Unpaired PET/CT image synthesis of liver region using CycleGAN

Gianmarco Santini[a], Constance Fourcade[a], Noémie Moreau[a], Caroline Rousseau[b,c], Ludovic Ferrer[b,c], Marie Lacombe[c], Vincent Fleury[c], Mario Campone[b,c], Pascal Jézéquel[d], and Mathieu Rubeaux[a]

[a]Keosys Medical Imaging, Nantes, France
[b]University of Nantes, CRCINA, INSERM UMR1232, CNRS-ERL6001, Nantes, France
[c]ICO Cancer Center, Nantes - Angers, France
[d]Bioinfomic unit, ICO Integrated Center for Oncology, Bd J. Monod, 44805 Nantes-Saint Herblain, France

ABSTRACT

Cross-modality synthesis represent nowadays a promising application in medical image processing to manage the problem of paired data scarcity. In this work we designed and trained a CycleGAN model to generate PET/CT data from 2D slices collected from the liver body region of twelve patients. The results obtained from the six test patients show how our model can outperform baseline CycleGAN framework and effectively be used for synthesizing artificial images to be used for data augmentation or dataset completion.

Keywords: GAN, Cross-modality-synthesis, PET/CT, liver

1. INTRODUCTION

Medical imaging encompasses different imaging modalities and processes to image the human body. Each modality (e.g. Magnetic Resonance Imaging (MRI), Computed Tomography (CT), Positron Emission Tomography (PET) etc.) has unique specificities and parameters which aids the physician to diagnose and treat patients.
When combining different modalities as done with PET/CT, clinical decision making is shown to improve as multimodal imaging provides more information than single modality imaging.[1] Many deep learning algorithms have therefore been proposed to solve segmentation[2–4] or registration task[5] combining multimodal information, and resulted to perform better than methods using a single modality.[4]
However, collecting sufficient high quality paired data from different modalities in order to train a deep learning algorithm can be challenging. In cases of data scarcity, the problem can be circumvented by applying data augmentation strategies, but when data is incomplete or series are missing, the problem is not be easily tackled. Generative adversarial networks (GANs)[6] comprises a series of unsupervised deep learning models which have archieved great results in the computer vision and medical image processing area thanks to their capability of extracting the data underlying distribution and their internal structure.[7] They represent nowadays one of the most promising techniques to tackle the problem of data generation and augmentation. The underlying idea in GAN frameworks is to use two distinct convolutional neural networks (CNNs) trained in a zero-sum game: the first CNN aims at generating images to be as similar as possible to the target ones, while the second one aims at distinguishing between the real and the generated data.
In medical imaging, GAN based methods are generally used for applications such as synthesis of new data for data augmentation purposes,[8] denoising,[9,10] motion correction[11,12] or image reconstruction operations.[13] Cross modality synthesis is one of the most studied and explored application in the context of data generation.[14] It consists in the creation of an image from a certain modality starting from another one, thus allowing to create artificial images to exploit the information of both modalities.

Send correspondence to:.
E-mail: gianmarco.santini@keosys.com

In most of the literature works the image-to-image translation application is carried out for the CT/MRI domains.[8,15–17] This is due to the fact that there is more MRI data available compared to CT scans, but also because MRI, unlike CT, does not use ionizing radiation and is therefore less harmful to patients. MRI data can also be created from CT with the aim to reduce the acquisition time required by certain sequences and protocols. GAN methods used to transition from CT to PET or vice versa are however less common, and few examples are reported in the literature. For instance Bi et al.[18] proposed a GAN solution to generate PET images of the chest region using a multi-channel input composed of the CT data and the corresponding lesion masks. Ben-Cohen et al.[19] placed in series a fully convolutional network and a cGAN to generate PET images and improve the lesion detection on liver. Finally Armanius et al.[20] developed a framework called MedGAN trained with paired data to perform PET denoising and correct motion artifacts.
For cross domain data synthesis, two GAN-based strategies received extensive coverage: the pix-to-pix based solutions[21] used when data are generally paired (i.e. a co-registration between cases is feasible and a good spatial correspondence between structures can be achieved) and the CycleGAN models[22] used when data are unpaired. In this paper we present a modified CycleGAN algorithm to perform an image-to-image translation on the liver region from PET/CT data trained in an unpaired way.

2. METHODS

2.1 Data

Our dataset is composed by of 18 patients treated for metastatic breast cancer, who underwent both CT and PET acquisitions. The CT full body volumes are acquired with an average tube voltage equals to 100 kVp, a tube current of 221 mA and come with spacing of $0.9\times0.9\times2\ mm^3$, resulting in 512×512 2D axial slices. The PET images are produced through the injection of 18-FDG radioactive tracer and are acquired with a resolution of $4.073\times4.073\times2.027\ mm^3$, bringing to a 200×200 pixel in plane field of view. The PET intensity values are then normalized using the standardized uptake value (SUV) expression:[23]

$$SUV = \frac{r}{a' \cdot w}\,, \tag{1}$$

where r is the radioactivity concentration measured by the PET scan $[kBq/ml]$, a' represents the decay-corrected amount of injected radiolabeled FDG $[kBq]$ and w is the patient weight $[g]$.

2.2 Preprocessing

To allow a direct comparison between the original image and the generated one and to help the synthesis operations a preprocessing step is performed. Both CT and PET images are processed in order to archive a dimension of 256×256 pixels in the axial plane: the CT scans are down-sampled by a factor of two, while the PET volumes are rigidly registered and resampled to the corresponding down-sampled CT acquisitions.
Then a pre-trained model we developed[24] for liver segmentation is used to limit the region of interest along the axial direction and to keep only the slices where the liver is present in both modalities.
As shown in the literature, the presence of constraints on the image values can help the model's generation process by limiting the window of values.[19] For this reason, we first clipped the value range and then normalized them between zero and one. In particular, for CT images we limited the values between -160 HU and 240 HU, while for PET we fixed the SUV range between 0 and 20.
Finally, in order to simplify the process of cross modality transfer we removed the table that is visible on the CT images with a combination of morphological and thresholding filters.

2.3 CycleGAN model

The CycleGAN represents an unsupervised technique to perform image-to-image translation across two different image domains by using collections of unpaired data (see Figure 1). Considering X and Y as two different domains (e.g. CT and PET), the model aims to find two transfer functions that allow to pass from a space to another according to $G_Y : X \rightarrow Y$ and $G_X : Y \rightarrow X$. The transfer functions G_X and G_Y are produced using two CNNs, adversarially trained with two discriminator networks (D_X, D_Y) which respectively assess whether

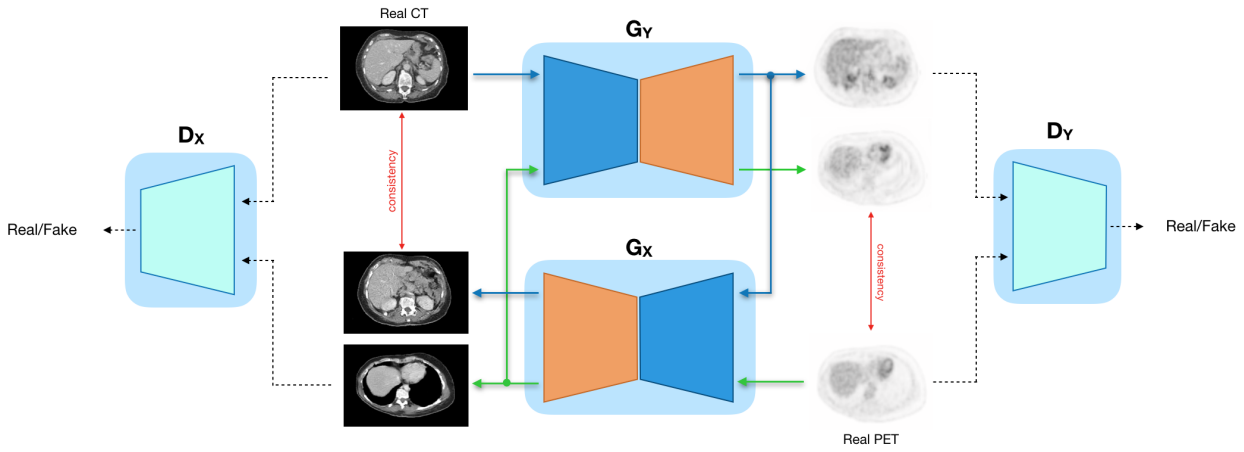

Figure 1. CycleGAN schematic model using unpaired combinations of input data.

the passed images are real or artificially generated.

To guarantee training stability during the learning process, we used the Least Square GAN version (LSGAN)[25] for our generative models. It has the advantage of penalizing the generated samples according to the distance from the decision boundary.
The adversarial contributes in the loss function can so be written as:

$$L_{adv_X}(G_X, D_X) = (1 - D_X(x_i))^2 + D_X(G_X(y_i))^2 \tag{2}$$

$$L_{adv_X}(G_Y, D_Y) = (1 - D_Y(y_i))^2 + D_Y(G_Y(x_i))^2 \tag{3}$$

The second contribute that is generally added in a CycleGAN model to make the overall training procedure more stable and to avoid generators producing inconsistent synthesized images is the cycle-consistency term. It represents a constraint introduced to minimize the reconstruction error between an image and the result of passing it through both the generators (blue and green cycle paths in Figure 1) as expressed in Eq. 4:

$$L_{cc}(G_X, G_Y, D_X, D_Y) = \|x_i + G_X(G_Y(x_i))\| + \|y_i - G_Y(G_X(y_i))\| \tag{4}$$

where x_i and y_i are two random real images sampled from the respective domains. The final loss function for a baseline CycleGAN results to be the sum of the adversarial contributions and the cycle consistency term. A schematic representation of a CycleGAN model is given in Figure 1.

IIn addition to the baseline CycleGAN loss function, we included two other contributes: 1) a consistency regularization (CR)[26] on discriminators and 2) a feature matching loss as a further constraint for the generators. The CR is produced applying a random transformation on the real data and passing it as a third input to the discriminator with the aim to make D_X and D_Y still capable of distinguish the real cases from the fake ones, even if perturbed.
It's given by:

$$L_{CR}(D_X, D_Y) = \sum_j \lambda_{CR_j} \|D_{X_j}(x) - D_{X_j}(T(x))\|^2 + \sum_j \lambda_{CR_j} \|D_{Y_j}(y) - D_{Y_j}(T(y))\|^2 \tag{5}$$

where j represents the generic layer of D used for the regularization and $T(\cdot)$ is a data augmentation function applied on the input.

The feature matching loss is applied on the generators to encourage them to produce more realistic images trying to fool the discriminator to recognize the salient features.[27,28] It is calculated minimizing the difference between the discriminator feature maps produced by real images and the ones derived from feeding the same network with generated samples as reported below.

$$L_{FM}(G_X, G_Y, D_X, D_Y) = \frac{1}{N-1}\sum_{j=1}^{N-1} \|D_{X_j}(x) - D_{X_j}(G_X(y))\|^2 + \frac{1}{N-1}\sum_{j=1}^{N-1} \|D_{Y_j}(y) - D_{Y_j}(G_X(x))\|^2 \quad (6)$$

The overall cost function can be finally defined as:

$$\begin{aligned} L(G_X, G_Y, D_X, D_Y) =& L_{adv_X}(G_X, D_X) + L_{adv_Y}(G_Y, D_Y) \\ &+ \lambda_{CC} L_{CC}(G_X, G_Y) + L_{CR}(D_X, D_Y \\ &+ \lambda_{FM} L_{FM}(G_X, G_Y, D_X, D_Y) \end{aligned} \quad (7)$$

To keep the formulation as general as possible we used the generic notations as X and Y.

2.4 Network architectures

To perform the domain transfer operation, we took into account two different solutions for the generator networks. The first one is a ResNet[22,29] composed of nine residual blocks. Each block is structured with two 3×3 convolutional layers followed by in-stance normalization and ReLU activation function. The blocks are also placed in the middle of a small contracting path and a symmetric up-sampling one.
The second solution employed is a UNet architecture to synthesize the fake images. It mainly consists of eight encoding levels, followed by an equal number of up-sampling operations in the decoding part for image reconstruction. Instance normalization is used through all the network, while the two branches are characterized by different activation functions: a LeakyReLU is used in the encoding part while a simple ReLU is employed during decoding. Dropout layer is finally added in the bridge part of the network and after every up-sampling layer to reduce the number of parameters that tend to significantly increase as a consequence of the concatenation between the low level and high level feature maps.
The discriminators employed are similar to the ones used in the original PatchGAN paper citezhu2017unpaired. Instead of returning a single value to assess whether the entire input image is real or not, the PatchGAN discriminator output produces a tensor where each pixel represents the classification of a single overlap-ping patch of shape 70×70 in the input image. This makes the networks more prone to use high frequency features to distinguish the input data. Finally, instance normalization was replaced by spectral normalization,[30] which is expected to give the best results in combination with the consistency regularization.[26]
A detailed description of all the network architectures with all the parameters is given in Table 2 and 3 in the Appendix A.

2.5 Evaluation metrics

To assess the quality of the synthesized images we used some well-established quantitative metrics.[15,19]
First, we computed the Mean Absolute Error (MAE) between the original image and the generated one:

$$MAE = \frac{1}{N}\sum_{i}^{N} |x_i - G_X(y_i)| \quad (8)$$

The second score used is the Peak Signal to Noise Ratio (PSNR), which is calculated as:

$$PSNR = 20 \cdot \log\left(\frac{MAX_I}{RMSE}\right) \quad (9)$$

where the MAX_I represent the maximum intensity value of the image and $RMSE$ is root mean square error between the real and the generated images. For the PET images the MAX_I value is set to 20, while for the CT

images we shifted the values in the range [0−400] HU and we used the new upper bound as maximum value.
In all cases we computed the above statistics only on the foreground portion of image, obtained by using a mask to isolate it. In the case of the CT images we removed all the background pixels having value equals to -1000 HU. For each PET image we extracted the foreground portion from the union of the intensity pixels greater than zero and the mask derived for the corresponding CT volume, since a registered step was used as preprocessing. Such operation was done to fill some holes that can be generated in PET foreground mask by simply taking the values greater than zero.

3. EXPERIMENTAL SETTINGS AND RESULTS

From the 18 patients we collected, we split our dataset in 12 training cases and left the remaining six for the test purpose.

3.1 Training settings

The presented model was implemented using Tensorflow framework and trained from scratch on a single NVIDIA GTX 1080 TI using only the axial slices from both modality volumes. We also used Adam optimizer with a first order momentum β of 0.5
Due to memory constraints we conducted our experiments training the models for 200 epochs with a batch size of only one sample per iteration (for both generators and discriminators) for each image domain. The initial learning rate was set at 0.002 and kept fixed till epoch 100, after which it was linearly decreased to zero according to the expression: $\mu_{ep} = \mu_{ep_{-1}}(E - ep)/(E - E_s)$, where E is total number of epochs, ep represents the current epoch and E_s is the 100^{th} epoch step.
With the configuration settings described above we performed a series of experiments in order to identify the best values to weight the different contributes which compose the model loss function defined in Eq.7. For the cycle consistency contributes λ_{CC} we kept the standard value of 10, while for the feature matching λ_{FM} we started form 10 until we reached an acceptable value of 1. Higher values seemed to encourage the generator to modify the shape of the synthesized image, especially when trying to generate a CT image from a PET acquisition.
The consistency regularization term added in the discriminator loss was obtained exploiting only the information coming from the last layer and as consequence weighted with a single λ_{CR} value equals to 10. We observed that smaller values did not produced significant differences in terms of image consistency compared to a state-of-the-art version of the CycleGAN i.e. without the CR. Moreover, the employment of such regularization allows to make the prediction function smoother. As stated in the original paper[26] the applied transformations to augment the real data ($T(\cdot)$ of Eq.5) have to preserve the input semantic information. For this reason, the discriminators were fed with samples coming through a simple random translation function which is produced by a first enlargement to 286×286 pixels with a padding operation and followed by a random crop to the original dimensions.

Table 1. The simulation results for different network configurations with and without the penalty and regularization terms (PR), trained with paired (P) or unpaired (U) data. For each modality we reported the mean absolute error (MAE) and the peak signal to noise ration (PSNR).

Model	**MAE_{PET}**	**MAE_{CT}**	**$PSNR_{PET}$**	**$PSNR_{CT}$**
UNet	0.43 ± 0.03	92.82 ± 8,95	29.03 ± 2.70	11.52 ± 0.78
ResNet	0.42 ± 0.03	64.48 ± 4.17	30.80 ± 2.58	13.06 ± 0.57
UNet+PR+U	0.41 ± 0.05	62.45 ± 3.69	31.36 ± 2.45	13.20 ± 0.48
UNet+PR+P	0.42 ± 0.04	62.57 ± 3.90	31.20 ± 2.28	13.09 ± 0.52
ResNet+PR+U	**0.40 ± 0.04**	**61.75 ± 3.62**	**31.50 ± 2.63**	**13.27 ± 0.50**
ResNet+PR+P	0.40 ± 0.05	61.87 ± 3.98	31.34 ± 2.55	13.20 ± 0.54

3.2 Results

A series of experiments were carried out evaluating how the addition of penalty and regularization terms (PR), described in section 2.3, influenced the results compared to the baseline models considering two different types of architecture. Furthermore, exploiting the advantage of having paired data, in the case of the modified models, we evaluated the algorithm performance when trained with paired (P) and unpaired (U) combination of data. The results obtained from all the simulations are summarized in Table 1.

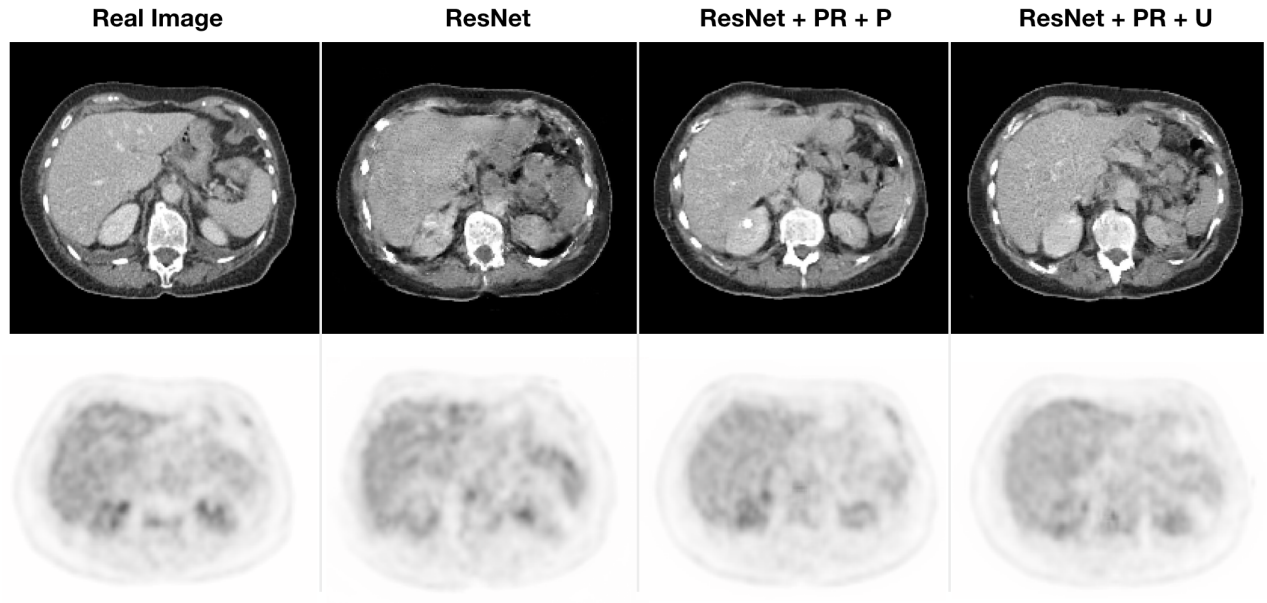

Figure 2. Comparison between the real images and the artificial ones, created with a baseline model, and a model with penalty and regularization terms (PR) trained with paired (P) or un paired (U) data. For simplicity we reported the images produced using only the ResNet architecture since it performed better than the UNet.

4. DISCUSSION AND CONCLUSION

In this preliminary work we presented a method based on CycleGAN to perform cross modality synthesis of PET/CT data focusing our analysis only on the liver region.

The results obtained show that adding of constraints term on the generator and the regularization on discriminators improves the model's performance to synthesize compared to the baseline models. Using consistency regularization in conjunction with spectral normalization on both discriminators further stabilizes the training process, avoiding a rapid fall of the discriminator loss to zero. It also forces the generators to create more realistic images, with sharper borders, especially when creating CT images. With baseline models the presence of grid-like artifact were constantly present over the generated CT slices.

However, compared to the PET generated images, the synthesized CT data show significant lower values of PSNR. This is certainly due to the fact that the generation of CTs uses PET data which naturally present less information.

Regarding the two architectures employed, the ResNet seems to slightly outperform the model designed with a UNet for the generators. Moreover just as stated in the literature[15] the model trained with unpaired data slightly surpass the performance obtained when training everything with the paired data. This is probably a consequence of some small misalignments that can occur despite registration is performed. This anyway confirms the feasibility of using unpaired data for this task.

Some improvements still need to be incorporated. In particular we didn't specify any terms to take into account high and low intensity SUV values as done by[19] in their loss function. As a consequence of that, in the few cases of the dataset presenting liver lesions, these were not recognized and synthesized (see Figure 3, Fake PET). It is

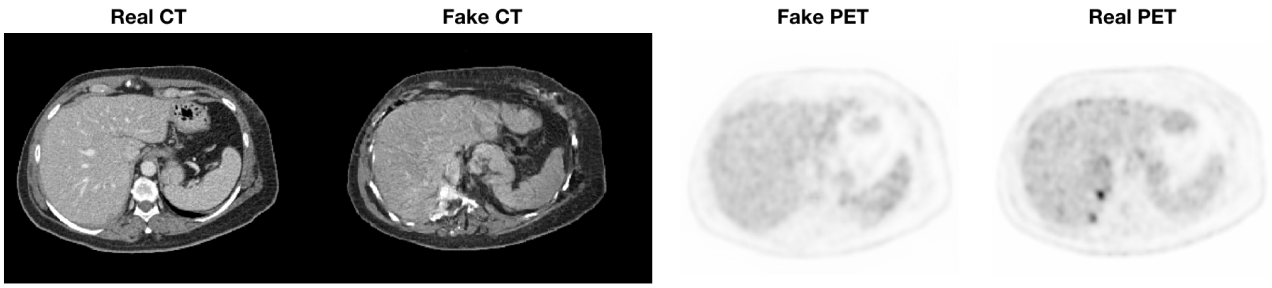

Figure 3. Example of missing lesion generation on the PET liver image and bone artifact generation on the corresponding CT image.

also true that their identification is not at all immediate from CT images.
Additionally, the data presented many metastatic lesions visible on the PET images and localized on the patients' backbone. We observed that without any constraints, the generator of CT images creates fake bones or parts of them in the corresponding regions of the PET images where high intensity values are present (see Figure 3 fake CT part). In future work we will address this problem by adding specific penalty terms to limit such effect. Furthermore, the use of a 3D approach will be explored, considering that in some cases there can be small inconsistencies across the axial slices, especially when generating CT images.
Nonetheless, compared to Ben-Cohen et al.[19] or Bi et al.[18] who used paired samples, we have presented in this paper a method capable of performing image-to-image translation of the abdominal region using unpaired PET/CT data configuration in a fully unsupervised way. This is the most common situation both in clinical practice and in the field of research when looking for publicly available datasets.
Even though we chose the liver as target for this application, the method can be easily applied to other body areas such as the head-neck or the thoracic region to create new data for data augmentation purposes or for further processing operations

APPENDIX A. ARCHITETURE DETAILS

Table 2. ResNet architecture. In the parameter column are reported the kernel size, the stride s, the activation function and the padding type (If not specified padding type is *same*). The table is also organized in a way that the full structure is reported on the left while the details about the res blocks on the right part of table.

Network			**Res Block**		
Layer name	**Parameters**	**Feature size**	**Layer name**	**Parameters**	**Feature size**
Input		256×256×1			
enc1	7×7×64; s=1 relu	256×256×64	conv1	3×3×512; s=1 relu, pad:reflect	64×64×512
enc2	3×3×128; s=2 relu	128×128×128	conv2	3×3×512; s=1 pad:reflect	64×64×512
enc3	3×3×256; s=2 relu	64×64×256	conv1+conv2		64×64×512
res blocks×9	3×3×512; s=1 relu	64×64×512			
dec3	3×3×512; s=2 relu	128×128×256			
dec2	3×3×512; s=2 relu	256×256×64			
Output	7×7×1; s=1 tanh pad:reflect	256×256×1			

Table 3. UNet architecture. In the parameter column are reporterd the kernel size, the stride s, the activation function and the application of dropout is present. The table is also organized in a way that layers on the same row are concatenated.

Encoder			**Decoder**		
Layer name	**Parameters**	**Feature size**	**Layer name**	**Parameters**	**Feature size**
Input		256×256×1	Output+sigmoid	4×4×1; s=2	256×256×1
enc1	3×3×64; s=2 Lrelu	128×128×64	dec1+concat	4×4×64; s=2 relu; drop(0.5)	128×128×128
enc2	3×3×128; s=2 Lrelu	64×64×128	dec2+concat	4×4×128; s=2 relu, drop(0.5)	64×64×256
enc3	3×3×256; s=2 Lrelu	32×32×256	dec3+concat	4×4×256; s=2 relu, drop(0.5)	32×32×512
enc4	3×3×512; s=2 Lrelu	16×16×512	dec4+concat	4×4×512; s=2 relu, drop(0.5)	16×16×1024
enc5	3×3×512; s=2 Lrelu	8×8×512	dec5+concat	4×4×512; s=2 relu, drop(0.5)	8×8×1024
enc6	3×3×512; s=2 Lrelu	4×4×512	dec6+concat	4×4×512; s=2 relu, drop(0.5)	4×4×1024
enc7	3×3×512; s=2 Lrelu	2×2×512	dec7+concat	4×4×512; s=2 relu, drop(0.5)	2×2×1024
Bridge					
brd1	3×3×512; s=2	1×1×512			

ACKNOWLEDGMENTS

This work is financed by FEDER funds through "Programme opérationnel regional FEDER-FSE Pays de la Loire 2014-2020" n° PL0015129 (EPICURE).

REFERENCES

[1] Gerth, H. U., Juergens, K. U., Dirksen, U., Gerss, J., Schober, O., and Franzius, C., "Significant benefit of multimodal imaging: Pet/ct compared with pet alone in staging and follow-up of patients with ewing tumors," *Journal of Nuclear Medicine* **48**(12), 1932–1939 (2007).

[2] Kumar, A., Fulham, M., Feng, D., and Kim, J., "Co-learning feature fusion maps from pet-ct images of lung cancer," *IEEE Transactions on Medical Imaging* **39**(1), 204–217 (2019).

[3] Zhong, Z., Kim, Y., Plichta, K., Allen, B. G., Zhou, L., Buatti, J., and Wu, X., "Simultaneous cosegmentation of tumors in pet-ct images using deep fully convolutional networks," *Medical physics* **46**(2), 619–633 (2019).

[4] Li, L., Zhao, X., Lu, W., and Tan, S., "Deep learning for variational multimodality tumor segmentation in pet/ct," *Neurocomputing* (2019).

[5] Qin, C., Shi, B., Liao, R., Mansi, T., Rueckert, D., and Kamen, A., "Unsupervised deformable registration for multi-modal images via disentangled representations," in [*International Conference on Information Processing in Medical Imaging*], 249–261, Springer (2019).

[6] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., "Generative adversarial nets," in [*Advances in neural information processing systems*], 2672–2680 (2014).

[7] Wolterink, J. M., Kamnitsas, K., Ledig, C., and Išgum, I., "Generative adversarial networks and adversarial methods in biomedical image analysis," *arXiv preprint arXiv:1810.10352* (2018).

[8] Chartsias, A., Joyce, T., Dharmakumar, R., and Tsaftaris, S. A., "Adversarial image synthesis for unpaired multi-modal cardiac data," in [*International workshop on simulation and synthesis in medical imaging*], 3–13, Springer (2017).

[9] Wolterink, J. M., Leiner, T., Viergever, M. A., and Išgum, I., "Generative adversarial networks for noise reduction in low-dose ct," *IEEE transactions on medical imaging* **36**(12), 2536–2545 (2017).

[10] Yi, X. and Babyn, P., "Sharpness-aware low-dose ct denoising using conditional generative adversarial network," *Journal of digital imaging* **31**(5), 655–669 (2018).

[11] Oksuz, I., Clough, J., Bustin, A., Cruz, G., Prieto, C., Botnar, R., Rueckert, D., Schnabel, J. A., and King, A. P., "Cardiac mr motion artefact correction from k-space using deep learning-based reconstruction," in [*International Workshop on Machine Learning for Medical Image Reconstruction*], 21–29, Springer (2018).

[12] Armanious, K., Gatidis, S., Nikolaou, K., Yang, B., and Kustner, T., "Retrospective correction of rigid and non-rigid mr motion artifacts using gans," in [*2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*], 1550–1554, IEEE (2019).

[13] Mardani, M., Gong, E., Cheng, J. Y., Vasanawala, S. S., Zaharchuk, G., Xing, L., and Pauly, J. M., "Deep generative adversarial neural networks for compressive sensing mri," *IEEE transactions on medical imaging* **38**(1), 167–179 (2018).

[14] Yi, X., Walia, E., and Babyn, P., "Generative adversarial network in medical imaging: A review," *Medical image analysis* **58**, 101552 (2019).

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

[16] Huo, Y., Xu, Z., Moon, H., Bao, S., Assad, A., Moyo, T. K., Savona, M. R., Abramson, R. G., and Landman, B. A., "Synseg-net: Synthetic segmentation without target modality ground truth," *IEEE transactions on medical imaging* **38**(4), 1016–1025 (2018).

[17] Maspero, M., Savenije, M. H., Dinkla, A. M., Seevinck, P. R., Intven, M. P., Jurgenliemk-Schulz, I. M., Kerkmeijer, L. G., and van den Berg, C. A., "Dose evaluation of fast synthetic-ct generation using a generative adversarial network for general pelvis mr-only radiotherapy," *Physics in Medicine & Biology* **63**(18), 185001 (2018).

[18] Bi, L., Kim, J., Kumar, A., Feng, D., and Fulham, M., "Synthesis of positron emission tomography (pet) images via multi-channel generative adversarial networks (gans)," in [*molecular imaging, reconstruction and analysis of moving body organs, and stroke imaging and treatment*], 43–51, Springer (2017).

[19] Ben-Cohen, A., Klang, E., Raskin, S. P., Soffer, S., Ben-Haim, S., Konen, E., Amitai, M. M., and Greenspan, H., "Cross-modality synthesis from ct to pet using fcn and gan networks for improved automated lesion detection," *Engineering Applications of Artificial Intelligence* **78**, 186–194 (2019).

[20] Armanious, K., Jiang, C., Fischer, M., Küstner, T., Hepp, T., Nikolaou, K., Gatidis, S., and Yang, B., "Medgan: Medical image translation using gans," *Computerized Medical Imaging and Graphics* **79**, 101684 (2020).

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

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

[23] Kinehan, P. and Fletcher, J., "Pet/ct standardized uptake values (suvs) in clinical practice and assessing response to therapy," *Semin Ultrasound CT MR* **31**(6), 496–505 (2010).

[24] Santini, G., Fourcade, C., Rousseau, C., Ferrer, L., Campone, M., Colombié, M., and Rubeaux, M., "Segmentation automatique des métastases hépatiques en imagerie tep/tdm basée sur l'apprentissage profond dans le cadre du cancer du sein métastatique," *Médecine Nucléaire* **44**(2), 135 (2020).

[25] Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., and Paul Smolley, S., "Least squares generative adversarial networks," in [*Proceedings of the IEEE international conference on computer vision*], 2794–2802 (2017).

[26] Zhang, H., Zhang, Z., Odena, A., and Lee, H., "Consistency regularization for generative adversarial networks," *arXiv preprint arXiv:1910.12027* (2019).

[27] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X., "Improved techniques for training gans," in [*Advances in neural information processing systems*], 2234–2242 (2016).

[28] Gokaslan, A., Ramanujan, V., Ritchie, D., In Kim, K., and Tompkin, J., "Improving shape deformation in unsupervised image-to-image translation," in [*Proceedings of the European Conference on Computer Vision (ECCV)*], 649–665 (2018).

[29] Johnson, J., Alahi, A., and Fei-Fei, L., "Perceptual losses for real-time style transfer and super-resolution," in [*European conference on computer vision*], 694–711, Springer (2016).

[30] Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y., "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957* (2018).