Synthesize 3D Realistic CT Textures and Anatomy in The XCAT Phantom Using Generative Adversarial Network (GAN)

by

Yijie Yuan

Graduate Program in Medical Physics
Duke University

Date:_____

Approved:

_____
Lei Ren, Advisor

_____
Paul Segars

_____
Christopher Kelsey

Thesis submitted in partial fulfillment of
the requirements for the degree of
Master of Science in Graduate Program in Medical Physics in the
Graduate School of Duke University

2021

ABSTRACT

Synthesize 3D Realistic CT Textures and Anatomy in The XCAT Phantom Using Generative Adversarial Network (GAN)

by

Yijie Yuan

Graduate Program in Medical Physics
Duke University

Date:_____

Approved:

_____
Lei Ren, Advisor

_____
Paul Segars

_____
Christopher Kelsey

Thesis submitted in partial fulfillment of
the requirements for the degree of
Master of Science in Graduate Program in Medical Physics in the
Graduate School of Duke University

2021

Abstract

Objectives: Generate 3D realistic textures and anatomy in the thorax region of extended cardiac torso (XCAT) phantom.

Methods: We proposed to generate realistic anatomical textures and structures in the XCAT phantom using a conditional generative adversarial network (CGAN) and 3D U-net. 300 and 100 3D CT images were used for training and validation, respectively. Organ maps were generated from patient CT images with uniform intensities in each organ to mimic XCAT images. The model was trained to simulated realistic CT textures and structures in organ maps to match with real CT images. To capture the fine details and guarantee the continuity of lung vessels, we used multiple generators to generate lung vessels and other body parts separately before integrating them together. The results from the training and validation group were evaluated by measuring the SSIM, L1, and L2 metric between the real CT images and simulation images generated from organ maps. After training, XCAT phantoms were input into the model to generate textured XCAT phantoms. The result was compared to our prior work using the 2D GAN model.

Results: Compared to the 2D GAN model, the 3D model achieved higher SSIM and PSNR and lower L1 and L2 losses in the validation group. In the generated XCAT phantoms, the model generated similar contrast and anatomical textures as real CT images. The generated phantom also showed 3D continuity by visual examination from different views, especially for lung vessels. Additionally, we found the strategy of using separate generators for lung vessels and other parts greatly enhanced the realism of the generated images.

Contents

List of Tables

List of Figures

1 Introduction

1.1 Medical imaging and imaging phantom

Medical imaging refers to techniques and processes used to create images of various parts of the human body. Different modalities of medical imaging techniques include X-ray radiography (Figure 1), Computed Tomography (CT), Fluoroscopy, Magnetic resonance imaging (MRI) and among others. These days, with the rapid development of imaging techniques, medical imaging plays an important role in the patient diagnosis and treatment process. For instance, X-ray imaging is widely applied in the diagnosis of cancer. Due to the importance of medical imaging in clinical scenarios, researchers are devoted to coming up with new imaging modality and advancing current medical imaging techniques by improving imaging quality at less cost.

Figure 1. Chest X-ray image

When a new medical imaging technique is proposed, testing is a critical step for performance evaluation, which decides whether this technique can be used in the clinic after a human trial. Among all testing methods for medical imaging techniques, patient studies are a widely applied testing method that researchers acquire directly from patients. However, radiation dose is a concern in testing an ionization-based imaging system. In the ionized imaging process, ionizing radiation (e.g., X-ray) passes through the patient body

and generates electrons from incident photons. The generated electrons will be absorbed by the patient and increase the total radiation dose. For example, ionized imaging techniques such as X-ray radiography involves a large amount of radiation dose to the patient. However, exposure to very high levels of radiation can cause acute health effects such as skin burns and long-term health effects such as cancer and cardiovascular disease. Thus, the researchers need to carefully control the dose output at a low level at the expense of the experiment results' accuracy and reliability.

Several efforts have been made to solve the problem of excessive radiation dose, such as using lead plates to shield critical organs for patients and developing advanced data processing methods so that less dose is required to achieve the same results. These methods can reduce the amount of radiation dose in patients, but researchers still face the risk of excessive radiation dose given to the patient. In practice, using an imaging phantom is the most common method to test an imaging technique with minimum radiation dose to participants. It is a specially designed object that is scanned or imaged in medical imaging

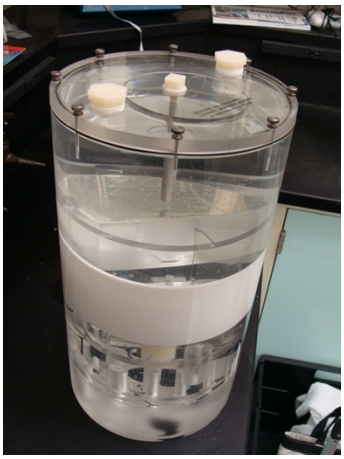

Figure 2 Water-equivalent Imaging phantom

to evaluate, analyze, and tune the performance of various imaging devices. The materials used to fabricate a phantom usually have similar properties to the human body to mimic

real situations, such as density and the ability to attenuate X-ray. The imaging phantom facilitates the researcher to evaluate the imaging system quantitatively without extra radiation dose. However, an imaging phantom is different from a real patient. It is a simplified version of a real situation in the clinic. To match with clinical needs, researchers invented various types of imaging phantoms for different goal setting.

1.2 Digital phantom and XCAT phantom

Digital phantom is a commonly used tool in testing new medical imaging or therapy techniques. Given a mathematical model of the imaging process, we can simulate images of different modalities from the phantom. Among various computational anthropomorphic phantoms, the 4D extended cardiac-torso (XCAT) phantom (Figure 3) provides a highly detailed representation of whole-body anatomies for the adult male and female. Images of different modalities can be simulated from the XCAT phantom and be used in biomedical imaging and therapy research, such as medical imaging technique evaluation, radiation dosimetry, radiation therapy technique evaluation. Moreover, user-defined characteristics dramatically increase the degree of freedom to manipulate simulated data.

Nevertheless, one of the limitations of the XCAT is the lack of intra-organ textures and over-simplified structures such as lung vessels. For example, one slice of a simulated CT image of the XCAT phantom is shown in figure 4. Most organs in the XCAT phantom are defined as homogeneous. And the absence of heterogeneous textures limits the feasibility of XCAT phantoms in research related to image quality attributes, such as contrast, noise, and resolution.

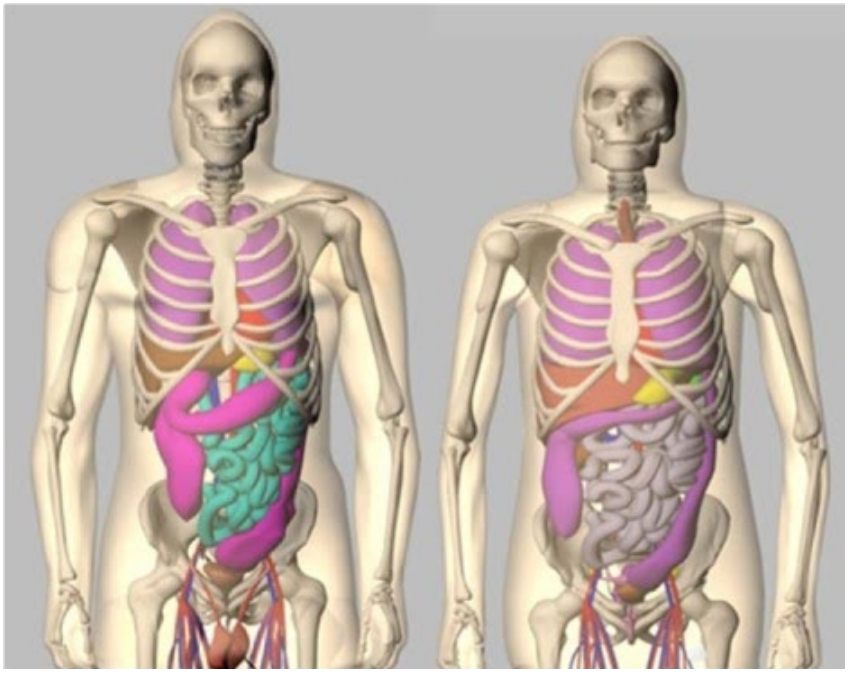

Figure 3. Anterior views of the male (left) and female (right) anatomies of the 4D XCAT phantom (Segars, 2008).

Figure 4. Axial views of a CT slice simulated from 4D XCAT phantom

1.3 Generative Adversarial Networks (GAN)

1.3.1 Vanilla GAN

Generative Adversarial Networks (Goodfellow, 2014), or GANs for short, is a sub-filed of deep learning. GAN provides a new training method for networks in a higher level that differs from traditional deep learning. GAN calculates the loss function using a separate model named 'discriminator' the and have shown to be well suited for the generation of photo-realistic images across a range of problem domains, most notably in image-to-image translation tasks such as translating photos of day to night, and in generating photorealistic photos of objects such as human faces that even humans cannot tell are fake or not.

Vanilla GAN is the simplest type of GAN. The workflow of a Vanilla GAN is shown in Figure 5. The GAN's framework is a minimax two-player game. GANs are generative models that learn a mapping from random noise vector z to output image y, $G: z \rightarrow y$. The generator G is trained to produce outputs y that cannot be distinguished from "real" images x by an adversarially trained discriminator, D, which is trained to do as well as possible at detecting the generator's "fakes". In this adversarial training process, both generator and discriminator are getting more powerful. And more realistic images can be generated.

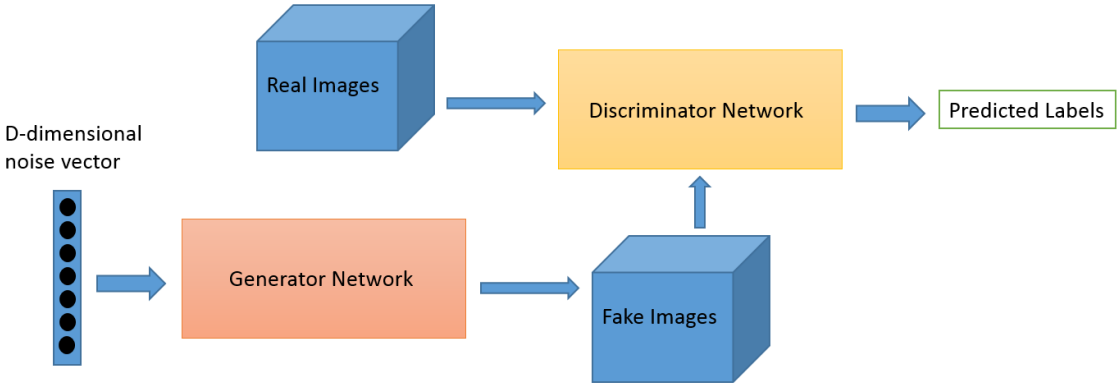

Figure 5. GAN workflow

To train the generator and discriminator, the cost function is calculated to provide gradients used for updating the models, which is similar to the back-propagation method in training a model for a general machine learning task. The generator is trained to minimize $\mathbb{E}_{z\sim p_z(z)}\log\left(1-D\left(G(z)\right)\right)$, and the discriminator is trained to maximize $\mathbb{E}_{x\sim p_x(x)}\log\left(D(x)\right)+\mathbb{E}_{z\sim p_z(z)}\log\left(1-D\left(G(z)\right)\right)$, where $p_z(z)$ is the distribution of input noise z and $p_x(x)$ is the distribution of real images x. Usually, the noise input is a random D-dimensional vector that aligned with the input dimension.

Although, there are still problems remaining in the GAN training, such as unstable training and mode collapse. Many medical image analysis and processing applications could also benefit from artificial datasets of realistic images, e.g. by generating ground truth data for augmentation purposes.

1.3.2 Image-to-Image translation using Conditional GAN

In contrast to Vanilla GAN, conditional GANs, or CGANs for short, learn a mapping from observed images or labels x and random noise vector z, to y, G: {x, z} → y. The input to the generator is the combination of both condition and noise. In 2018, Isola et al. released Pix2Pix software and proposed to use CGANs for as a general-purpose solution to image-to-image translation problems. As shown in figure 6, if we want to train the generator to generate colored shoes from the sketch, then the sketch is our condition, we need to encode

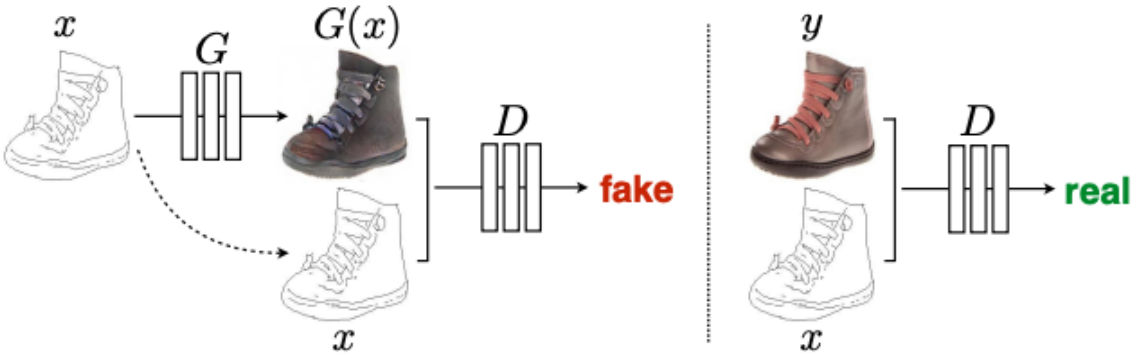

Figure 6. Training a conditional GAN (CGAN) to map edges → photo. (Isola, 2018)

the noise and combine it with the sketch, then use the combination as the input to the generator. For the discriminator, it is only expected to output real when its input is the correct pair of sketch and shoes from training dataset.

The cost function, or objective of a CGAN can be expressed as,

$$\mathbb{L}_{CGAN}(\mathrm{G}, \mathrm{D}) = \mathbb{E}_{\mathrm{x,y}}\log\big(D(x,y)\big) + \mathbb{E}_{\mathrm{x,z}}\log\Big(1 - D\big(x, G(x,z)\big)\Big)$$

In Isola's work, it also mentioned that mixing the GAN objective with a more traditional loss, such as L1 distance will be beneficial. The generator is tasked to not only fool the discriminator but also to be penalized by the distance to the ground truth in an L1 sense.

After mixing the L1 loss, the final cost function is

$$\mathbb{L}_{L1}(\mathrm{G}) = \mathbb{E}_{\mathrm{x,y,z}}[\left||y - G(x,z)|\right|_1]$$

$$\mathbb{L}^*(\mathrm{G},\mathrm{D}) = \mathbb{L}_{CGAN}(\mathrm{G},\mathrm{D}) + \lambda \times \mathbb{L}_{L1}(\mathrm{G})$$

, where λ is a weight factor that control the ratio of traditional L1 loss and GAN objective. A proper λ will be found through experiments.

In these two years, the framework of pix2pix has been widely applied in medical image translation tasks such as CT to MRI translation (Lei, 2019). More applications and development of CGAN will be discussed in the next section.

1.4 Related Work

Conditional GANs have been developed on discrete labels, such as generating an image of a specified Arabic number (Mirza. 2014). Several other papers have also used GANs for image-to-image mappings, but only applied the GAN unconditionally, relying on other terms (such as L2 synthesis loss) to force the output to be conditioned on the input. These papers have achieved impressive results on inpainting (D. Pathak, 2016). Conditional GAN was first used in image-to-image translation in the Pix2Pix paper (Isola, 2017), Isola et al. used a "U-Net"-based generator architecture and a convolutional "PatchGAN" (Demir, 2018) classifier as discriminator, which only penalizes structure at the scale of image patches. After this, GAN has been used for image translation between different imaging modalities such as CT to MRI (Lei, 2019), CT reconstruction (Ying, 2019) and other image tasks. And GAN has been used in enhancing the XCAT phantom. In 2020, Chang developed a 2D dual-discriminator conditional-generative adversarial network (D-CGAN) to synthesize anatomical textures in the XCAT. As figure 7 shows, the generated 2D textured XCAT slice has similar contrast and HU values to real CT images but lacks continuity between adjacent slices. 3D applications using GAN are very limited. Wu et al. used a 3D-GAN to generate 3D objects such as chairs, cars in low resolution(64x64x64). Moschoglou et al. developed a 3DFaceGAN to for generating 3D human face representation. And GAN has also been used in 3D medical imaging translation. To solve the large GPU memory in 3D GAN training, a memory-efficient GAN was proposed for 3D medical image translation tasks (Uzunova, 2020). Although more and more advanced GAN structures have been developed, practice in 3D images is still very rare. And most of tasks are basically style-transfer tasks. In our task, we have developed our GAN structure

based on previous method to generate various anatomy and textures including vessel structures from a uniform organ map.

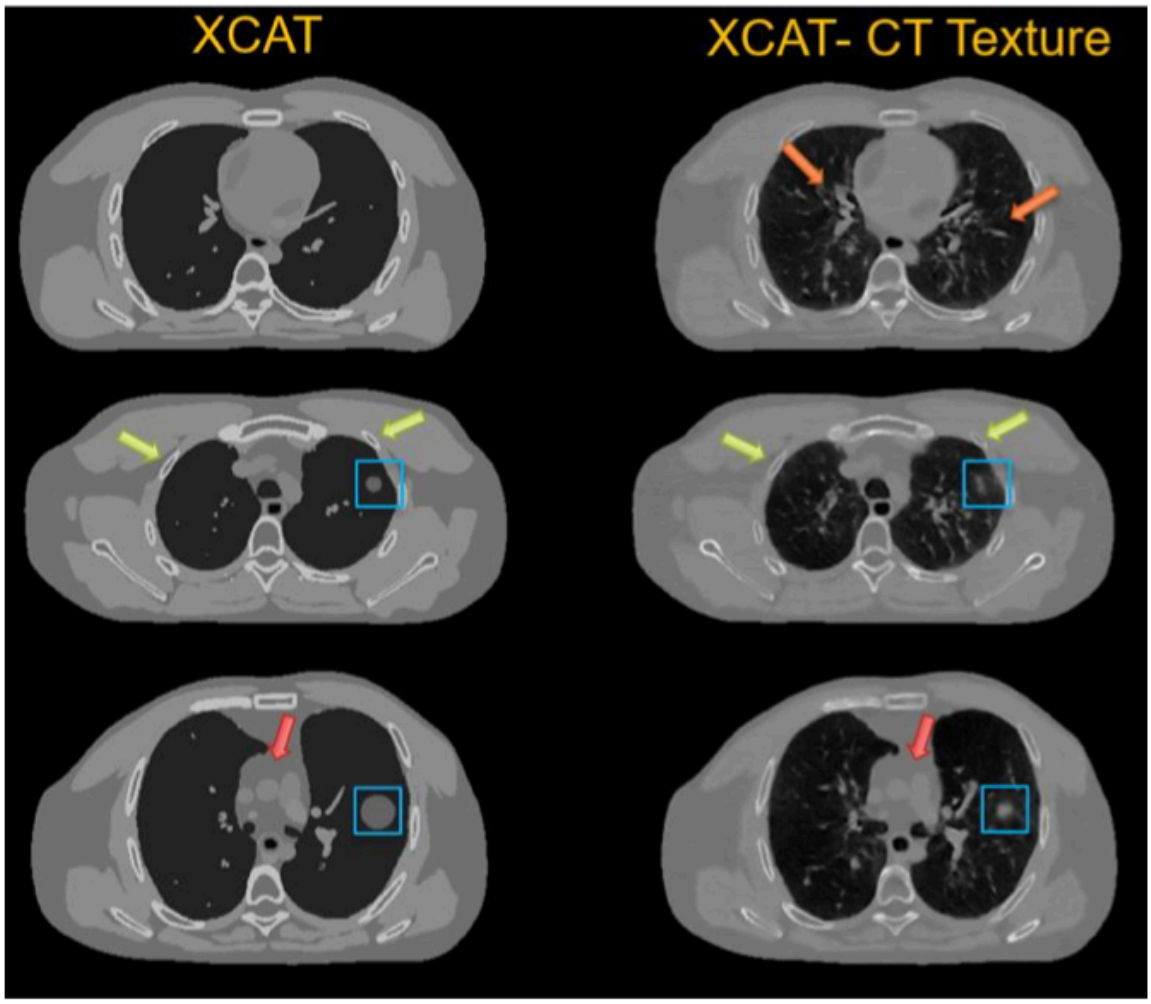

Figure 7. Examples of textured XCAT images generated by 2D model (Chang, 2020).

2. Method

2.1 Dataset

400 non-small cell lung cancer (NSCLC) patients were collected and selected from The Cancer Image Archive (TCIA).

2.1.1 Data preprocessing

First, we cleaned out defective data from the dataset. The TCIA dataset contains the CT image and various contours for lungs, heart, esophagus, and gross tumor volume (GTV). We noticed that some patient data missed one or two types of contours mentioned above and cleaned them out. Since the TCIA dataset didn't have body contour, we used an automated body segmentation algorithm (Özsavaş, 2014) to obtain body masks from CT images.

Each slice of our collected image has a resolution of 512 x 512. But we noticed that there is a large gap between the body contour and the image border. To standardize the size of image input to the generator and reduce the image size. As figure 8 shows, we first cropped the CT images with 8 pixels left from the body contour to the margin. And then, for each slide, we scaled the image size in the coronal and sagittal direction into 256 pixels and scaled the size in the axial direction into 128 pixels. Also, it is a good practice to normalize the pixel value to the range of 0 to 1, which reduces the computation cost in training the model. In this study, we set all pixel value greater than 1000 or less than -

1000 to 1000 and -1000 respectively, and then divide all pixel value by 2000 plus all values with 1.

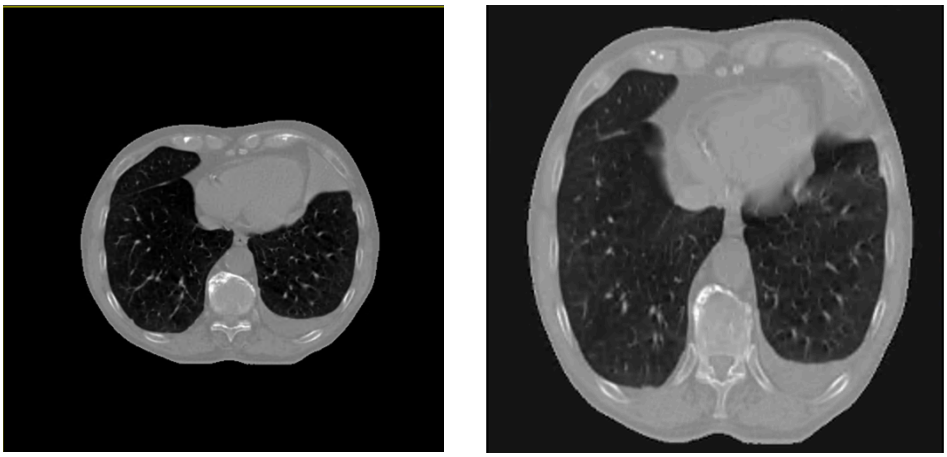

Figure 8. An example of axial slice of original CT image (left) and processed CT image (right)

2.1.2 Organ map generation

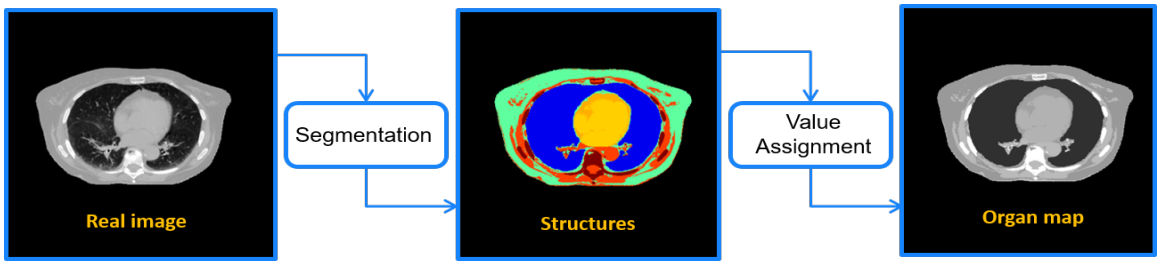

Figure 9. Organ map generation process

In this study, pairs of organ map and real CT image were created to train the models. We used organ map to mimic XCAT phantom and used the generation from organ map to real CT images to mimic the process of texture and anatomy generation. Organ maps were generated using the method from the previous paper (Chang, 2020). First, we segmented real CT images based on structures defined in the XCAT generation procedure. After segmentation, we assigned a distinct value to each organ that we segmented (Figure 9) (Chang, 2020). As Table 1 shows, the assigned value was decided by the average HU value in the region calculated from the CT dataset collected (Chang, 2020).

Table 1. Value assignment for Organ map generation

	Structure	Segmentation condition	HU assigned to organ map
Manual Contour	Lungs	N.A.	- 700
	GTV	N.A.	- 200
Threshold	Bones	HU > 150	500
	Muscle	- 15< HU <= 150	100
	Body background	- 800 < HU <= - 15	- 100
Manual Contour + threshold	Airway	HU < - 800	- 920
	Myocardium	HU > 0	50
	Heart Chamber	HU <= 0	- 50

2.2 Workflow

The workflow of synthesizing 3D realistic CT textures and anatomy in the XCAT phantom is shown in Figure 10. Considering the difficulty of generating 3D objects, we split this synthesis task into two sub-tasks and used a two-step method in training our model. Each step's generator is tasked to generating textures and anatomy for parts of the XCAT/organ map.

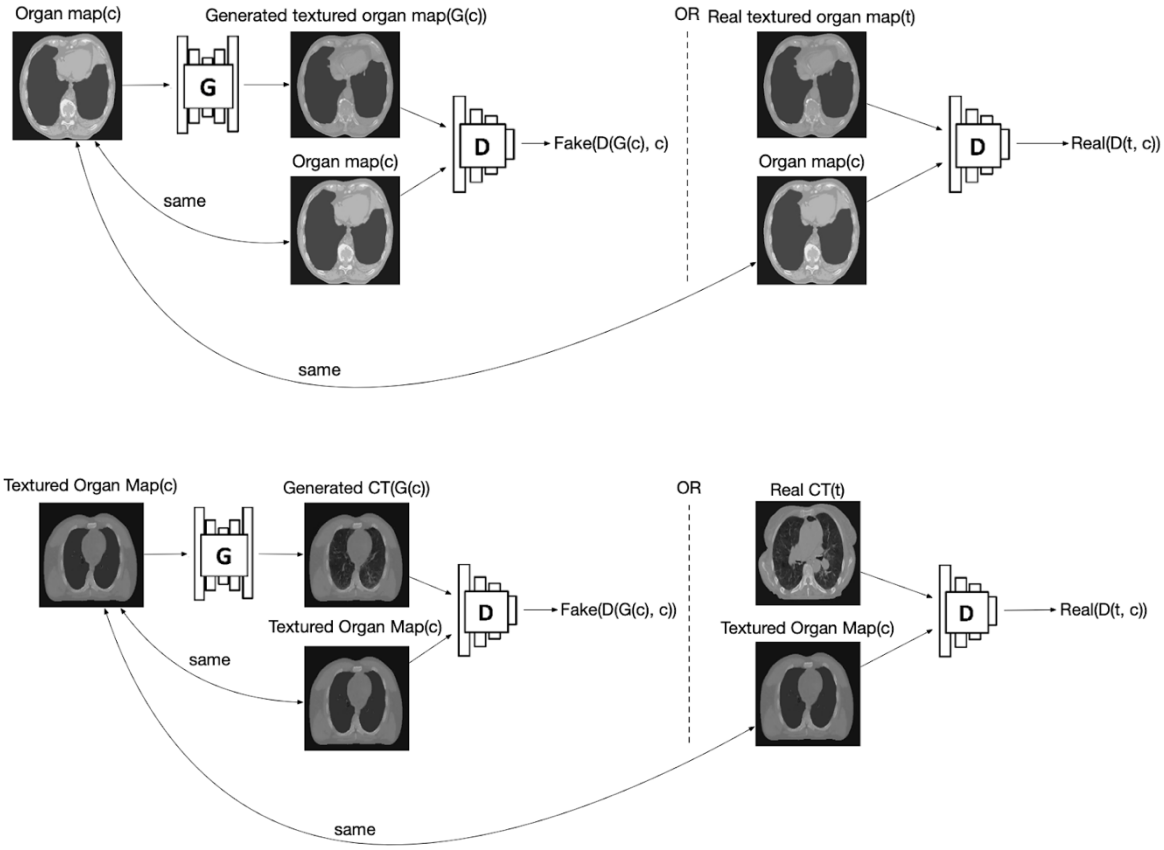

Figure 10. Steps for training the models. The generator in the first step (top) is trained to generator textures for body parts except lungs. The generator in the second step (bottom) is trained to generate textures and anatomy for lung. Where c is the condition or the input to the generator, t is the target image for generator. G is the generator and D is the discriminator.

In the first step, we trained the generator to add textures and anatomy for all body parts in the organ map except for the lungs. The training target is real CT image with a uniform lung. The generated result in step 1 is named 'generated textured organ map'. As described in the workflow of Pix2Pix that we mentioned in section 1.3.2, the generated textured organ map is compared with the target image to calculate the synthesis loss $\mathbb{L}^*$,

$$\mathbb{L}^*(\mathrm{G},\mathrm{D}) = \mathbb{L}_{CGAN}(\mathrm{G},\mathrm{D}) + \lambda \times \mathbb{L}_{L1}(\mathrm{G})$$

This loss is calculated in each batch of training to update both generator and discriminator until the training ends. With the definition of our objective, we can use the gradient descent method to update the discriminator and maximize its output when we used real images (Organ map for lung and real CT image for other body parts). The discriminator's output can be interpreted as the probability of the input image being real, which ranges from 0 to 1. The generator was updated to fool the discriminator and minimize the output of the discriminator when the input to the discriminator is a combination of the textured organ map and the organ map used to generate it.

In the second step, we used the same training strategy, but another generator was trained to synthesize textures and anatomy to the uniform lung regions in the generated textured organ map. Next, The generated CT is input into the discriminator concatenated with the textured organ map used to generate it. The synthesis loss is then calculated to update the models.

After training, two generator works successively to add textures and anatomy in the XCAT phantom for evaluation (Figure 11).

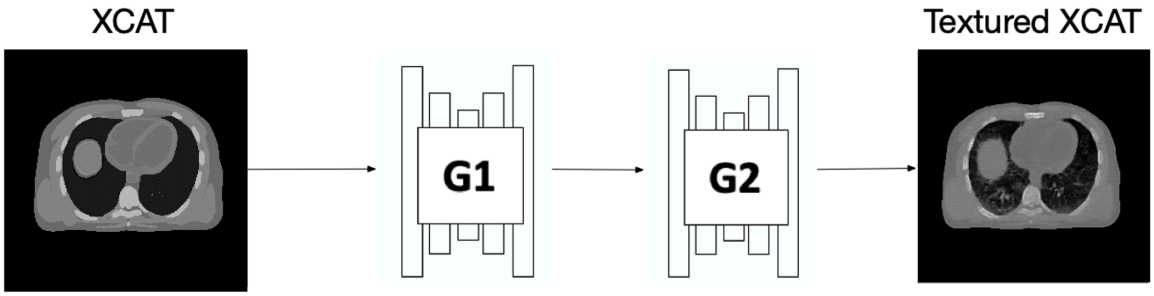

Figure 11. 2 steps of Generating texture and anatomy in the XCAT phantom

2.3 Model architecture

2.3.1 Network structures

The network structure of the generator and discriminator are illustrated in Figure 12 and 13. The basic network structure is from 3D U-net and we added some modification for this task. The U-net is widely used in medical image synthesis and segmentation tasks. Low-level information can be shared between the input and output through the skip connection. Both generator and discriminator use modules of the form of convolution-LeakyReLu in the down-sampling blocks and use convolution-ReLu in the up-sampling blocks. In the generator structure, stride of 2 x 2 x 2, convolution kernel of 4 x 4 x 4, and same padding (Expects padding to be such that the input and output is of the same size) was used in each block. The feature channels increase at each down-sampling layer and decreases at each up-sampling layer. Compared with other commonly used U-net structures, we removed batch normalization layers and dropout layers.

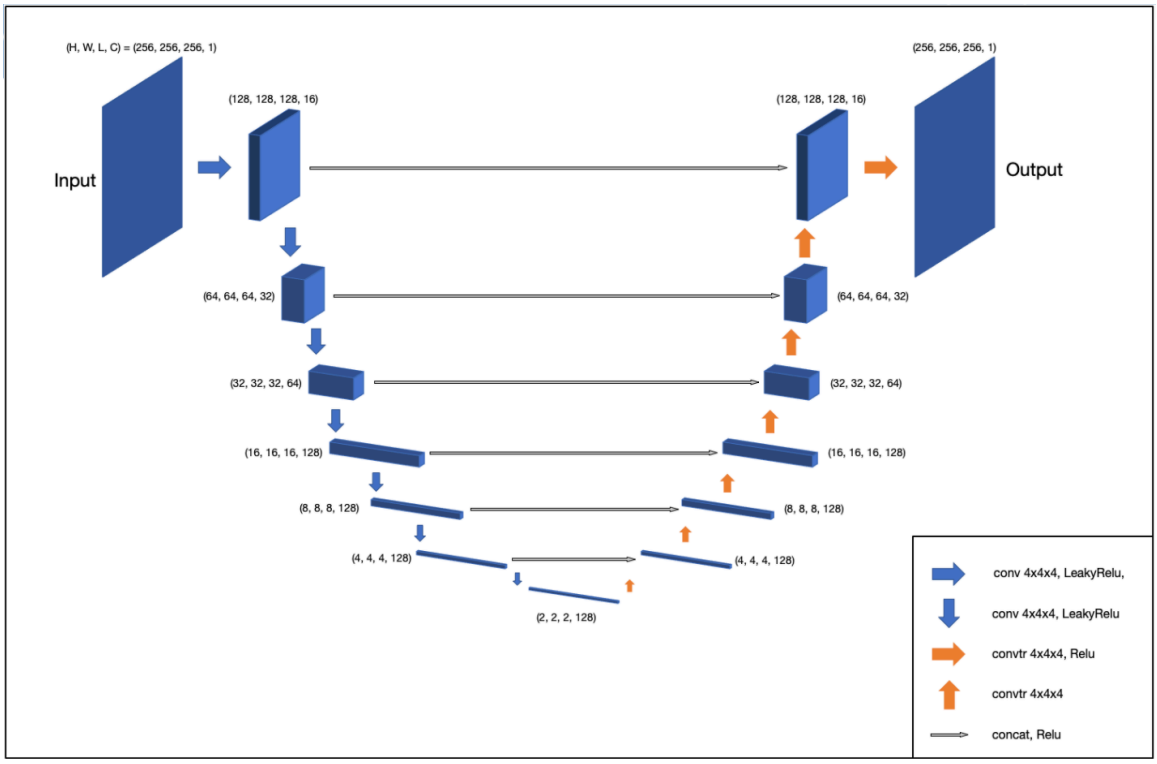

Figure 12. Network structure of the generator

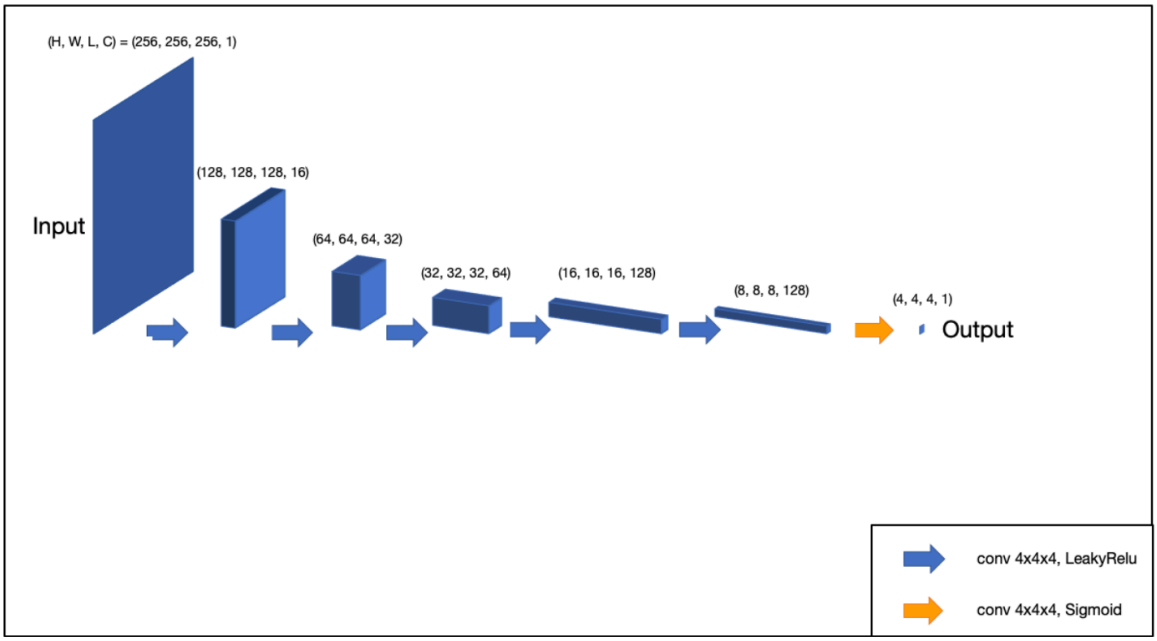

Figure 13. Network structure of the discriminator

2.3.2 Loss function

Similar to the original GAN loss. In this study, the loss function of the generator can be expressed as,

$$L_G = L_{CGAN-G} + \lambda \times L_{L1}$$

$$L_{CGAN-G}(G, D) = E_{G(c)}\left[\log\left(1 - D(G(c),\ c)\right)\right]$$

$$L_{L1} = E_c[\left|\left|t - G(c)\right|\right|_1]$$

And the loss function for the discriminator can be expressed as,

$$L_D = L_{CGAN-D}$$

$$L_{CGAN-D}(G, D) = E_{G(c)}\left[\log\left(D(G(c),\ c)\right)\right] + E_t[log(1 - D(t, c))]$$

Where G and D are the generator and the discriminator respectively, c is the condition, G(c) is the output of generator and t is the target image. L_{L_1} is the L1 synthesis loss. Here the combinations of L1 with adversarial loss is used. According to Pix2Pix (Isola, 2017), L1 alone leads to reasonable but blurry results, while the adversarial loss alone gives much sharper results but introduces visual artifacts on. We used the L1 loss weight factor λ of 10 in practice.

2.3.3 Experimental setup

Parameters or hyperparameters we used in training and validation are shown in Table 2.

Table 2. Parameters/Hyperparameters used in training and validation

Parameters/Hyperparameters	Value
Optimizer	Adam
Batch size	1
Training data size	300
Validation data size	100
Learning rate	1e-4

2.4 Evaluation

We evaluated the performance of a well-trained generator in validation and test. In the validation, the generator is tasked to add textures and anatomy in the organ map to generate fake CT images. L1 synthesis loss, L2 synthesis loss, Peak signal-to-noise ratio (PSNR), and structural similarity index (SSIM) were calculated between the generated images and the validation cohort's real images (Figure 14). The model was fine-tuned based on the validation results.

In the test, the generator is tasked to generate textured XCAT phantom. Considering that there was no ground truth to evaluate the, we mainly used qualitative evaluation textured XCAT phantoms. Different views of textured XCAT such as 2D axial/sagittal view and 3D views generated by the model were put together to be compared. To generate a 3D view, we used the 3D view function in ImageJ software and tuned the transparency of the displayed image to make the vessels visible. Further improvement can be made on quantitative evaluation methods or implementing an observer study.

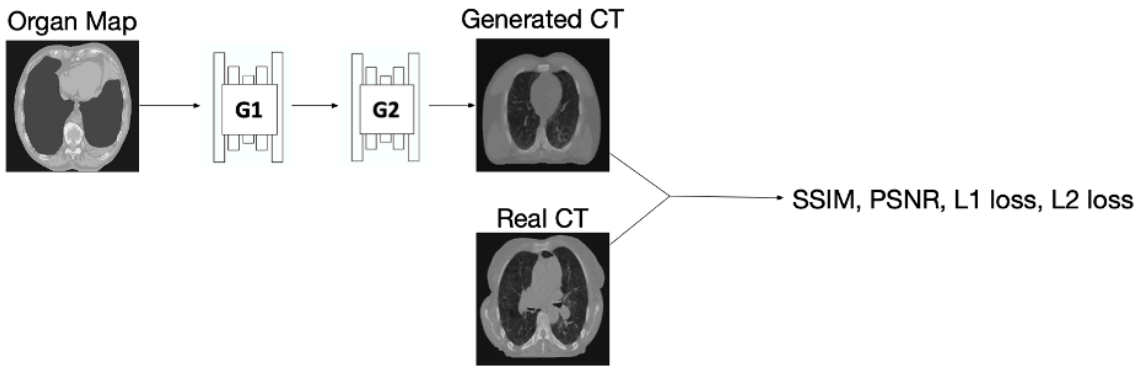

Figure 14. Quantitative evaluation of validation results.

3. Results

The results include textured XCAT phantoms with CT textures in the test and the textured organ map and quantitative evaluations in the validation. The overall training took approximately 8 hours on an NVIDIA Quadro RTX 8000 GPU. Generating a textured XCAT phantom took only about several seconds.

3.1 Quantitative Evaluation

To quantitatively evaluate the model performance, we calculated SSIM, PSNR, L1 and L2 synthesis loss for the CT images generated from organ maps in the validation group. As table 3 shows, our results performed better in terms of all metrics compared with results generated from 2D model.

Table 3. Quantitative evaluation in validation group

Methods	SSIM	PSNR	L1	L2
2D model	0.8653 ± 0.0235	26.48 ± 1.13	0.019 ± 0.003	0.0023 ± 0.0006
3D model	0.8793 ± 0.0266	28.35 ± 1.11	0.015 ±0.003	0.0015 ± 0.0004

3.2 Visual Examination

3.2.1 Results in 2D views

Figure 15 and Figure 16 shows the 2D axial and sagittal view of the generated image in the validation and test. The first row illustrates the result in the validation—an axial slice of real CT image and a textured slice generated from the organ map. And the second row shows the test results. We illustrate the texture generation performance for 3D and 2D models in the XCAT phantom.

In the validation, similar general contrast and anatomical textures as real CT images were observed, and we can also find a clear structure of lung vessels. If we slide through the axial direction, we can see the vessels structures are continuous between adjacent slices.

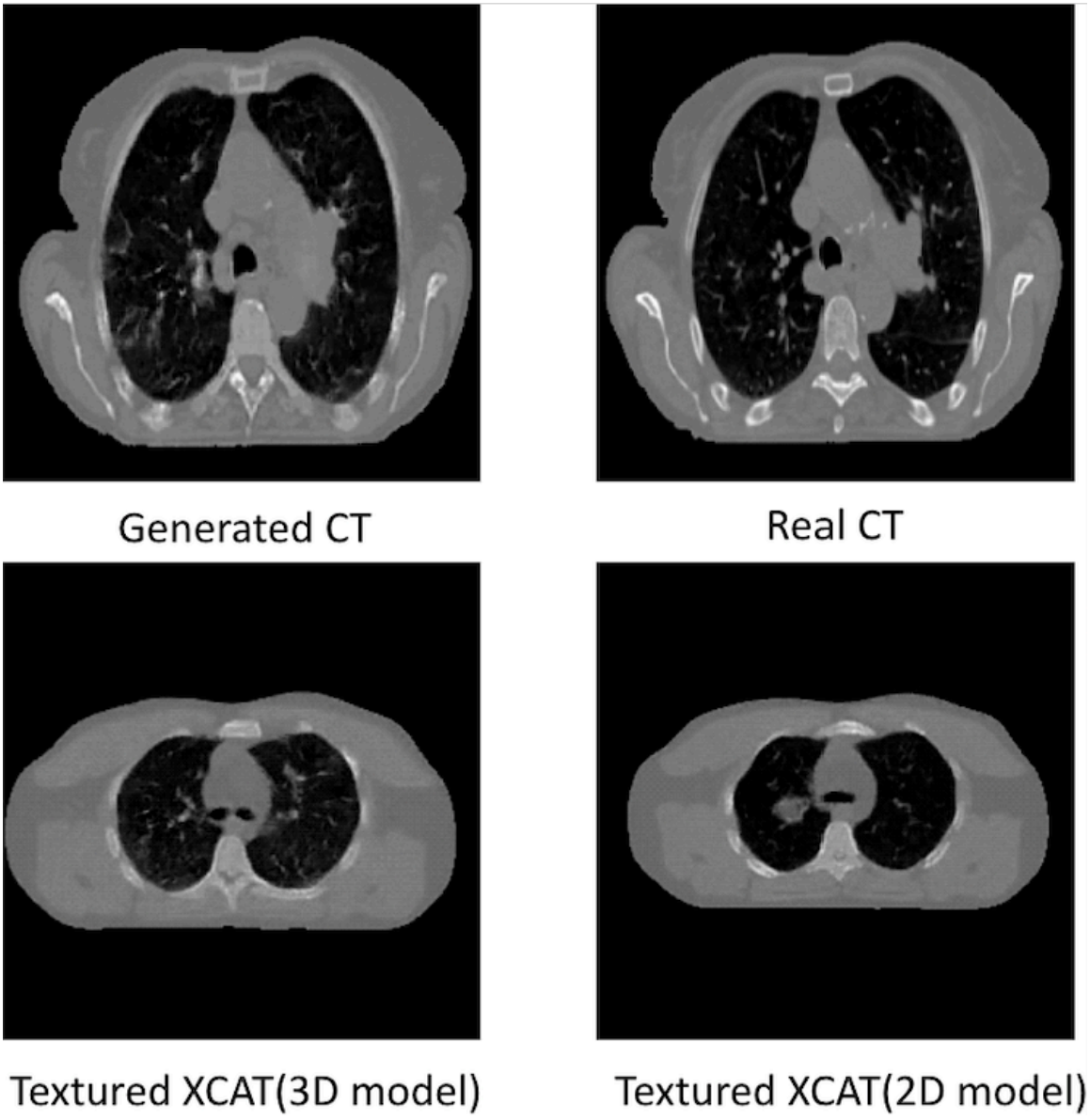

Figure 15. Axial view of generation results from 2D and 3D models in the validation and test

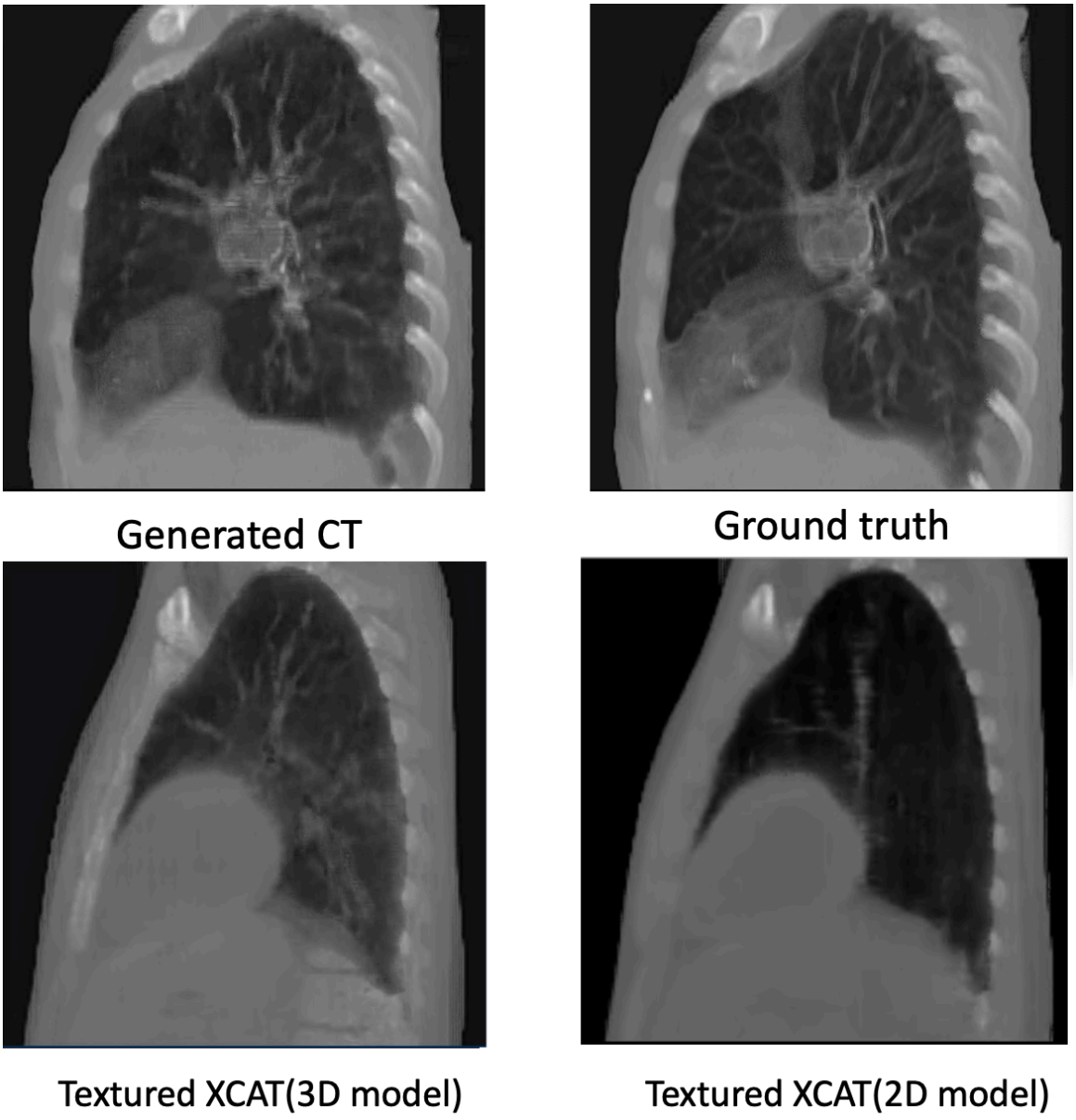

Figure 16. Sagittal view of generation results from 2D and 3D models in the validation and test

3.2.2 Results in 3D views

Figure 17 shows the 3D axial view of the generated image in the validation and test. The first row illustrates the result in the validation—a 3D real CT image and a 3D CT image generated from the organ map. And the second row shows the test results. From 3D view, we can clearly observe that the generated CT and texture XCAT image has much better continuity compared with results generated from 2D model. While the vessel structures in the result from 2D model is blurrier and straighter, tree-like structures appears in the generated CT and textured XCAT generated by 3D model, which is consistent with what we observed in 2D view. More Animations for the generation results can be found in this link: https://github.com/YijieYuan/AAPM_abstract_example.

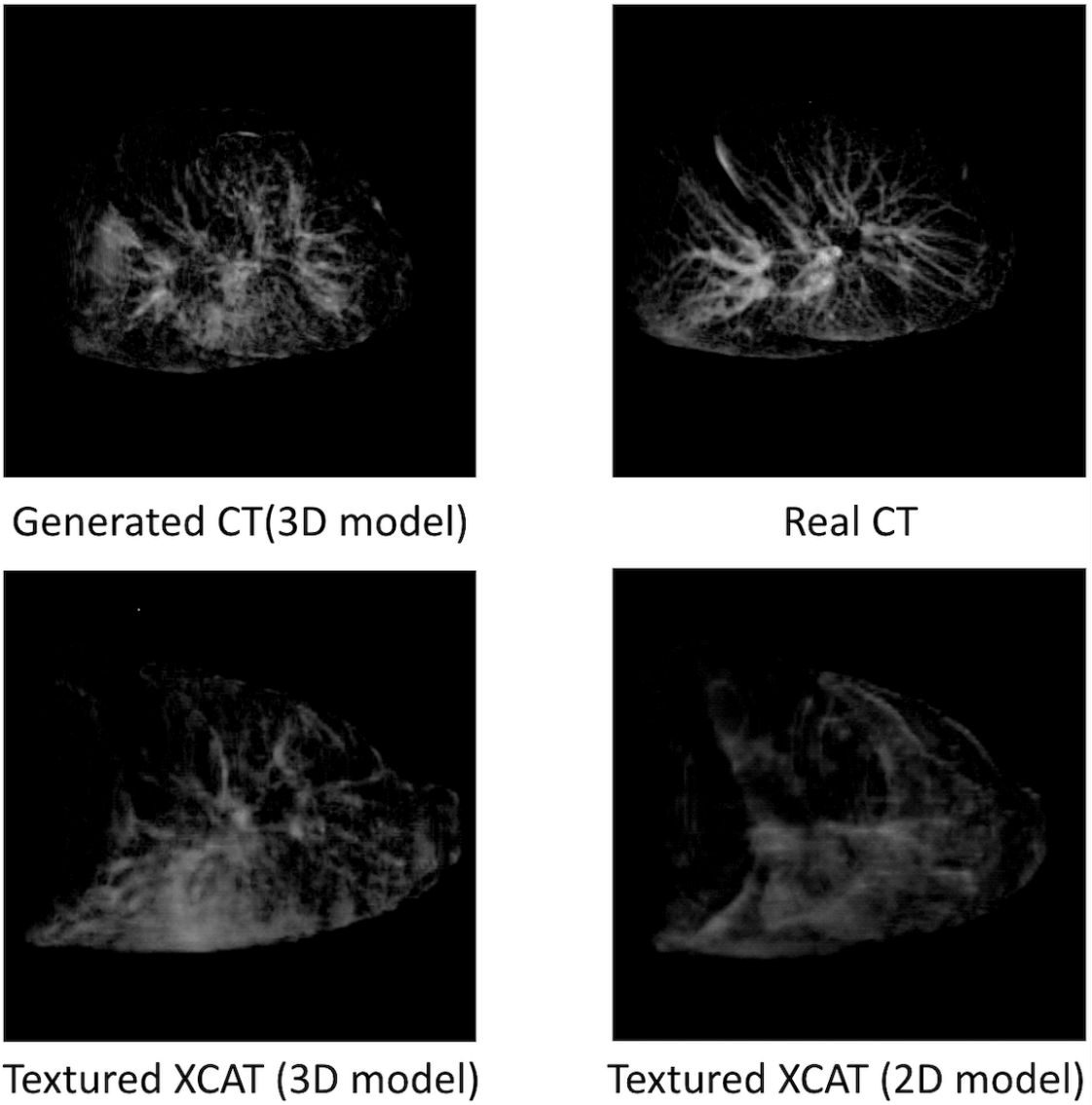

Figure 17. Examples of generation results from 3D and models in the validation and test

4. Discussion

4.1 Instability of Training

As mentioned in the introduction, one of the biggest challenging in GAN training is instability. Figure 19 shows an example of the generator loss plot in the entire training process. As we can see from the plot, unlike other non-adversarial model training processes, the generator loss is fluctuating while training rather than keeps decreasing. This is because the simultaneous training of generator and discriminator models in GANs is inherently unstable. When the discriminator has been trained to be good enough and can easily discriminate generated images from real images, the generator will not be provided with enough gradients to make progress, vice versa. So, with the training going, one of the generator and discriminator loss will indicate one of them has defeated the other model. To overcome GAN training's instability, we have tried various groups of parameters and hyperparameters and optimized the network structures. We also tried to implement Wasserstein GAN(WGAN) (Arjovsky, 2017), an extension to the GAN that improves the stability when training the model and provides a loss function that correlates with the quality of generated images. Although the training is more stable, we found WGAN doesn't give us better generation results.

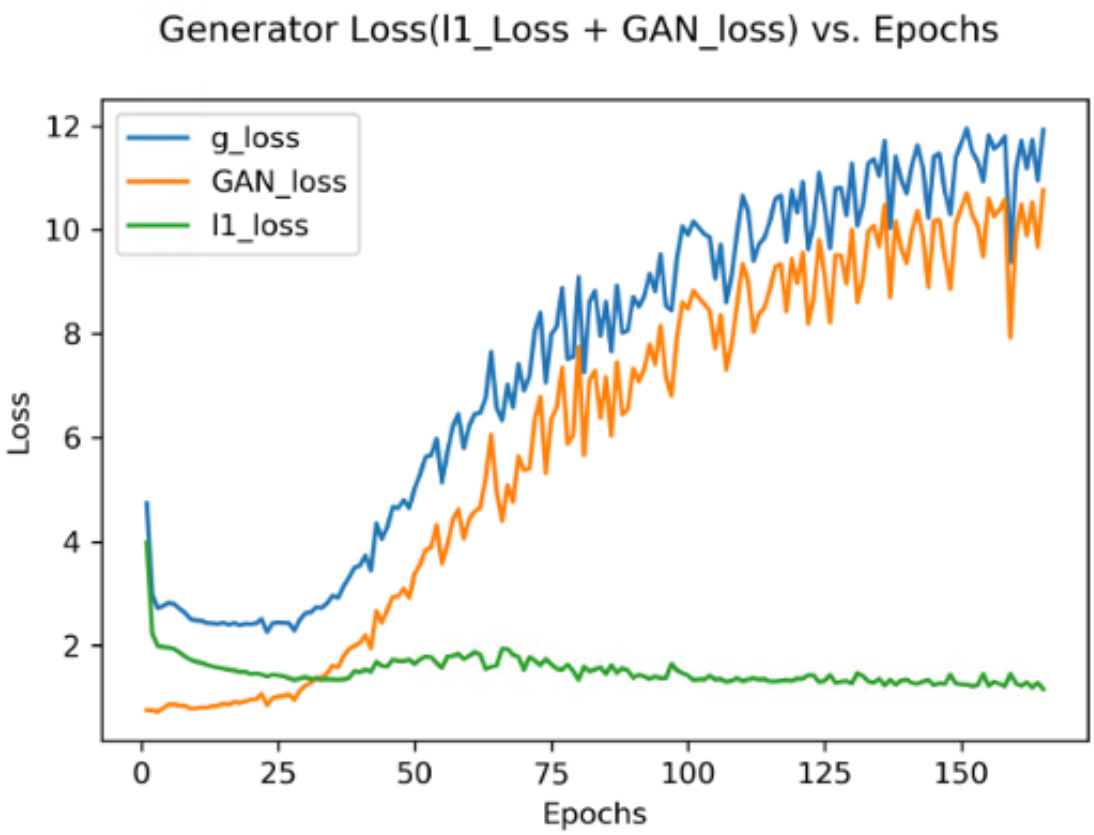

Figure 18. An example of training loss curves

4.2 Model performance is sensitive to parameters

Although we have trained the model to generate textured XCAT with continuous anatomy and textures. We noticed that our model is very sensitive to the choices of parameters. For example, if we adjust the learning rate from 1e-4 to 1e-5, the best results generated in this training cycle will look much poorer. And we own this sensitivity to the difficulty of generating 3D objects.

4.3 Evaluation methods

In this study, we used both qualitative evaluation and quantitative evaluation. For GAN, there is no ground truth of generated result to do the direct calculation. Finding a metric that aligned with human perception is difficult, especially for GAN, even for the metrics we used in this study, such as SSIM and PSNR. Previous work has developed a radiomics-based quantitative evaluation method (Chang, 2020). But in our experiments, the pass rate of radiomics distribution in the Wilcoxon-ranksum test can be apparently different for two images even if they look the same to human eyes. In further study, we can try to measure the Fréchet Inception Distance (FID) (Heusel, 2017), or Earth-move Distance (ED) for the distributions of generated images and real images.

5. Conclusion

One critical limitation of the XCAT phantom is the lack of intra-organ anatomical details and textures. Previously we developed a dual-discriminator conditional-GAN(D-CGAN) algorithm to generate realistic anatomical textures in the XCAT phantom. However, this method used a 2D generation model. In the results, generated textured XCAT phantom lacks continuity between adjacent slices, especially for lung vessels. This limitation impairs the usage of textured XCAT phantom in 3D tasks. In this study, we expanded the algorithm from 2D to 3D to improve the continuity between adjacent slices for the textured XCAT phantom. To achieve this, we used separate generators to generate lung vessels and other body parts and then integrate all parts, which reduced the training difficulty and helped the model capture the fine details in specified regions. Our proposed method can also be generalized to other body parts other than the chest region. In a word, The study demonstrated the feasibility of generating realistic 3D textures in XCAT phantoms using the proposed method. This development dramatically enhances the phantom's value for various clinical applications, such as testing and optimizing novel imaging techniques. Further improvement can be made on quantitative evaluation methods.

References

Segars WP, Mahesh M, Beck TJ, Frey EC, Tsui BM. Realistic CT simulation using the 4D XCAT phantom. Med Phys. 2008 Aug;35(8):3800-8. doi: 10.1118/1.2955743. PMID: 18777939; PMCID: PMC2809711.

Goodfellow, Ian J., et al. "Generative adversarial networks." arXiv preprint arXiv:1406.2661 (2014).

D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In CVPR, 2016.

Mirza, Mehdi, and Simon Osindero. "Conditional generative adversarial nets." arXiv preprint arXiv:1411.1784 (2014).

Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

Demir, Ugur, and Gozde Unal. "Patch-based image inpainting with generative adversarial networks." arXiv preprint arXiv:1803.07422 (2018).

Lei Y, Harms J, Wang T, Liu Y, Shu HK, Jani AB, Curran WJ, Mao H, Liu T, Yang X. MRI-only based synthetic CT generation using dense cycle consistent generative adversarial networks. Med Phys. 2019 Aug;46(8):3565-3581. doi: 10.1002/mp.13617. Epub 2019 Jun 12. PMID: 31112304; PMCID: PMC6692192.

Ying, Xingde, et al. "X2CT-GAN: reconstructing CT from biplanar X-rays with generative adversarial networks." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019

Chang, Yushi, et al. "Development of realistic multi-contrast textured XCAT (MT-XCAT) phantoms using a dual-discriminator conditional-generative adversarial network (D-CGAN)." Physics in Medicine & Biology 65.6 (2020): 065009.

Wu, Jiajun, et al. "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling." arXiv preprint arXiv:1610.07584 (2016).

Moschoglou, Stylianos, et al. "3dfacegan: Adversarial nets for 3d face representation, generation, and translation." International Journal of Computer Vision 128 (2020): 2534-2551.

Uzunova, Hristina, Jan Ehrhardt, and Heinz Handels. "Memory-efficient GAN-based domain translation of high resolution 3D medical images." Computerized Medical Imaging and Graphics 86 (2020): 101801.

Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein generative adversarial networks." International conference on machine learning. PMLR, 2017.

Heusel, Martin, et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium." arXiv preprint arXiv:1706.08500 (2017).