This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2020.2989073, IEEE Transactions on Radiation and Plasma Medical Sciences

Pseudo CT Image Synthesis and Bone Segmentation from MR Images Using Adversarial Networks with Residual Blocks for MR-Based Attenuation Correction of Brain PET Data

Li Tao, Jonathan Fisher, Emily Anaya, Xin Li, and Craig S. Levin

***Abstract*—For photon attenuation correction, current PET/MR imaging systems typically use methods based on MR image segmentation with subsequent assignment of empirical attenuation coefficients in PET image reconstruction. Delineation of bone in MR images has been challenging, especially in the head and neck areas, due to the difficulty of separating bone from air. In this work, we study deep learning techniques that assist the MR-based attenuation correction (MRAC) process for PET/MR systems, with focus on the brain region. We use a generative adversarial network (GAN) with residual blocks in a conditional setting for this task. We studied the performance of the designed network on image translation and segmentation tasks, which are essential for MRAC. For both tasks, the network generates pseudo CT images that resemble real CT images with normalized pixel value difference of around 5% and structural similarity (SSIM) index of around 0.8.**

***Index Terms*—PET/MR, MR-based attenuation map, image translation and segmentation, deep learning, generative adversarial network (GAN).**

I. Introduction

Positron emission tomography systems combined with magnetic resonance imaging (PET/MR) have shown promise to provide simultaneous molecular and morphological evaluation of a variety of diseases [1], [2]. In order to acquire accurate quantitative PET images from a PET/MR system, MR-based photon attenuation correction (MRAC) methods have been actively researched and developed [3], [4].

Most current PET/MR systems use methods based on MR image segmentation and subsequent assignment of empirical attenuation coefficients for MRAC [5], [6]. The performance of these methods can be significantly affected by segmentation and tissue classification inaccuracy. For example, delineation of bone structures in MR images is challenging due to the difficulty of separating bone from air, especially in the head and neck areas. Therefore special MR sequences, such as ultrashort TE (UTE) and zero echo time (ZTE) sequences, are often necessary for segmentation-based MRAC methods [7], [8], [9], [10]. Atlas-based MRAC methods, which rely on computed tomography (CT) generated attenuation maps and co-registration of MR and CT images, are also studied and used [11]. Commercial MRAC products used for daily clinics typically use segmentation-based or atlas-based methods as discussed in [6], [11].

With the widespread use of deep neural networks in medical imaging related tasks [12], deep learning based MRAC methods have been studied in recent years [13], [14], [15]. Most of these methods use a specially designed convolutional neural network (CNN) to generate pseudo-CT images from MR images in order to achieve the MRAC task [16], [17], [18]. Researchers have also used a deep learning network to reduce the noise of maximum-likelihood reconstruction of attenuation and activity (MLAA) generated attenuation maps and in turn to improve the quality of mu-maps for PET/MR systems [19].

In this work we follow this route and study a deep learning based MRAC method, which can facilitate the automatic generation of attenuation maps directly from MR images for PET/MR systems. Specifically, we choose to use a generative adversarial network (GAN) in a conditional setting (cGAN).

GANs have been widely studied and used for various applications in recent years, especially for image processing and generation related tasks [20]. They have also been studied for medical image synthesis [21], [22], [23], [24]. Conditional GANs feed input examples, rather than random noise in the non-conditional case, into the generator. Conditional GANs have shown outstanding performance in image translation tasks including colorization, style conversion, and photo synthesis [25].

Since MRAC is essentially an image translation problem, i.e., translating MR images to corresponding attenuation maps, cGANs show great potential for solving this problem. The key features of cGANs can improve its performance for image translation tasks as compared to simple Autoencoder [26] or U-net [27] networks, which are widely used for deep learning-based MRAC [16], [17], [18]. For example, adding skip connections in a cGAN network can avoid the bottleneck of information flow that is typically seen in an Autoencoder network. In addition, adopting GAN loss instead of a simple L1 / L2 loss (normally used for U-net) helps to preserve high spatial frequency information (fine structures) in images.

In this work, we will study the feasibility of using a cGAN network for MRAC. We study the performance of the designed network in different tasks that are essential for MRAC, including generating pseudo-CT images from MR images, and segmenting bone in MR images.

L. Tao, J. Fisher, E. Anaya and C. S. Levin are with the Molecular Imaging Instrumentation Laboratory, Stanford University, Stanford, CA, 94305 USA e-mail: cslevin@stanford.edu.

X. Li is with the Center for Gamma-Ray Imaging, University of Arizona, Tucson, AZ, 85721 USA.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2020.2989073, IEEE Transactions on Radiation and Plasma Medical Sciences

Compared to other work applying adversarial networks for MRAC [22], [23], [24], [28], in this manuscript, we have designed a cGAN with a U-net shape generator containing residual blocks [29]. The combination of U-net and residual blocks can facilitate smoother information flow between the input and output of the network, as well as within each encoder/decoder layer. In addition, when testing the performance of the network, in addition to using regular CT and MR images, we have also used patient data on both PET/CT and PET/MR scanners to develop our approach. Since the MRAC task is mainly needed by PET/MR scanners, testing the network's performance on real patient data acquired from a PET/MR scanner can better resemble the real application scenario. Lastly, we have tested another possible application of the cGAN network for MRAC-related task, i.e., bone segmentation in ZTE MR images. For brain PET/MR scans, accurate delineation of bone is the most critical aspect to achieve accurate PET photon attenuation correction. The network's capability of segmenting bone can lead to new ways of using it for MRAC.

II. Materials and Methods

A. Network Structure

The workflow for a cGAN network is illustrated in Fig. 1. It is separated into a generator (G) and a discriminator (D). The generator conditions on the input (x) and tries to generate a fake example (G(x)) that resembles the label (y). The discriminator is trained to distinguish between the fake example (G(x)) and the real label (y). In this conditional setting, the real input example (x) is fed into both the generator and discriminator. During training, the two networks are trained simultaneously.

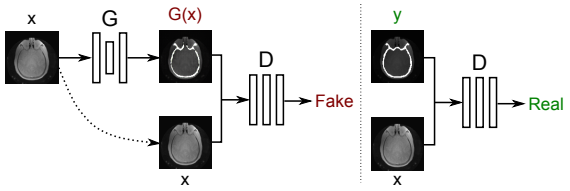

Fig. 1. Illustration for a conditional generative adversarial network (cGAN), with "G" representing the generator, "D" representing the discriminator, "x" representing the input, "y" representing the label, and "G(x)" representing the output from the generator (fake example).

The network architecture adopted in this work is inspired by the *pix2pix* network [25] and *FusionNet* [30], where we inserted residual blocks into a cGAN network. We also only focused on using a 2D network in this study. The network is implemented with TensorFlow [31]. The detailed network structure and the objective functions are described below.

1) Generator: The generator network structure is illustrated in Fig. 2. We used a U-net [27] structure for the generator network with 8 encoder layers and 8 decoder layers. Each encoder layer consists of a residual block, a downsampling convolution (Conv) layer, an optional batch normalization (BatchNorm) layer [32], and a leaky rectified linear unit (ReLU) activation layer. The residual block includes two sets of convolution-leaky ReLU-BatchNorm layers. The convolution layer in the residual block uses 4×4 filters with stride 1. The residual block adds an extra skip connection between the input and output via element-wise addition. The downsampling Conv layer also uses 4×4 filters but with stride 2. All leaky ReLU layers adopt a slope of 0.2.

Each decoder layer consists of a residual block, an upsampling deconvolution (Deconv) layer, an optional BatchNorm layer, and a ReLU (non-leaky) activation layer. The residual block is the same as in the encoder. The Deconv layer uses 4×4 filters with stride 2. A dropout layer [33] with a keep probability of 50% is added to the first three decoder layers. A tanh activation function is applied after the last decoder layer to generate the final output. Skip connections are adopted in the U-net structure to connect the output of the encoder to the corresponding input of the decoder via concatenation in the third dimension.

The input image to the generator is first resized to 256×256 and has one color channel (gray-scale image). The output of the generator has the same dimensions as the input image. The inner latent space has dimensions of 1×1×512. The actual input and output dimensions for each encoder and decoder layer are marked on the top of Fig. 2, and example input and output dimensions of different layers are illustrated on the bottom of Fig. 2. Only the downsampling Conv layer and the upsampling Deconv layer change the dimensions of the input, which respectively shrink the input size by 2 and expand the input size by 2. All the other layers do not change the dimensions of the input. For the skip connections, concatenation in the third dimension expands the third dimension by a factor of 2, while element-wise addition does not change the dimensions of the input. The number of convolution filters (kernels) for the 8 encoder layers are: 64 - 128 - 256 - 512 - 512 - 512 - 512 - 512. The number of deconvolution filters (kernels) for the 8 decoder layers are: 512 - 512 - 512 - 512 - 256 - 128 - 64 - 3.

2) Discriminator: The discriminator network structure is illustrated in Fig. 3. The discriminator has 5 encoder layers. The first three encoders consist of a downsampling convolution (Conv1) layer (4×4 filter, stride 2), an optional BatchNorm layer, and a leaky ReLU activation layer (slope = 0.2). Encoder 4 includes a convolution (Conv2) layer with 4×4 filter and stride 1, a BatchNorm layer, and a leaky ReLU layer (slope = 0.2). The last encoder uses the same Conv2 layer with 4×4 filter and stride 1, and adopts a sigmoid activation function to generate the final output.

The input and label (or alternatively the output from generator) images are both resized to 256×256 and fed into the discriminator network. The two images are concatenated in the third dimension, expanding this dimension by a factor of 2. The final output of the discriminator has a dimension of 30×30×1, which is used to calculate the objective functions. The actual input and output dimensions of each encoder layer are marked on the left side of Fig. 3, and example input and output dimensions of different layers are illustrated on the right side of Fig. 3. Only Conv1 layer shrinks the input dimensions by 2. All the other layers do not change the dimensions of the input. The numbers of convolution filters (kernels) for the 5 encoder layers are: 64 - 128 - 256 - 512 - 1.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2020.2989073, IEEE Transactions on Radiation and Plasma Medical Sciences

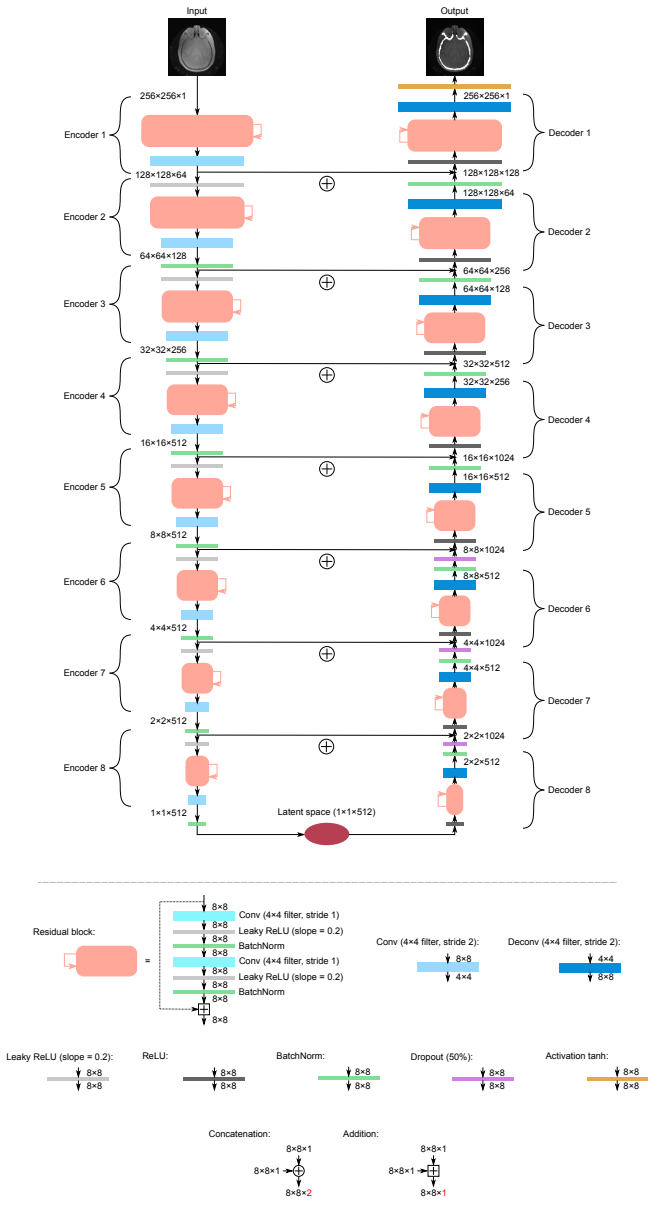

Fig. 2. Illustration for the generator network structure, with the actual input and output dimensions for each encoder/decoder marked on the top, and the example input and output dimensions for different layers illustrated on the bottom.

3) Objective functions: Here we adopt the same symbols as in Fig. 1, and let x denote the input image, y denote the label image, $G(x)$ represent the output from the generator, $D(x, y)$ represent the output from the discriminator when feeding the input (x) and label (y) into the network (predicting true), and $D(x, G(x))$ represent the output from the discriminator when feeding the input (x) and generator output $(G(x))$ into it (predicting fake). We also use $E[X]$ to denote the mean/expectation of all the elements in matrix X. For example, the output from the discriminator (e.g., $D(x, y)$) has a dimension of 30×30×1. With the mean/expectation operation ($E[D(x, y)]$), it was averaged across all the 30×30 elements to give one single number.

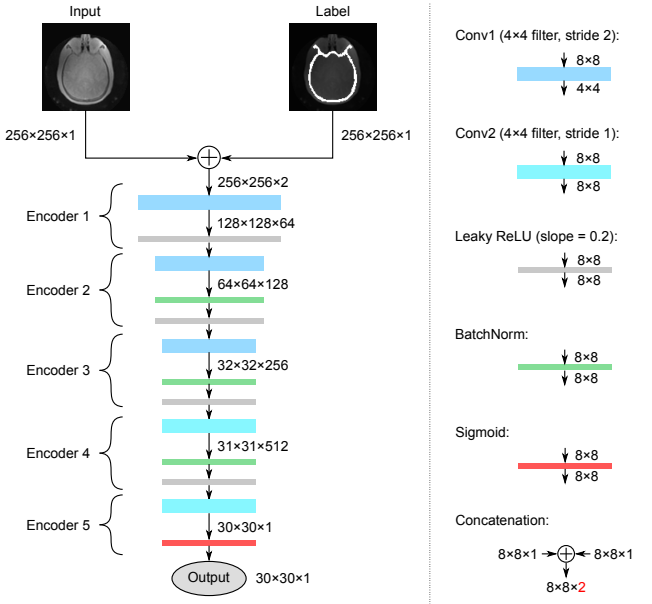

Fig. 3. Illustration for the discriminator network structure, with the actual input and output dimensions for each encoder marked on the left, and the example input and output dimensions for different layers illustrated on the right.

The discriminator objective function is then defined as:

$$L_{dis} = E[\log(D(x, y)) + \log(1 - D(x, G(x)))]. \quad (1)$$

The discriminator is trained to maximize the probability of predicting true $(D(x, y))$ and minimize the probability of predicting fake $(D(x, G(y)))$, so that the objective function is maximized.

The generator objective function is defined as a weighted sum of the cGAN loss and the L1 distance between the output and label images, which can be written as:

$$\begin{aligned} L_{gen} &= a_1 \times L_{gen-cGAN} + a_2 \times L_{gen-L1} \\ &= a_1 \times E[\log(D(x, G(x))) + a_2 \times E[||G(x) - y||_1]. \end{aligned} \quad (2)$$

Here we use $||X-Y||_1$ to represent the L1 distance between matrix X and Y. The generator is trained to maximize the probability of the discriminator predicting fake $(D(x, G(x)))$, and at the same time minimize the L1 distance between the output and label images, so that the overall objective function is maximized. Currently we choose to use a weight combination of $a_1 = 1$ and $a_2 = 100$.

B. Datasets

We used three different datasets to train the cGAN network and test its performance for different MRAC-related tasks. The datasets are described below.

1) Co-registered MR and CT images: Input images are clinical T1-weighted MR brain images from a 3 Tesla GE Discovery 750T scanner. Label images are co-registered CT images of the same patients in the same region. The images were acquired from a total of 11 patients with 100 slices

sampled for each patient from the axial view. Any metal artifacts in the CT images were manually removed. The images from 2 patients (randomly picked) were used as the test set. The rest of the images were used as the training set. We use this dataset to test if the cGAN network can convert MR images (acquired with a regular MR scanner) to CT images (acquired with a regular CT scanner).

2) Pairing PET/MR and PET/CT images: Input images are clinical brain MR images using a two-point Dixon MR sequence (water only) on a Signa PET/MR (GE Healthcare) scanner. Label images are pairing CT images of the same patients in the same region acquired on Siemens healthcare mCT PET/CT scanner. The MR and CT images were manually co-registered using 3D Slicer and MATLAB and the metal artifacts in the CT images were removed. The images were acquired from a total of 10 patients with 100 slices sampled for each patient from the axial view. The images from 2 patients (randomly picked) were used as the test set. The rest of the images were used as the training set. We use this dataset to test if the cGAN network can convert MR images (acquired with a PET/MR scanner) to CT images (acquired with a PET/CT scanner).

3) ZTE MR images: Input images are zero echo time (ZTE) MR images of a patient's brain. Label images are corresponding images with bone manually segmented. A total of 115 image slices were acquired from 1 patient. Axial view images were used. 10 images were randomly picked to serve as the test set. The rest of the images were used as the training set. We use this dataset to test if the cGAN network can perform segmentation tasks for bone since bone is the most important part to accurately annotate for the MRAC task, especially for the head and neck region. If bone can be accurately segmented in brain PET/MR images, we can later assign empirical attenuation values for brain PET attenuation correction as done in segmentation-based MRAC methods.

For the MR/CT and PET/MR - PET/CT datasets, we performed image registration between MR and CT images using image transforms in 3D Slicer and the image registration function in MATLAB. For the ZTE MR dataset, since the label images are the same as the input images, just with bone manually segmented, the input and label images are already perfectly aligned, therefore no extra registration is needed.

The images were converted from Hounsfield units (HU) to PNG images for the network input. Specifically, the range of [-600, 1400] HU values of the original CT images was linearly converted to the range of [0,255] pixel values of the converted PNG images, which is consistent with the pre-processing done in [34], [23]. The inputs of the cGAN network are the converted gray-scale PNG images with a range of [0,255] pixel values, and the outputs are also PNG images with a range of [0,255] pixel values. Then the outputs from the cGAN network are linearly re-scaled reversely from the range of [0,255] to [-600, 1400] in order to convert the images back to HU values.

For this manuscript, the main goal is to present the cGAN network structure we designed, and to show it can work effectively for different MRAC-related tasks. We believe even the limited datasets are more than adequate to support this aim.

C. Training Details

The input images were fed into the 2D network slice by slice during training. As done similarly in [23], to augment the number of training samples, each input image was first padded to a dimension of 286×286. Sub-images with dimensions of 256×256 were later randomly cropped to feed into the network. For different datasets, we trained the cGAN network for different numbers of iterations since the sizes of the datasets differ. For the co-registered MR/CT dataset and the pairing PET/MR - PET/CT dataset, we trained the network for 5k iterations with a batch size of 50. For the ZTE dataset, we trained the network for 500 iterations (also with a batch size of 50) to prevent over-fitting for this relatively small dataset. Adam optimization [35] was adopted for the training process with a momentum term of 0.5. A fixed learning rate of 0.0002 was used for the training process. For each training step, the discriminator was trained first and then the generator was trained. When calculating the discriminator and generator losses, we used an exponential moving average (EMA) with a decay coefficient of 0.99. The EMA assigns decaying weights on the losses acquired from earlier training steps. Its formula can be written as:

$$S_t = \begin{cases} L_1, & \text{if } t = 1 \\ \alpha \cdot L_t + (1-\alpha) \cdot S_{t-1}, & \text{if } t > 1 \end{cases} \tag{3}$$

where t represents the training iteration, L_t is the loss output from the network at iteration t, S_t is the EMA calculated loss at iteration t, and α is the decay coefficient, which is set as 0.99 in our training process.

D. Cross Validation

For the MR/CT and PET/MR - PET/CT datasets, we performed five-fold cross validation. For example, for the PET/MR - PET/CT dataset, we randomly split the images from 10 patients into five sets, with each set containing images from 2 different patients. Then we used one set for testing, and the rest of the images for training. We repeated this process five times until all the five sets have been used for testing. A similar procedure was performed for the MR/CT dataset, except that we have a total of 11 patients in this dataset and one was always used for training. Cross validation was not done for the ZTE MR dataset since the images were all acquired from the same patient.

E. Evaluation Metrics

To evaluate the quality of the network-generated CT images (pseudo CT images), we calculated the pixel-wise value difference (the absolute value) between the pseudo CT and label CT images. We further divided this value by 255 to acquire the normalized pixel value difference. For one pair of images, we averaged the pixel value difference across all pixels to represent the deviation of the pseudo CT image from the label CT image for this specific image pair. For an entire dataset, we further averaged the normalized pixel value difference across

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2020.2989073, IEEE Transactions on Radiation and Plasma Medical Sciences

all image pairs to represent the overall performance for this dataset.

As a similar metric, we also calculated the normalized root-mean-square error (NRMSE) to evaluate the similarity between the pseudo CT and label CT images. The NRMSE can be calculated as:

$$\mathrm{NRMSE} = \frac{\sqrt{\mathrm{MSE}}}{L} = \frac{1}{L}\sqrt{\frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[X(i,j)-Y(i,j)]^2}, \tag{4}$$

where $X(i,j)$ and $Y(i,j)$ are corresponding pixel values in two images, m and n represent the image dimensions, which is 256×256 in our case, and L is the dynamic range of the pixel values, which is 255 in our case.

In addition, we calculated the peak signal-to-noise ratio (PSNR) from the mean squared error (MSE). We calculated the MSE as:

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[X(i,j)-Y(i,j)]^2, \tag{5}$$

where $X(i,j)$ and $Y(i,j)$ are corresponding pixel values in the pseudo CT and label CT images, but converted to HU, m and n represent the image dimensions, which is 256×256 in our case. Then PSNR is calculated as:

$$\mathrm{PSNR} = 10 \cdot log_{10}(\frac{I_{max}^2}{\mathrm{MSE}}), \tag{6}$$

where MSE represents the mean squared error as defined above, and I_{max} is the maximum intensity value of the CT images in HU, which we set as 4095 in consistency with [22], [23]. For NRMSE and PSNR, we still averaged the values across all image pairs in a dataset to represent the performance for this dataset. Lower NRMSE and higher PSNR indicate better similarity between the pseudo CT and label CT images, i.e., better quality of the pseudo CT images generated from the cGAN network.

We also used the structural similarity (SSIM) index [36] to evaluate the similarity between the pseudo CT and label CT images. The SSIM index between two image windows x and y of common size $N \times N$ is calculated as:

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{({\mu_x}^2 + {\mu_y}^2 + c_1)({\sigma_x}^2 + {\sigma_y}^2 + c_2)}, \tag{7}$$

where μ_x is the average of x, μ_y is the average of y, ${\sigma_x}^2$ is the variance of x, ${\sigma_y}^2$ is the variance of y, σ_{xy} is the covariance of x and y, $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ are two variables to stabilize the division with weak denominator, with L being the dynamic range of the pixel values (255 in our case), and $k_1 = 0.01$ and $k_2 = 0.03$ by default.

The SSIM index has a value between -1 and 1, where 1 represents two identical images, and 0 indicates no structural similarity. The output image from the cGAN network has a dimension of 256×256. We used an image window of size 8×8 to calculate the SSIM index, and averaged across all image windows to acquire the SSIM index for one image. For an entire dataset, again we averaged the SSIM index across all images to represent the performance for this dataset. The image background is included for the SSIM index calculation.

For the ZTE MR dataset, since it is used to evaluate the cGAN network's performance for bone segmentation task, we used the Sørensen-Dice similarity coefficient (DSC) [37] as an additional quantitative evaluation metric. To calculate the DSC, we first converted both the pseudo CT and label CT images to binary images with 1 indicating bone and 0 otherwise. This can be done by making pixels with values above a set threshold (e.g., we chose to use 240 as the threshold) have a binary pixel value of 1, and other pixels 0 for both images. Then the DSC between the two binary images can be calculated as:

$$\mathrm{DSC}(X,Y) = \frac{2|X \cap Y|}{|X| + |Y|}, \tag{8}$$

where X and Y represent the converted binary images from the pseudo CT and label CT images, $|X|$ and $|Y|$ represent how many pixels these two images have in total respectively, and $|X \cap Y|$ represents how many common pixels they have (where pixels in corresponding positions both have a value of 1 or 0). DSC has a value between 0 and 1, where 1 represents two identical images, and 0 indicates no common pixels (no similarity). For the entire test set, the DSC was also averaged across all images.

III. Results

A. Co-registered MR and CT Images

In Fig. 4, we show the input MR image, cGAN output pseudo CT image, label CT image and the normalized pixel value difference map between the pseudo CT and label CT image from example slices of one patient in one of the test sets.

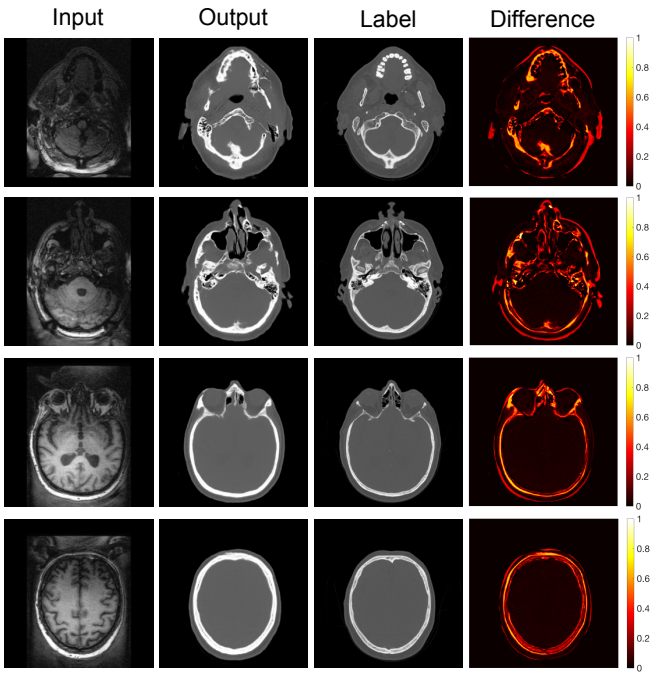

Fig. 4. Input MR image, cGAN output pseudo CT image, label CT image and the normalized difference map between the pseudo CT and label CT image from example slices of one patient in one of the test sets for the co-registered MR/CT dataset.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2020.2989073, IEEE Transactions on Radiation and Plasma Medical Sciences

The normalized pixel value difference, NRMSE, PSNR and the SSIM index for different cross validation sets are summarized in Table I. The averaged results and variances (across different cross validation sets) are: normalized pixel value difference of $4.54 \pm 0.13\%$, NRMSE of $10.50 \pm 1.42\%$, PSNR of 26.30 ± 1.08, and SSIM index of 0.76 ± 0.0007.

TABLE I
SUMMARY OF THE NORMALIZED PIXEL VALUE DIFFERENCE, NRMSE, PSNR AND SSIM INDEX FOR DIFFERENT CROSS VALIDATION SETS IN THE CO-REGISTERED MR/CT DATASET

	Set 1	Set 2	Set 3	Set 4	Set 5
Pixel value diff.	4.16%	5.04%	4.61%	4.67%	4.22%
NRMSE	8.62%	11.24%	11.31%	11.33%	10.00%
PSNR	27.97	25.71	25.50	25.67	26.67
SSIM index	0.80	0.77	0.77	0.75	0.73

B. Pairing PET/MR and PET/CT Images

In Fig. 5, we show the input MR image, cGAN output pseudo CT image, label CT image and the normalized pixel value difference map between the network-generated pseudo CT image and label CT image from example slices of one patient in one of the test sets.

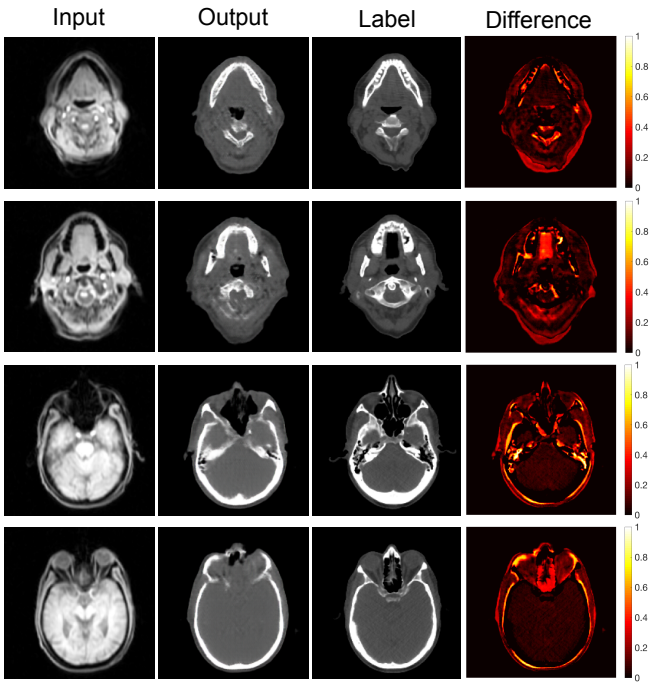

Fig. 5. Input MR image, cGAN output pseudo CT image, label CT image and the normalized pixel value difference map between the network-generated pseudo CT image and label CT image from example slices of one patient in one of the test sets for the pairing PET/MR - PET/CT dataset.

The normalized pixel value difference, NRMSE, PSNR and the SSIM index for different cross validation sets are summarized in Table II. The averaged results and variances (across different cross validation sets) are: normalized pixel value difference of $5.59 \pm 0.24\%$, NRMSE of $13.85 \pm 0.85\%$, PSNR of 23.58 ± 0.31, and SSIM index of 0.76 ± 0.0007.

TABLE II
SUMMARY OF THE NORMALIZED PIXEL VALUE DIFFERENCE, NRMSE, PSNR AND SSIM INDEX FOR DIFFERENT CROSS VALIDATION SETS IN THE PAIRING PET/MR - PET/CT DATASET

	Set 1	Set 2	Set 3	Set 4	Set 5
Pixel value diff.	5.77%	5.35%	4.94%	5.62%	6.27%
NRMSE	13.37%	13.23%	13.05%	15.22%	14.38%
PSNR	23.80	23.92	24.15	22.79	23.24
SSIM index	0.74	0.80	0.75	0.76	0.73

C. ZTE MR Images

In Fig. 6, we show the input MR image, cGAN output image with bone segmented, label image with bone segmented and the normalized pixel value difference map between the cGAN output and the label image in the test set.

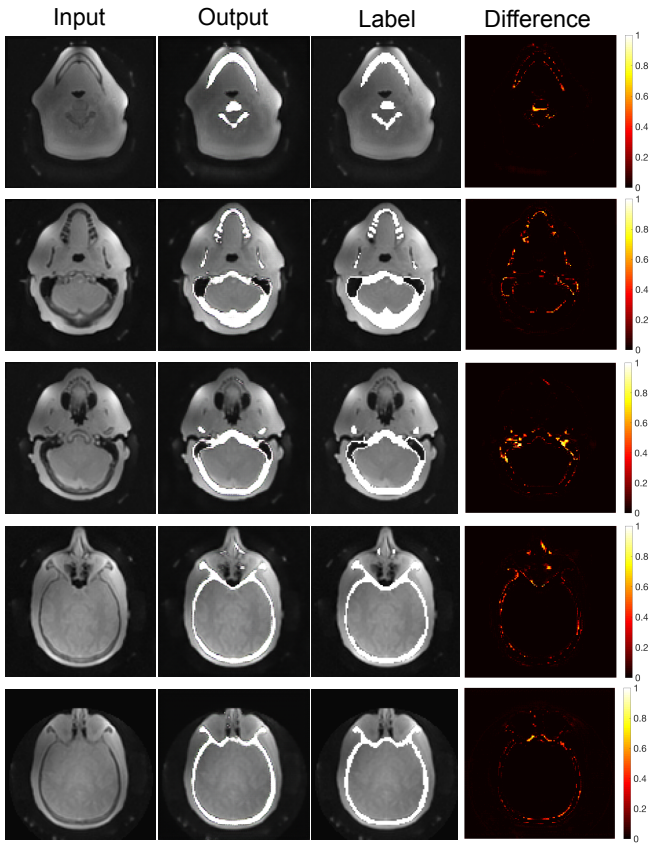

Fig. 6. Input MR image, cGAN output image with bone segmented, label image with bone segmented and the normalized pixel value difference map between the cGAN output and the label image in the test set for the ZTE MR dataset.

The quantitative metrics averaged across all images in the test set are: normalized pixel value difference of 2.48%, NRMSE of 7.07%, PSNR of 29.35, and SSIM index of 0.90.

In Fig. 7, we show examples of converting the pseudo CT and label CT images to binary images based on bone segmentation. Based on the segmented images, we calculated the average DSC for the pairing images in the test set as: 0.83.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2020.2989073, IEEE Transactions on Radiation and Plasma Medical Sciences

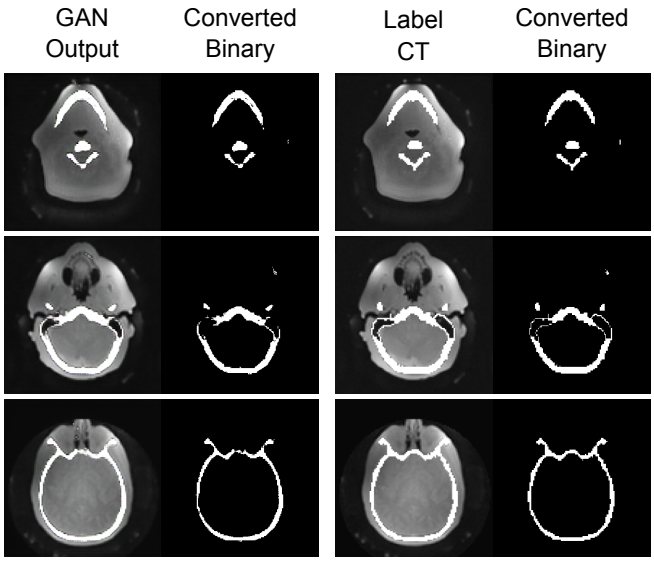

Fig. 7. Example slices of converting the pseudo CT and label CT images to binary images based on bone segmentation.

D. Comparing the Conditional GAN Network with Other Network Structures

We have compared the performance of the cGAN network discussed in this manuscript with that of other network structures we studied before. First of all, we compared the performance of the cGAN network with that of the simple Autoencoder used in our previous work [26]. With the PET/MR - PET/CT dataset, the simple Autoencoder generates a normalized pixel value difference of around 15% and a SSIM index of around 0.5. We have also tried to use the original *pix2pix* network [25] without adding residual blocks for the MRAC task [38]. For the co-registered MR/CT dataset, the *pix2pix* network generates an average normalized pixel value difference of 7.62%, NRMSE of 17.46%, PSNR of 21.98, and SSIM index of 0.67.

We also compared our results with the results acquired by other researchers. We still take the co-registered MR/CT dataset as an example. We have achieved a normalized pixel value difference of 4.54%. If we convert this to the mean absolute error (MAE) in HU, it corresponds to around 90 HU. This value is comparable to the MAE range of 75 - 100 HU reported in [22], [23]. Looking at the PSNR and SSIM index, we have achieved a PSNR of 26.30 and SSIM index of 0.76. The PSNR is comparable to the PSNR of 27 reported in [22] but slightly lower than the PSNR of 30 reported in [23]. The SSIM index is slightly lower than the SSIM index of 0.85 reported in [28]. If we look at the DSC of 0.83 acquired from the ZTE MR dataset, we can see it is comparable to the DSC of 0.80 reported in [28].

IV. Discussion

From Fig. 4, we can see that the cGAN network can be trained to convert T1 MR images to CT images, and then the pseudo CT images can be further used to generate photon attenuation maps for MRAC. From Fig. 5, we can see that the network can also perform the conversion task for Dixon MR images acquired with a PET/MR scanner. From Table I and II, we can see that the network generalizes very well when using different cross validation sets.

From Fig. 6, we can see that the cGAN network can also successfully segment bone from ZTE MR images, which is essential for the MRAC task. We should note that due to the limited resource of ZTE MR images, we only have ZTE MR images from one patient with manually segmented label images, and therefore performed training and testing on images of the same patient for the ZTE MR dataset. This was done to show another possible application of the proposed cGAN network structure for MRAC-related tasks, i.e., bone segmentation in MR images. It will also be useful for readers interested in a different approach for achieving and evaluating accuracy of bone segmentation. A more ideal case would be to verify the network's function by training and testing on different patients, which will be done in our future work. We will also study the segmentation of other tissues in the future.

Comparing the results shown in Table I and II, as well as the results discussed in Sec. III-C, we can see that the cGAN network performs differently (judged by the quantitative metrics) for different datasets/tasks. This is related to the difference in the patient data for each dataset, the difference in the statistics of the input images due to different imaging sequences and scanners, the difference in the nature of the task (e.g., image translation vs. segmentation), as well as the difference in the nature of the training and test sets (e.g., if the training and testing were done on the same patient). The performance of the network can also be affected by the registration quality between the input and label images for each dataset, as well as the quality (e.g., contrast) of the input images.

Based on the results presented in Sec. III-D, we can see that the cGAN network discussed in this manuscript has shown greatly improved performance compared to the simple Autoencoder used in our previous work [26]. It also outperforms the original *pix2pix* network [25] without adding residual blocks, since the residual blocks further facilitate a smoother information flow within each encoder/decoder layer. In addition, it shows comparable performance to the deep learning networks studied by other researchers for MRAC task. As we have discussed above, we should again note that the difference in the quantitative metrics is affected by the difference in the datasets, the image registration quality, and the quality of the input images. For our future work, we will acquire better quality patient data (e.g., MR images with better contrast) with improved registration and fine tune the network structure and training hyper-parameters to improve the results.

In this work, we chose to use a cGAN network with residual blocks since some of its key features have enabled its improved performance over a simple Autoencoder or U-net network. For example, our cGAN network avoids the problem of having a very small dimensional inner-most latent layer as in a simple encoder - decoder network by adopting the U-net structure. This feature works to preserve a smooth information flow between the input and output images. Adding in extra residual blocks further helps with the information flow within each encoder/decoder layer. In addition, our cGAN network uses

an objective function that combines the L1 loss and the GAN loss. This GAN structure and loss function treat the entire input image as a whole instead of performing pixel-wise image conversion. This feature enables the cGAN network to be less sensitive to the misalignment between the input and label images, as well as mitigates the artificial smoothing of the output images as when using a simple L1 or L2 loss. At the same time, we adopted a PatchGAN loss [25] instead of a single value for the GAN loss (i.e., the final output from the discriminator is 30×30 instead of a single value). This allows the cGAN network to focus better on the high frequency information in the images.

V. Conclusion and Future Work

In this work, we have designed and trained a cGAN network with residual blocks for MRAC task. We studied the performance of the network on image translation and segmentation tasks, which are essential for MRAC. The network is proved to successfully convert MR images to CT images, as well as segment bone from MR images.

For future work, we will further fine tune the network architecture and adjust the training hyper-parameters (e.g., learning rate, number of training iterations) to improve its performance. We will also acquire images from more patient subjects for our study. Especially for the ZTE MR dataset, we will acquire images from other patients in order to train and test the network on different patients. We will work to achieve better registration on these images. We also plan to perform data augmentation to further augment the datasets.

Instead of only focusing on the brain, we will study the performance of the network on other parts of the body. In addition, we will upgrade the 2D training to 2.5D training, and develop a 3D network (which makes better use of the spatial information/relation between consecutive slices) based on the 2D network used in this manuscript. Finally, we will convert the network output pseudo CT images to photon attenuation maps and test their performance for attenuation correction during PET image reconstruction.

Acknowledgment

The authors would like to thank Sam Xu for the hard work on this project, and Jonathan Fisher and Dr. Garry Chinn for helpful discussions. The co-registered MR and CT data are provided by the Focused Ultrasound Foundation and the University of Virginia, Department of Neurosurgery. We also thank Dr. Andrei H. Iagaru, Dawn Holley and Kimberly Ramos for providing the PET/MR and PET/CT scans. In addition, we thank Dr. Keum Sil Lee and Jim Andrew Best-Devereux for helping with the bone segmentation for the ZTE MR images. This material is based upon work supported by the National Science Foundation under Grant No. 1828993.

References

[1] D. A. Torigian, H. Zaidi, T. C. Kwee, B. Saboury, J. K. Udupa, Z.-H. Cho, and A. Alavi, "Pet/mr imaging: technical aspects and potential clinical applications," *Radiology*, vol. 267, no. 1, pp. 26–44, 2013.

[2] C. E. Mader, T. Fuchs, D. A. Ferraro, and I. A. Burger, "Potential clinical applications of pet/mr." *IEEE Transactions on Radiation and Plasma Medical Sciences*, 2020.

[3] C. N. Ladefoged, I. Law, U. Anazodo, K. S. Lawrence, D. Izquierdo-Garcia, C. Catana, N. Burgos, M. J. Cardoso, S. Ourselin, B. Hutton *et al.*, "A multi-centre evaluation of eleven clinically feasible brain pet/mri attenuation correction techniques using a large cohort of patients," *Neuroimage*, vol. 147, pp. 346–359, 2017.

[4] M. Cencini, M. Tosetti, and G. Buonincontri, "An aristotelian view on mr-based attenuation correction (aristomrac): combining the four elements," *IEEE Transactions on Radiation and Plasma Medical Sciences*, vol. 3, no. 4, pp. 491–497, 2019.

[5] Y. Berker, J. Franke, A. Salomon, M. Palmowski, H. C. Donker, Y. Temur, F. M. Mottaghy, C. Kuhl, D. Izquierdo-Garcia, Z. A. Fayad *et al.*, "Mri-based attenuation correction for hybrid pet/mri systems: a 4-class tissue segmentation technique using a combined ultrashort-echo-time/dixon mri sequence," *Journal of nuclear medicine*, vol. 53, no. 5, pp. 796–804, 2012.

[6] J. Yang, F. Wiesinger, S. Kaushik, D. Shanbhag, T. A. Hope, P. E. Larson, and Y. Seo, "Evaluation of sinus/edge-corrected zero-echo-time–based attenuation correction in brain pet/mri," *Journal of Nuclear Medicine*, vol. 58, no. 11, pp. 1873–1879, 2017.

[7] S. Roy, W.-T. Wang, A. Carass, J. L. Prince, J. A. Butman, and D. L. Pham, "Pet attenuation correction using synthetic ct from ultrashort echo-time mr imaging," *Journal of Nuclear Medicine*, vol. 55, no. 12, pp. 2071–2077, 2014.

[8] K.-H. Su, H. T. Friel, J.-W. Kuo, R. Al Helo, A. Baydoun, C. Stehning, A. N. Crisan, M. S. Traughber, A. Devaraj, D. W. Jordan *et al.*, "Ute-mdixon-based thorax synthetic ct generation," *Medical physics*, vol. 46, no. 8, pp. 3520–3531, 2019.

[9] G. Delso, B. Fernandez, F. Wiesinger, Y. Jian, C. Bobb, and F. Jansen, "Repeatability of zte bone maps of the head," *IEEE Transactions on Radiation and Plasma Medical Sciences*, vol. 2, no. 3, pp. 244–249, 2017.

[10] G. Delso, D. Gillett, W. Bashari, T. Matys, I. Mendichovszky, and M. Gurnell, "Clinical evaluation of 11 c-met-avid pituitary lesions using a zte-based ac method," *IEEE Transactions on Radiation and Plasma Medical Sciences*, vol. 3, no. 4, pp. 504–508, 2018.

[11] S. Wollenweber, S. Ambwani, G. Delso, A. Lonn, R. Mullick, F. Wiesinger, Z. Piti, A. Tari, G. Novak, and M. Fidrich, "Evaluation of an atlas-based pet head attenuation correction using pet/ct & mr patient data," *IEEE Transactions on Nuclear Science*, vol. 60, no. 5, pp. 3383–3390, 2013.

[12] D. Shen, G. Wu, and H.-I. Suk, "Deep learning in medical image analysis," *Annual review of biomedical engineering*, vol. 19, pp. 221–248, 2017.

[13] F. Liu, H. Jang, R. Kijowski, T. Bradshaw, and A. B. McMillan, "Deep learning mr imaging–based attenuation correction for pet/mr imaging," *Radiology*, vol. 286, no. 2, pp. 676–684, 2018.

[14] F. Liu, H. Jang, R. Kijowski, G. Zhao, T. Bradshaw, and A. B. McMillan, "A deep learning approach for 18 f-fdg pet attenuation correction," *EJNMMI physics*, vol. 5, no. 1, pp. 1–15, 2018.

[15] J. Yang, D. Park, G. T. Gullberg, and Y. Seo, "Joint correction of attenuation and scatter in image space using deep convolutional neural networks for dedicated brain 18f-fdg pet," *Physics in Medicine & Biology*, vol. 64, no. 7, p. 075019, 2019.

[16] K. Gong, J. Yang, K. Kim, G. El Fakhri, Y. Seo, and Q. Li, "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, p. 125011, 2018.

[17] T. J. Bradshaw, G. Zhao, H. Jang, F. Liu, and A. B. McMillan, "Feasibility of deep learning–based pet/mr attenuation correction in the pelvis using only diagnostic mr images," *Tomography*, vol. 4, no. 3, p. 138, 2018.

[18] A. P. Leynes, J. Yang, F. Wiesinger, S. S. Kaushik, D. D. Shanbhag, Y. Seo, T. A. Hope, and P. E. Larson, "Zero-echo-time and dixon deep pseudo-ct (zedd ct): direct generation of pseudo-ct images for pelvic pet/mri attenuation correction using deep convolutional neural networks with multiparametric mri," *Journal of Nuclear Medicine*, vol. 59, no. 5, pp. 852–858, 2018.

[19] D. Hwang, S. K. Kang, K. Y. Kim, S. Seo, J. C. Paeng, D. S. Lee, and J. S. Lee, "Generation of pet attenuation map for whole-body time-of-flight 18f-fdg pet/mri using a deep neural network trained with simultaneously reconstructed activity and attenuation maps," *Journal of Nuclear Medicine*, pp. jnumed–118, 2019.

[20] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in

Advances in neural information processing systems, 2014, pp. 2672–2680.

[21] Y. Wang, B. Yu, L. Wang, C. Zu, D. S. Lalush, W. Lin, X. Wu, J. Zhou, D. Shen, and L. Zhou, "3d conditional generative adversarial networks for high-quality pet image estimation at low dose," *NeuroImage*, vol. 174, pp. 550–562, 2018.

[22] D. Nie, R. Trullo, J. Lian, C. Petitjean, S. Ruan, Q. Wang, and D. Shen, "Medical image synthesis with context-aware generative adversarial networks," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2017, pp. 417–425.

[23] J. M. Wolterink, A. M. Dinkla, M. H. Savenije, P. R. Seevinck, C. A. van den Berg, and I. Išgum, "Deep mr to ct synthesis using unpaired data," in *International workshop on simulation and synthesis in medical imaging*. Springer, 2017, pp. 14–23.

[24] H. Emami, M. Dong, S. P. Nejad-Davarani, and C. K. Glide-Hurst, "Generating synthetic cts from magnetic resonance images using generative adversarial networks," *Medical physics*, vol. 45, no. 8, pp. 3627–3636, 2018.

[25] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[26] K. S. Lee, L. Tao, J. Best-Devereux, and C. S. Levin, "Study of a convolutional autoencoder for automatic generation of mr-based attenuation map in pet/mr," in *2017 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*. IEEE, 2017, pp. 1–3.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[28] H. Arabi, G. Zeng, G. Zheng, and H. Zaidi, "Novel adversarial semantic structure deep learning for mri-guided attenuation correction in brain pet/mri," *European journal of nuclear medicine and molecular imaging*, vol. 46, no. 13, pp. 2746–2759, 2019.

[29] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[30] T. M. Quan, D. G. Hildebrand, and W.-K. Jeong, "Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics," *arXiv preprint arXiv:1612.05360*, 2016.

[31] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 2016, pp. 265–283.

[32] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[33] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[34] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[36] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2. Ieee, 2003, pp. 1398–1402.

[37] D. A. Jackson, K. M. Somers, and H. H. Harvey, "Similarity coefficients: measures of co-occurrence and association or simply measures of occurrence?" *The American Naturalist*, vol. 133, no. 3, pp. 436–453, 1989.

[38] L. Tao, X. Li, J. Fisher, and C. S. Levin, "Application of conditional adversarial networks for automatic generation of mr-based attenuation map in pet/mr," in *2018 IEEE Nuclear Science Symposium and Medical Imaging Conference Proceedings (NSS/MIC)*. IEEE, 2018, pp. 1–3.