm to solve subproblem (14) and perform the initial network training.

B. Simulation results

Fig. 5 shows three orthogonal views of the reconstructed images using different methods for the simulated dataset. The kernel method and the DIPRecon method both reveal more cortex structures and have lower noise compared to the EM-plus-filter method and the CNNPenalty method. Compared with the kernel method, the proposed DIPRecon method can recover even more details of the cortices and the white matter regions are cleaner. Furthermore, compared to the kernel method, the tumor uptake using DIPRecon is higher and the tumor shape is closer to the ground truth. In this simulation set-up, there are no tumor signals in the prior MRI image, and the DIPRecon method can still recover PET signals, which is a sign of robustness to potential mismatches between PET and prior images. Besides, by comparing the CNNPenalty method and the proposed DIPRecon method, we can also see that updating the network parameters inside the reconstruction loop can recover more brain structures and reduce the image noise. Fig. 6 shows the CRC-STD curves for different methods. For both the gray matter region and the tumor region, we can see that at fixed STD, the CRC of the DIPRecon method is higher than other methods. This observation quantitatively shows that the DIPRecon method out-performs other methods.

C. Real data results

Fig. 7 shows the reconstructed images and the corresponding MR prior images of the real brain dataset with inserted lesion using different methods. Reconstructed images from the real dataset without inserted lesion are shown in the supplemental material. The high-count images were reconstructed from the combined 40-min scanning for reference. Compared to the EM-plus-filter method and the CNNPenalty method, the kernel method and the DIPRecon method can recover more cortex details and the image noise in the white matter is much reduced. The cortex shape using the DIPRecon method is clearer than the kernel method. For the tumor region which is unobserved in the MR image, the uptake is higher in the DIPRecon method compared with the kernel method. Fig. 8 shows the CR-STD curves for different methods. Clearly the DIPRecon method has the best CR-STD trade-off compared with the other methods.

VI. Discussion

The number of training variables for the U-net structure implemented in this work is around 1.5 million. For both simulation and real data sets, the number of voxels for the network input is more than 1.5 million. Compared to the traditional iterative reconstruction, the number of unknowns is reduced in our proposed framework. During our implementation, we replaced the feature concatenation with feature adding to reduce the network training parameters, performed bi-linear interpolation to replace the deconvolution upsampling, and finally used the L-BFGS algorithm to train the network. All these three changes improved the network performance in our experiments. Finding a network structure with less trainable parameters but stronger representation power is our on-going work.

In this paper we used the PET image reconstruction as an example to demonstrate the effectiveness of the proposed DIPRecon framework. Subproblem (13) was solved using the optimization transfer method, and subproblem (14) was treated as the traditional network training with MSE-based loss function. This proposed framework can also be applied to other image reconstruction problems where a patient-specific image prior is available. All we need to change is the imaging model and likelihood function of the measured data. Here the initial network training was performed based on one single patient dataset. We expect that if the network was pre-trained from other patient data sets, and later on fine-tuned in the DIPRecon framework, the results can be better as the variables in this population-based initial network is more robustly estimated.

Benefits of employing patient's prior images as network input can be observed comparing Fig. 3(a) and Fig. 3(b). Currently we only use one T1-weighted MR as the anatomical prior. More prior information, such as multiple MR images using different sequences and the temporal information from dynamic PET, can be included by adding more input channels. As no pre-defined weighting is needed when combing multiple priors, this proposed DIPRecon framework can make use of the patient's prior information more efficiently. Both KMRI and DIPRecon are constructed based on representation unknown image x using the kernel matrix or CNN. We have calculated likelihood values for the EM, KMRI and DIPRecon methods regarding images shown in Fig. 5. From high to low, the likelihood values are -7.9588×10^7 (EM), -7.9649×10^7 (DIPRecon) and -7.9654×10^7 (KMRI). We

can see that DIPRecon method can better fit the data compared with the KMRI method, which is based on the strong approximation ability of CNN.

As for the ADMM algorithm used, there is no convergence theory due to non-convexity of the neural network. The reason to use ADMM is to de-couple the image update step and the network training step. The network training step (subproblem 14) needs more update steps than the iterative reconstruction step (subproblem 13). If they are coupled together, every time updating the network parameters, we need to compute the forward and backward projections. As the computation of projections are very time-consuming for fully-3D PET, decoupling these two steps will save more computational time. Furthermore, after using the ADMM algorithm, subproblem 13 is a traditional penalized reconstruction and can reuse current image reconstruction packages, which makes the DIPRecon method adaptable to current iterative frameworks. Compared to the DIP framework, the proposed DIPRecon framework uses raw sinogram data and the objective function is constructed from imaging physics, which exploits more useful information compared with using noisy images as the training labels. This can be observed comparing Fig. 3(b) and Fig. 3(c).

Compared to the simulation results shown in Fig. 5, the real data results of the DIPRecon method shown in Fig. 7 are a little blurred. Two possible reasons are: (1) the number of voxels for the real data is larger than the simulation study and more iterations are needed to achieve the similar sharpness as the simulation data; (2) in the simulation, the MR boundary are perfectly matched with PET images. For real datasets, there might be boundary mismatches between T1 MR and PET images. Currently we only tested our methods on simultaneous PET-MR scanning, but does not perform experiments based on data sets acquired during different scans, in which cases large registration errors might arise. One possible solution is to couple a registration network, such as the spatial transformer network [53], to the U-net to perform additional registration. Testing and adjusting the proposed framework regarding large registration errors are our future research direction. Finally, our evaluations are based on FDG tracers, and the FDG brain uptake has similar patterns as the T1 MR images. Further evaluations based on data sets with other tracers are needed to test the robustness of the proposed framework.

VII. Conclusion

In this work, we propose a framework to include the personalized deep neural network into the iterative reconstruction method. Prior training image pairs are not needed in this process, but only the patient’s own prior images. PET image reconstruction was employed to demonstrate the effectiveness of the proposed DIPRecon framework. Both simulation and real brain data sets show that this proposed personalized medical imaging reconstruction framework performs better than the Gaussian filter, the penalized reconstruction using a neural network penalty, and the kernel method, in terms of contrast recovery vs. noise trade-offs. Future work will focus on finding better network structures, evaluations using more clinical data sets with different tracers, and testing the robustness of the proposed method to registration errors.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgements

This work was supported by the National Institutes of Health under grant number R01AG052653 and P41EB022544.

References

[1]. Ronneberger O, Fischer P, and Brox T, "U-net: Convolutional networks for biomedical image segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2015, pp. 234–241.

[2]. Ren S, He K, Girshick R et al., "Faster r-cnn: Towards real-time object detection with region proposal networks," in Advances in neural information processing systems, 2015, pp. 91–99.

[3]. Gong K, Yang J, Kim K et al., "Attenuation correction for brain PET imaging using deep neural network based on Dixon and ZTE MR images," Physics in Medicine & Biology, vol. 63, no. 12, 2018.

[4]. Chen G-H, Tang J, and Leng S, "Prior image constrained compressed sensing (piccs): a method to accurately reconstruct dynamic ct images from highly undersampled projection data sets," Medical physics, vol. 35, no. 2, pp. 660–663, 2008. [PubMed: 18383687]

[5]. Wang G and Qi J, "PET image reconstruction using kernel method," IEEE Transactions on Medical Imaging, vol. 34, no. 1, pp. 61–71, 2015. [PubMed: 25095249]

[6]. Gnahm C and Nagel AM, "Anatomically weighted second-order total variation reconstruction of 23na mri using prior information from 1h mri," NeuroImage, vol. 105, pp. 452–461, 2015. [PubMed: 25462793]

[7]. Bai B, Li Q, and Leahy RM, "Magnetic resonance-guided positron emission tomography image reconstruction," Seminars in nuclear medicine, vol. 43, no. 1, pp. 30–44, 2013. [PubMed: 23178087]

[8]. Ulyanov D, Vedaldi A, and Lempitsky V, "Deep image prior," arXiv preprint arXiv:1711.10925, 2017.

[9]. Radford A, Metz L, and Chintala S, "Unsupervised representation learning with deep convolutional generative adversarial networks," arXiv preprint arXiv:1511.06434, 2015.

[10]. Mirza M and Osindero S, "Conditional generative adversarial nets," arXiv preprint arXiv: 1411.1784, 2014.

[11]. Isola P, Zhu J-Y, Zhou T et al., "Image-to-image translation with conditional adversarial networks," arXiv preprint, 2017.

[12]. Zhu J-Y, Park T, Isola P et al., "Unpaired image-to-image translation using cycle-consistent adversarial networks," arXiv preprint arXiv:1703.10593, 2017.

[13]. Çiçek Ö, Abdulkadir A, Lienkamp SS et al., "3D U-Net: learning dense volumetric segmentation from sparse annotation," in International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2016, pp. 424–432.

[14]. Zhu C, Byrd RH, Lu P et al., "Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization," ACM Transactions on Mathematical Software (TOMS), vol. 23, no. 4, pp. 550–560, 1997.

[15]. Gong K, Zhou J, Tohme M et al., "Sinogram blurring matrix estimation from point sources measurements with rank-one approximation for fully 3D PET," IEEE transactions on medical imaging, vol. 36, no. 10, pp. 2179–2188, 2017. [PubMed: 28613163]

[16]. Gindi G, Lee M, Rangarajan A et al., "Bayesian reconstruction of functional images using anatomical information as priors," IEEE Transactions on Medical Imaging, vol. 12, no. 4, pp. 670–680, 1993. [PubMed: 18218461]

[17]. Bowsher JE, Yuan H, Hedlund LW et al., “Utilizing MRI information to estimate F18-FDG distributions in rat flank tumors,” in Nuclear Science Symposium Conference Record, 2004 IEEE, vol. 4. IEEE, 2004, pp. 2488–2492.

[18]. Nuyts J, “The use of mutual information and joint entropy for anatomical priors in emission tomography,” in 2007 IEEE Nuclear Science Symposium Conference Record, vol. 6. IEEE, 2007, pp. 4149–4154.

[19]. Tang J and Rahmim A, “Bayesian PET image reconstruction incorporating anato-functional joint entropy,” Physics in Medicine and Biology, vol. 54, no. 23, p. 7063, 2009. [PubMed: 19904028]

[20]. Somayajula S, Panagiotou C, Rangarajan A et al., “PET image reconstruction using information theoretic anatomical priors,” IEEE Transactions on Medical Imaging, vol. 30, no. 3, pp. 537–549, 2011. [PubMed: 20851790]

[21]. Cheng-Liao J and Qi J, “Pet image reconstruction with anatomical edge guided level set prior,” Physics in Medicine & Biology, vol. 56, no. 21, p. 6899, 2011. [PubMed: 21983558]

[22]. Vunckx K, Atre A, Baete K et al., “Evaluation of three mri-based anatomical priors for quantitative pet brain imaging,” IEEE transactions on medical imaging, vol. 31, no. 3, pp. 599–612, 2012. [PubMed: 22049363]

[23]. Chan C, Fulton R, Barnett R et al., “Postreconstruction nonlocal means filtering of whole-body pet with an anatomical prior,” IEEE Transactions on medical imaging, vol. 33, no. 3, pp. 636–650, 2014. [PubMed: 24595339]

[24]. Wang S, Su Z, Ying L et al., “Accelerating magnetic resonance imaging via deep learning,” in Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on IEEE, 2016, pp. 514–517.

[25]. Kang E, Min J, and Ye JC, “A deep convolutional neural network using directional wavelets for low-dose x-ray ct reconstruction,” arXiv preprint arXiv:1610.09736, 2016.

[26]. Chen H, Zhang Y, Zhang W et al., “Low-dose ct via convolutional neural network,” Biomedical optics express, vol. 8, no. 2, pp. 679–694, 2017. [PubMed: 28270976]

[27]. Gong K, Guan J, Liu C-C et al., “PET image denoising using a deep neural network through fine tuning,” IEEE Transactions on Radiation and Plasma Medical Sciences, 2018.

[28]. Wu D, Kim K, El Fakhri G et al., “Iterative low-dose ct reconstruction with priors trained by artificial neural network,” IEEE transactions on medical imaging, vol. 36, no. 12, pp. 2479–2486, 2017. [PubMed: 28922116]

[29]. Gong K, Guan J, Kim K et al., “Iterative PET image reconstruction using convolutional neural network representation,” arXiv preprint arXiv:1710.03344, 2017.

[30]. Gupta H, Jin KH, Nguyen HQ et al., “Cnn-based projected gradient descent for consistent image reconstruction,” arXiv preprint arXiv:1709.01809, 2017.

[31]. Sun J, Li H, Xu Z et al., “Deep admm-net for compressive sensing mri,” in Advances in Neural Information Processing Systems, 2016, pp. 10–18.

[32]. Hammernik K, Klatzer T, Kobler E et al., “Learning a variational network for reconstruction of accelerated mri data,” Magnetic resonance in medicine, 2017.

[33]. Adler J and Öktem O, “Learned primal-dual reconstruction,”¨ IEEE Transactions on Medical Imaging, 2018.

[34]. Chen H, Zhang Y, Zhang W et al., “Learned experts’ assessment-based reconstruction network (” learn”) for sparse-data ct,” arXiv preprint arXiv:1707.09636, 2017.

[35]. Wu D, Kim K, Dong B et al., “End-to-end abnormality detection in medical imaging,” arXiv preprint arXiv:1711.02074, 2017.

[36]. Qin C, Schlemper J, Caballero J et al., “Convolutional recurrent neural networks for dynamic mr image reconstruction,” arXiv preprint arXiv:1712.01751, 2017.

[37]. Elad M and Aharon M, “Image denoising via sparse and redundant representations over learned dictionaries,” IEEE Transactions on Image processing, vol. 15, no. 12, pp. 3736–3745, 2006. [PubMed: 17153947]

[38]. Ravishankar S and Bresler Y, “Mr image reconstruction from highly undersampled k-space data by dictionary learning,” IEEE transactions on medical imaging, vol. 30, no. 5, pp. 1028–1041, 2011. [PubMed: 21047708]

[39]. Xu Q, Yu H, Mou X et al., “Low-dose x-ray ct reconstruction via dictionary learning,” IEEE Transactions on Medical Imaging, vol. 31, no. 9, pp. 1682–1697, 2012. [PubMed: 22542666]

[40]. Chen S, Liu H, Shi P et al., “Sparse representation and dictionary learning penalized image reconstruction for positron emission tomography,” Physics in Medicine & Biology, vol. 60, no. 2, p. 807, 2015. [PubMed: 25565039]

[41]. Wang X, Girshick R, Gupta A et al., “Non-local neural networks,” arXiv preprint arXiv: 1711.07971, 2017.

[42]. Qi J, Leahy RM, Cherry SR et al., “High-resolution 3D Bayesian image reconstruction using the microPET small-animal scanner,” Physics in medicine and biology, vol. 43, no. 4, p. 1001, 1998. [PubMed: 9572523]

[43]. Wang G and Qi J, “Penalized likelihood pet image reconstruction using patch-based edge-preserving regularization,” IEEE transactions on medical imaging, vol. 31, no. 12, pp. 2194–2204, 2012. [PubMed: 22875244]

[44]. Kingma D and Ba J, “Adam: A method for stochastic optimization,” arXiv preprint arXiv: 1412.6980, 2014.

[45]. Nesterov Y, “A method of solving a convex programming problem with convergence rate o (1/k2),” in Soviet Mathematics Doklady, vol. 27, no. 2, 1983, pp. 372–376.

[46]. Hutchcroft W, Wang G, Chen KT et al., “Anatomically-aided PET reconstruction using the kernel method,” Physics in Medicine and Biology, vol. 61, no. 18, p. 6668, 2016. [PubMed: 27541810]

[47]. Aubert-Broche B, Griffin M, Pike GB et al., “Twenty new digital brain phantoms for creation of validation image data bases,” IEEE transactions on medical imaging, vol. 25, no. 11, pp. 1410–1416, 2006. [PubMed: 17117770]

[48]. Gong K, Cheng-Liao J, Wang G et al., “Direct Patlak reconstruction from dynamic PET data using the kernel method with MRI information based on structural similarity,” IEEE Transactions on Medical Imaging, vol. 37, no. 4, pp. 955–965, 2018. [PubMed: 29610074]

[49]. Jakoby B, Bercier Y, Conti M et al., “Physical and clinical performance of the mCT time-of-flight PET/CT scanner,” Physics in medicine and biology, vol. 56, no. 8, p. 2375, 2011. [PubMed: 21427485]

[50]. Zhou J and Qi J, “Fast and efficient fully 3D PET image reconstruction using sparse system matrix factorization with GPU acceleration,” Physics in medicine and biology, vol. 56, no. 20, p. 6739, 2011. [PubMed: 21970864]

[51]. Catana C, Benner T, van der Kouwe A et al., “Mri-assisted pet motion correction for neurologic studies in an integrated mr-pet scanner,” Journal of Nuclear Medicine, vol. 52, no. 1, pp. 154–161, 2011. [PubMed: 21189415]

[52]. Izquierdo-Garcia D, Hansen AE, Förster S et al., “An spm8-based approach for attenuation correction combining segmentation and nonrigid template formation: application to simultaneous pet/mr brain imaging,” Journal of Nuclear Medicine, vol. 55, no. 11, pp. 1825–1830, 2014. [PubMed: 25278515]

[53]. Jaderberg M, Simonyan K, Zisserman A et al., “Spatial transformer networks,” in Advances in neural information processing systems, 2015, pp. 2017–2025.

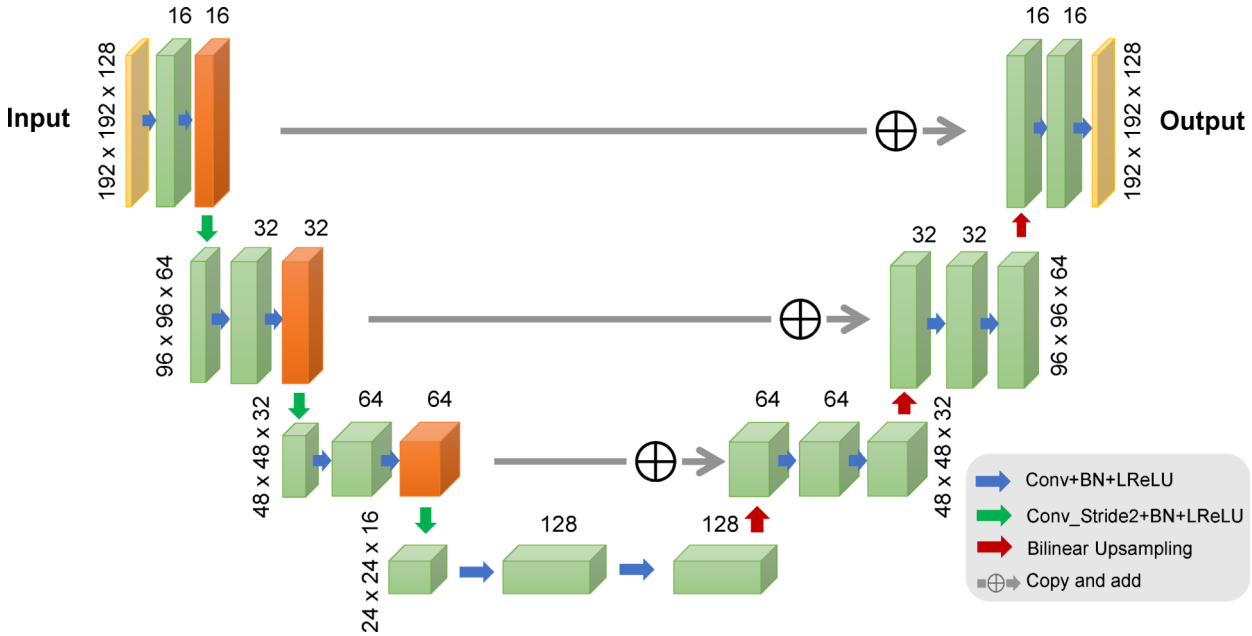

Fig. 1:

The schematic plot of the Network structure used in this work. The spatial input size for each layer is based on the real data experiment described in IV-B.

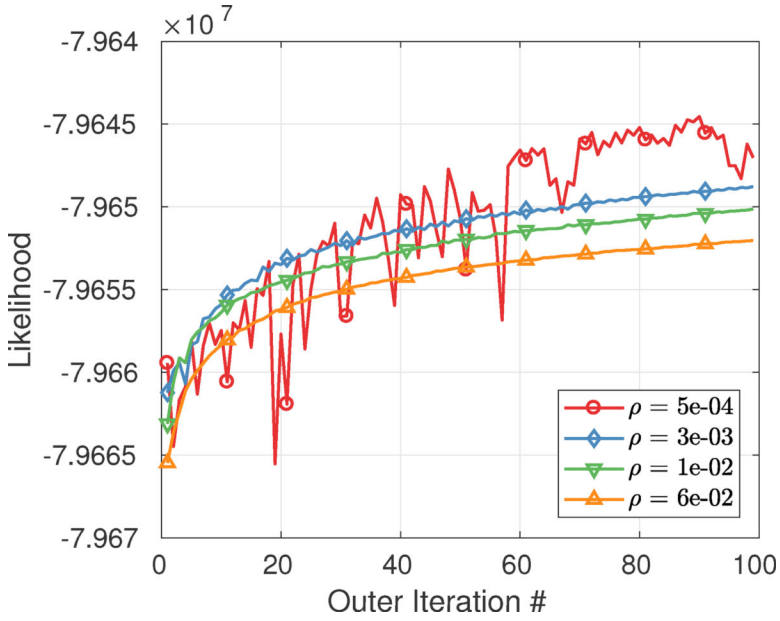

Fig. 2:

The effect of penalty parameter ρ on the likelihood $L(\mathbf{y}|f(\hat{\boldsymbol{\theta}}|z))$ based on the simulation data set described in Section IV-A.

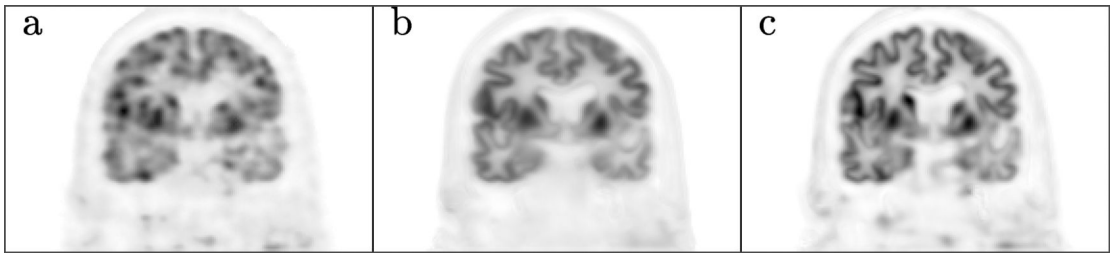

Fig. 3:

(a,b) One coronal view of the network output according to (a) the DIP framework using random noise as network input, (b) conditional DIP framework using the Patient's own MR image as network input and (c) the proposed DIPRecon framework using the Patient's MR image as network input. Images shown are based on the real data set described in Section IV-B.

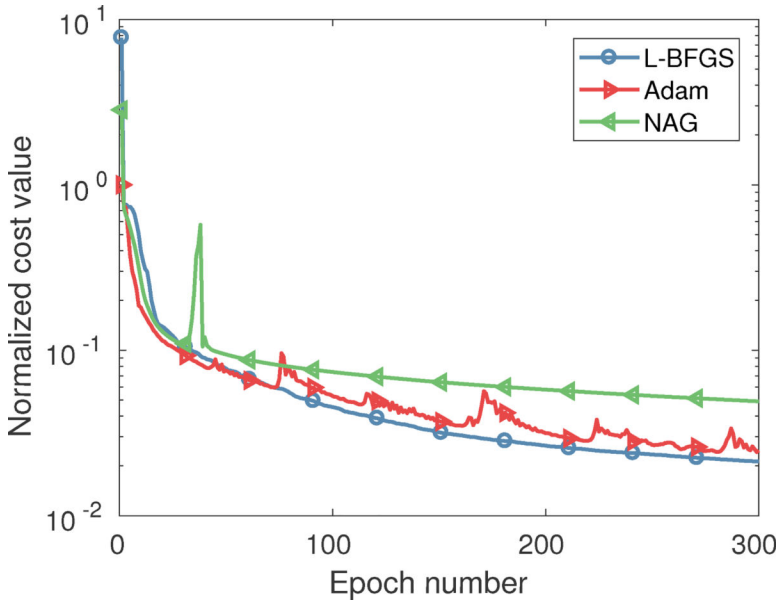

Fig. 4:

Comparison of the normalized likelihood for the Adam, Nesterov's accelerated gradient (NAG) and L-BFGS algorithms based on the real brain data set described in Section IV-B.

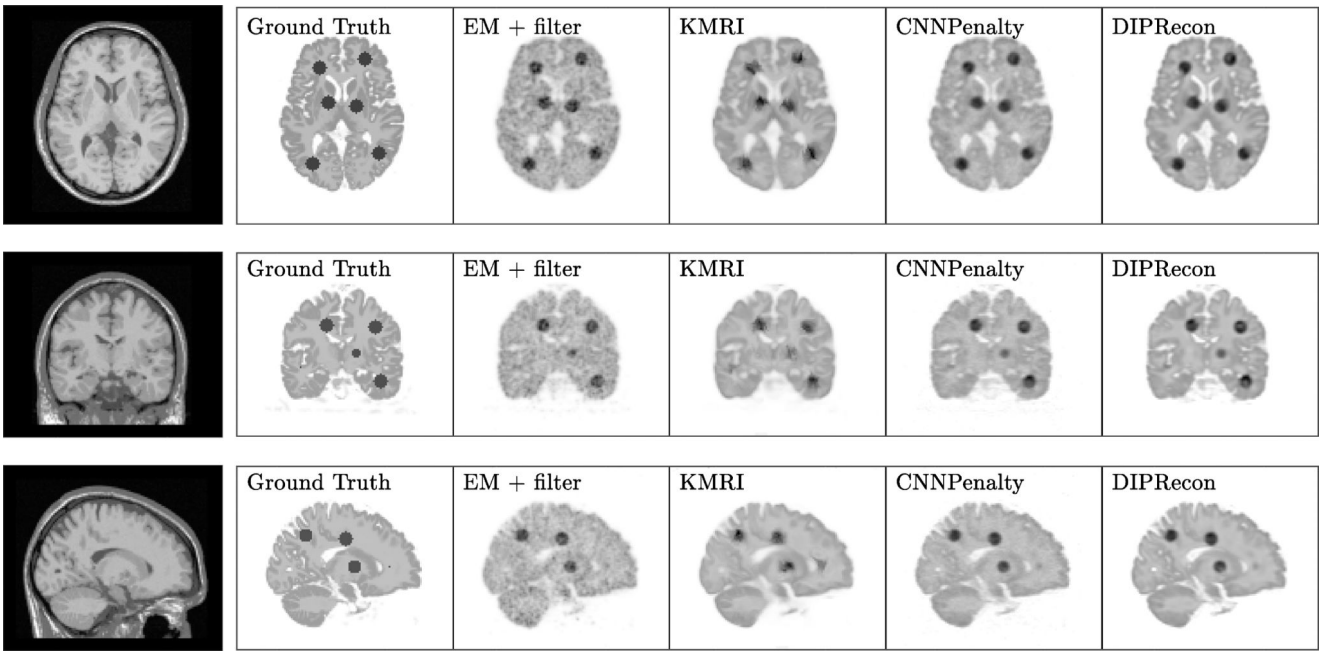

Fig. 5:

Three orthogonal slices of the reconstructed image using different methods for the simulation dataset. The iteration number is 100 for all methods. The first column is the corresponding MR prior image.

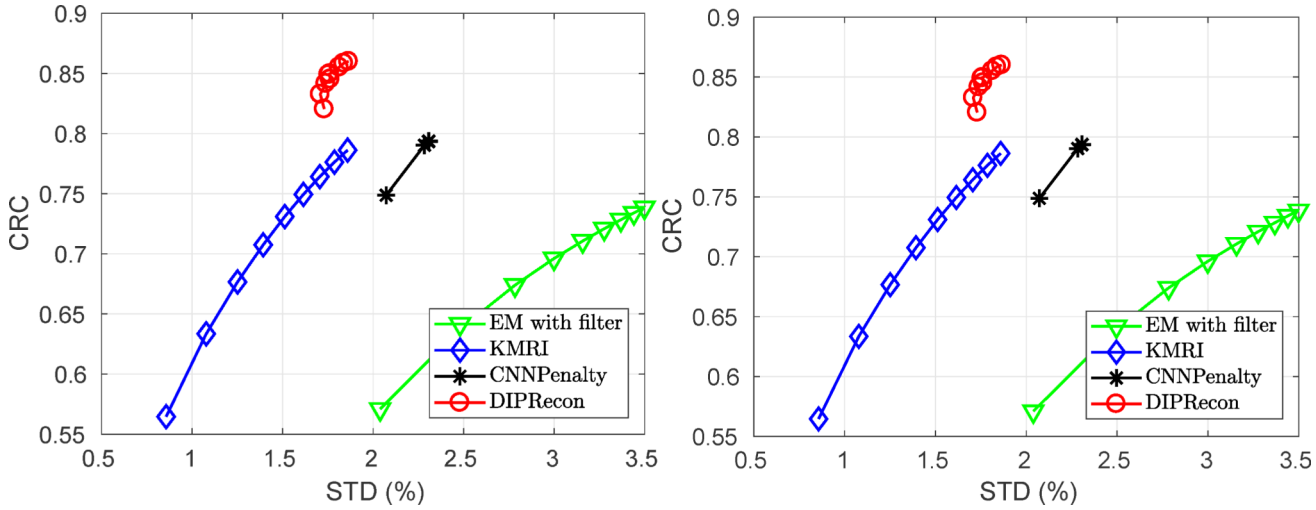

Fig. 6:

CRC-STD curves at the gray matter region (left) and the tumor region (right) for the reconstruction using the simulation dataset. Markers are plotted every ten iterations. The lowest point corresponding to the 20^{th} iteration.

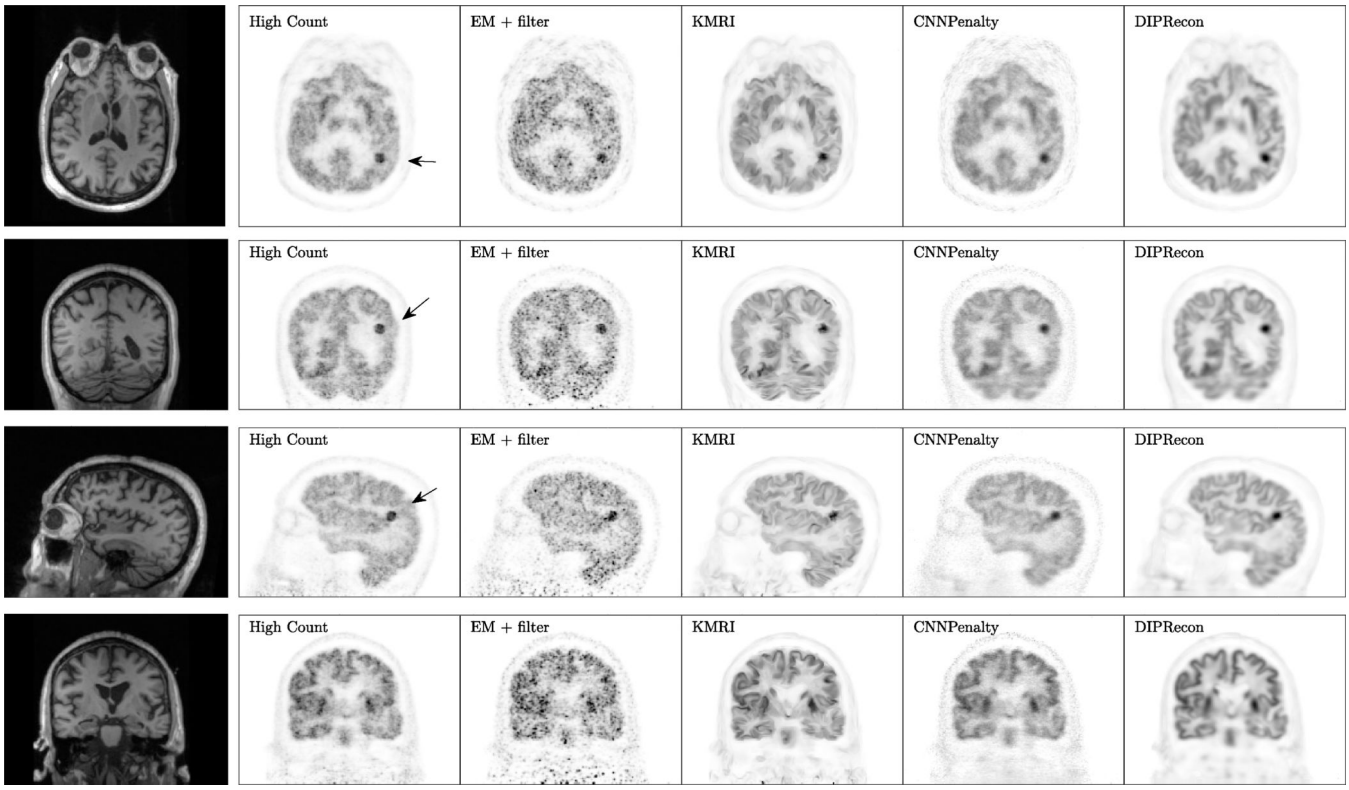

Fig. 7:

First three rows show three orthogonal slices through the simulated-tumor region for the real brain dataset. Arrows in the high-count image (40 min scanning) indicate the simulated tumor. The last row shows one coronal view which contains more cortex structures. The first column is the corresponding MR prior image. The iteration number is 100 for EM-plus-filter and the proposed DIPRecon method, 50 for the CNNPenalty method, and is 120 for the KMRI method.

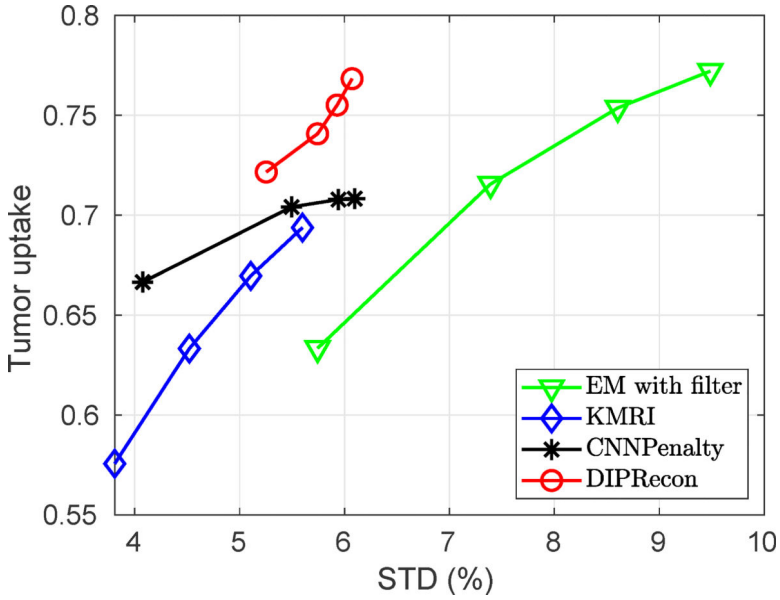

Fig. 8:

The CR-STD plot of the tumor uptake in the real brain data set. Markers are generated for every ten iterations for CNNPenalty method and every twenty iterations for other methods. For EM-plus-filter and DIPRecon method, the lowest point corresponding to the 40^{th} iteration. For KMRI method, the lowest point corresponding to the 60^{th} iteration. For CNNPenalty method, the lowest point corresponding to the 10^{th} iteration.