Synthesis of 3D Realistic High-resolution Lung Background Textures Using a Conditional Generative Adversarial Network (CGAN)

by

Yuhao Wang

Graduate Program in Medical Physics
Duke Kunshan and Duke University

Date:_____

Approved:

Paul Segars
_____
Paul Segars, Supervisor

Ehsan Abadi
_____
Ehsan Abadi

James Bowsher
_____
James Bowsher

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 Kunshan and Duke University

2022

ABSTRACT

Synthesis of 3D Realistic High-resolution Lung Background Textures Using a Conditional Generative Adversarial Network (CGAN)
by

Yuhao Wang

Graduate Program in Medical Physics
Duke Kunshan and Duke University

Date:_____
Approved:

Paul Segars
Paul Segars, Supervisor

Ehsan Abadi
Ehsan Abadi

James Bowsher
James Bowsher

An abstract of a 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 Kunshan and Duke University

2022

Abstract

Objectives: We develop machine-learning based methods to synthesize lung textures within computational phantoms for improved realism in simulating high-resolution patient CT imaging data to evaluate and improve imaging devices and techniques.

Methods: We first optimized a previously developed technique designed using a Conditional Generative Adversarial Network (CGAN), Project 1. The optimized model was trained and validated using clinical CT data. Generated texture images were evaluated qualitatively and quantitatively comparing them to the original CT data as well as to results from the previous work. Using what we learned from Project 1, in Project 2, we trained and validated a new generator using high-resolution micro-CT data of the lungs. The new generator was evaluated in a similar fashion.

Results: For Project 1, the model was unable to produce results better than the previous work; lung textures were found to be blurry and lacked detail. For Project 2, the trained generator was found capable of simulating variable 3D lung background textures similar to the micro-CT both qualitatively and quantitatively.

Conclusion: The CGAN method developed in this work, based on micro-CT data, can greatly improve the realism of computational phantoms by adding high-resolution background textures to the lungs. Such anatomical detail is necessary to evaluate higher-resolution CT imaging methods such as photon-counting CT.

Contents

List of Tables

List of Figures

1. Introduction

1.1 Medical Imaging Simulation

Medical imaging simulation or virtual imaging trials (Abadi, 2020) is a highly cost-effective method to investigate and improve the performance of medical imaging modalities. During a virtual imaging trial, computational phantoms (virtual patients) are imaged with algorithms that model the imaging process (virtual scanners), including all the artifacts of the imaging devices, to produce simulated patient images (Figure 1). In medical imaging simulation, phantoms provide a gold standard for comparison as you can compare the simulated data to the known phantom anatomy

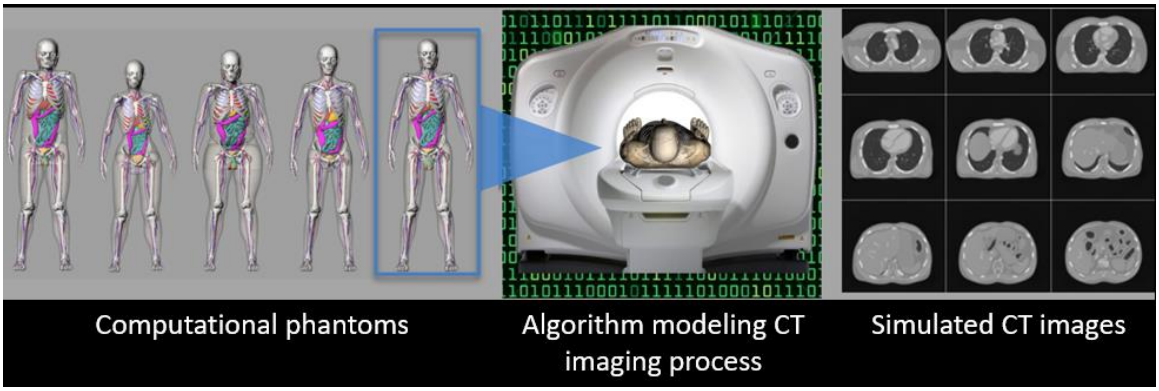

Figure 1: Medical imaging simulation (Abadi, 2020). Simulation of CT imaging is shown as an example.

For simulated studies to have merit, it is important that the computational phantoms and imaging algorithms used for them are realistic. Without this, simulated results may not be indicative of what would occur in live experiments. For phantoms, in particular, they need to be able to realistically model the variable organ anatomies of the population at large. They must also be defined at high levels of anatomical detail to evaluate high-resolution imaging methods such as photon-counting CT. Without such

detail, phantoms would not be able to evaluate the higher spatial resolution. To meet these requirements, many different phantoms have been developed over the years (Abadi, 2020)

1.2 4D Extended Cardiac-Torso (XCAT) Phantom

One of the most widely used phantoms in imaging research is the 4D extended cardiac-torso (XCAT) phantom (Segars, 2008). The XCAT includes a series of phantoms, each based on patient imaging data, encompassing both sexes and various heights and weights at different ages from newborn to adult (Figure 2). Organ shapes and structures in the phantoms are realistically defined with spline surfaces. Each phantom also includes parameterized models for the cardiac and respiratory motions. With this, the XCAT phantom can produce realistic and predictive 3D and 4D imaging data if combined with accurate imaging models (Segars, 2010). Moreover, the XCAT phantom can generate a series of anatomically varying phantoms representing the public to simulate a clinical study or trial and assist the investigation of motion effects (Segars, 2013). Despite these abilities, the XCAT phantom has some limitations. One of them is that the organs or tissues inside the phantom are modeled as homogeneous (Figure 3). In other words, each organ or tissue is represented by a single material. The phantom does not model the tissue heterogeneity within each organ. Consequently, simulated images using the XCAT phantom are less realistic, especially for high resolution imaging devices such as CT and MRI (Segars, 2018).

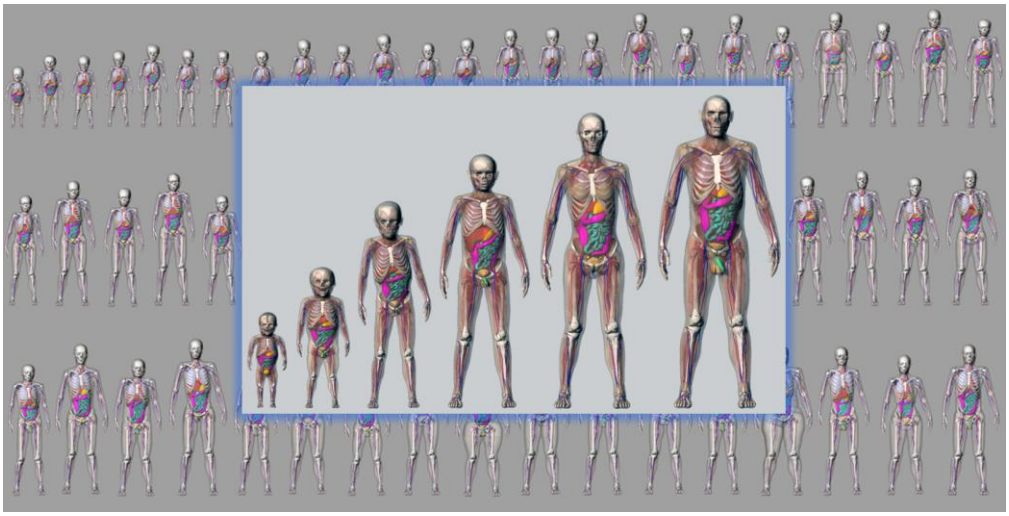

Figure 2: XCAT phantoms of varying ages and anatomies.

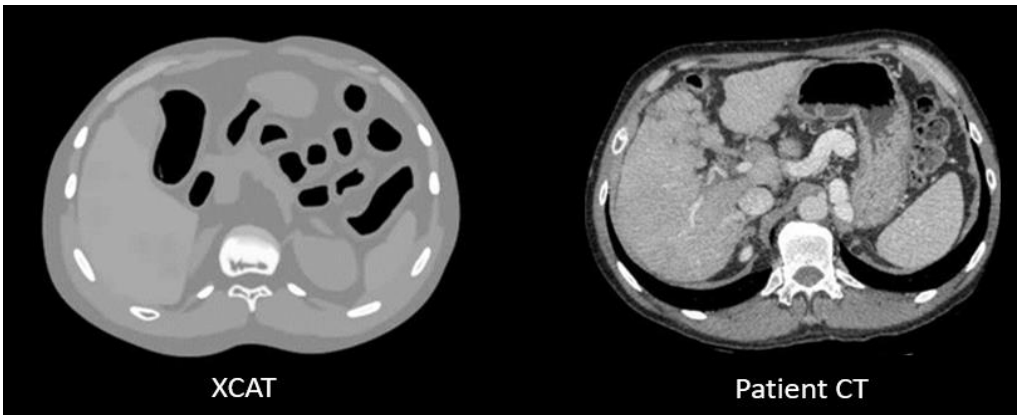

Figure 3: Simulated CT from an XCAT phantom vs patient CT. The XCAT tissues are uniform in appearance unlike the heterogenous tissues within the patient.

1.3 Prior Work in Modeling Tissue Heterogeneity

Previous work has investigated filling the XCAT organs with heterogeneous volumetric textures for more realistic simulations, specifically for CT. Focusing on the lungs, Abadi

et al (Abadi 2017) utilized a volume-filling branching algorithm to grow the higher generations of the airways and vessels to the level of terminal branches to provide more anatomical detail within the XCAT lungs. In addition, a non-parametric texture synthesis algorithm was used to simulate the heterogeneous background textures behind the airway and vessel anatomy. The enhanced lung models greatly improve the realism of the XCAT phantoms making them more applicable to virtual trials of lung imaging with CT (Figure 4).

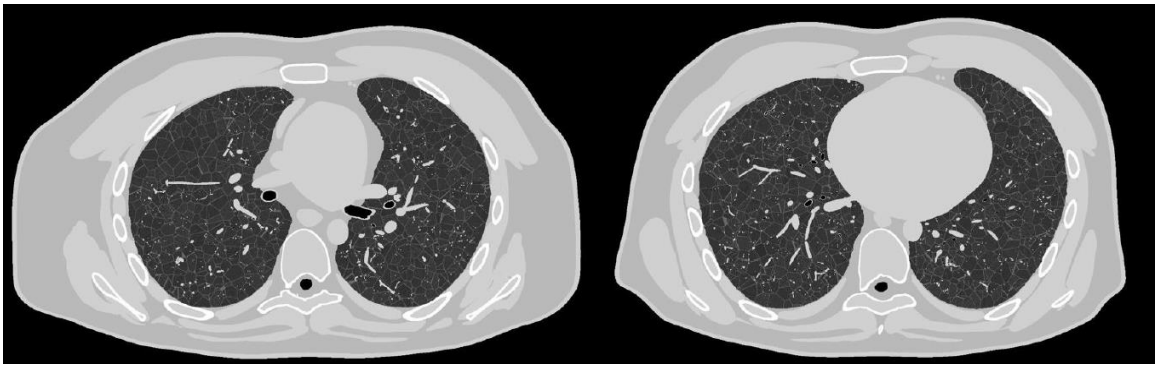

Figure 4: Example slices (left and right) of the voxelized phantom including parenchyma and non-parenchyma structures inside the lungs of an XCAT phantom from Abadi's work (Abadi, 2017)

Machine learning methods have been used in recent years with the goal of modeling tissue heterogeneity within the XCAT. In 2020, Chang et al (Chang, 2020) generated 2D textured XCAT phantom slices using a dual-discriminator conditional generative adversarial network (D-CGAN). Trained on patient CT data, the network could fill 2D slices of a given XCAT phantom with textures. A drawback to this method was that there

was no correspondence between adjacent slices. In 2021, Yuan et al (Yuan, 2021) expanded Chang's work to generate 3D textured XCAT phantoms by replacing the 2D convolutional layers with 3D layers in the model. He trained two generators separately by CGAN frameworks. One generator focused on generating the region other than lungs, another focused on generating the region in the lungs. The generated 3D phantoms (Figure 5) were found to have good continuity in 3D, especially for lung vessels traveling between slices.

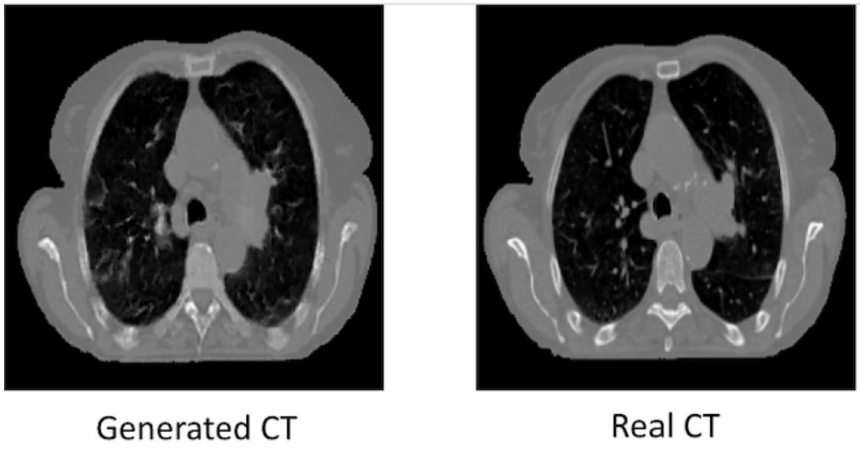

Figure 5: CGAN generated image (left) and its target image (right) (Yuan, 2021)

1.4 Generative Adversarial Networks (GAN)

1.4.1 Basic Principles

The methods above demonstrate the applicability of machine learning technology to enhance computational phantoms with organ textures. The basic idea is to train a neural network on imaging data then use it to fill organs within phantoms with volumetric

textures representing tissue heterogeneity. Generative Adversarial Networks (GAN) (Goodfellow, 2014), which is currently one of the most popular neural networks for image translations, can be an ideal model to achieve it.

The framework of a GAN is shown in Figure 6. It consists of two neural networks, a generator and a discriminator. The generator takes a random noise vector and outputs a fake sample. The discriminator takes the real sample from the training dataset and the fake sample generated by the generator and distinguishes between the real sample and the fake sample. During the training, the generator is trained to produce a fake sample that can make the discriminator output a "real" judgement while the discriminator is trained to output the right judgement. In other words, the generator and the discriminator are adversaries to each other.

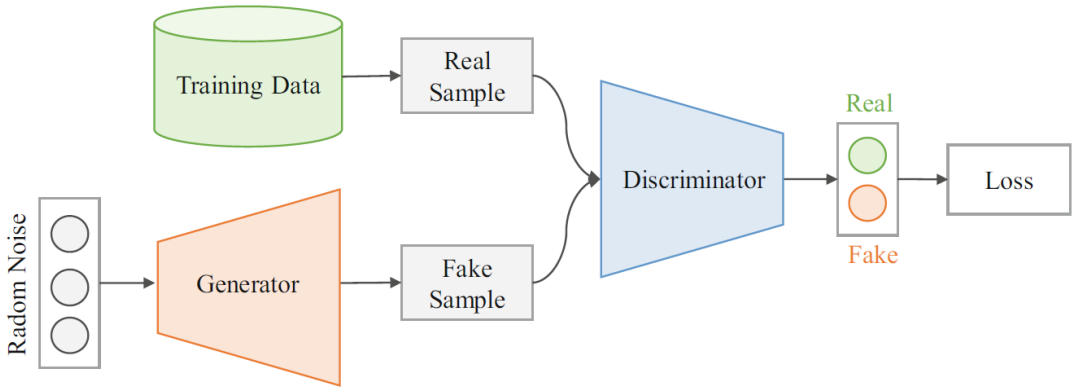

Figure 6: Framework of GAN (Mao, 2021)

Quantitatively, the recommended generator's loss function is:

$$\min_G V_{\text{GAN}}(G) = -\mathbb{E}_{z\sim p_z(z)}[\log(D(G(z)))].$$

where z is the input noise vector, $p_z(z)$ is a uniform or Gaussian distribution of z, $G(z)$ is the fake sample produced by the generator and $D(G(z))$ is the output ("real" or "fake") of the discriminator after taking the fake sample. The generator works to minimize the above loss function. The recommend loss function for the discriminator is:

$$\min_G \max_D V_{\text{GAN}}(D, G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

where x is the real sample and $p_{\text{data}}(x)$ is the dataset of real samples. The above equation clearly reveals the adversarial relationship between the generator and the discriminator. The discriminator works to maximize the right part of the above equation while the generator tries to minimize it. Note that when the generator minimizes its loss function, it reduces the discriminator's loss function as well.

Although GAN is a powerful tool for image generation, it still has some problems such as the framework is not stable, the generated image quality is low and there is a lack of appropriate evaluation metrics.

1.4.2 Conditional GAN (CGAN)

As mentioned before, our goal is to add realistic textures to the XCAT phantom, which involves the image-to-image translation. There is a widely used model for image-to-image translation called pix2pix (Isola, 2017) which utilizes the Conditional GAN (CGAN) framework to generate realistic images and keep the semantic consistency between the input and output images of the generator. The framework of CGAN is

shown in Figure 7.

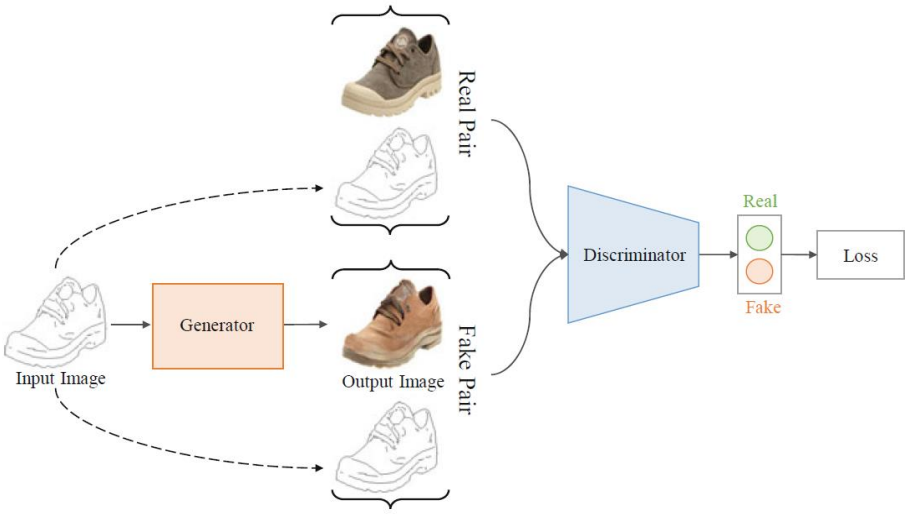

Figure 7: Framework of CGAN (Mao, 2021)

The CGAN's framework is similar to GAN's. The differences are that the generator takes a "sketch", which has a shape similar to the target image as the input instead of a random vector. The "sketch" which is the condition is combined with the real sample to compose the real pair and combined with the generated sample to compose the fake pair. The discriminator distinguishes between the real and fake pair.

Quantitatively, the loss function of the CGAN is:

$$\min_G \max_D \mathcal{L}(G, D) = \mathbb{E}_{(x,y)}[\log D(y|x)] + \mathbb{E}_{x,z}[\log(1 - D(G(z|x)|x))]$$

where x is the input image, y is the target image and z is the noise vector. Moreover, the mean absolute error (L1 loss) between the generated and target images is added to

the CGAN's loss function. The L1 loss is expressed as:

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

The reason why the above equations still use the noise vector is that the generation of images should be stochastic. However, many training trials prove that it makes no difference whether to add the noise or not. As a result, the loss function used in this thesis does not include the noise.

For the architectures, CGAN uses the U-Net (Ronneberger et al, 2015) for the generator's network and a patch-based architecture for the discriminator's network.

1.5 Goals

Our goal in this work is to add realistic textures to the XCAT phantoms (with a specific focus on the lungs) making the phantoms more realistic to investigate high-resolution CT imaging methods. For this, we utilize a CGAN trained on patient imaging data. This thesis consists of two projects. The task of the first project was to optimize the CGAN framework to generate better 3D textured XCAT lung phantoms building on the previous work by Yuan (Yuan, 2021). For Project 2, we applied what we learned from Project 1 to synthesize 3D high resolution lung background textures using an optimized CGAN trained on high-resolution micro-CT data.

2. Project 1: Optimization of Yuan's CGAN method

2.1 Dataset

We utilized the same dataset that was used in the previous work by Yuan et al (Yuan, 2021). The data consisted of 400 non-small cell lung cancer (NSCLC) CT images as well as their segmentation masks which delineated major organs as well as tumor volumes. The data is available at https://wiki.cancerimagingarchive.net/display/Public/NSCLC-Radiomics#16056854fa40518ae0634edca8cfa8e1df141bda

2.2 Data Processing

The data was processed as done previously The CT images and their segmentation masks were cropped and rescaled to 128x256x256; the CT images were also normalized. The segmentations of some image samples were missing key organs, such as the left and right lungs and heart. Those were excluded resulting in a final total of 340 sample pairs with a CT image and its corresponding organ segmentation map in each pair (Figure 8). In the organ maps, each organ was set to its own unique gray scale value.

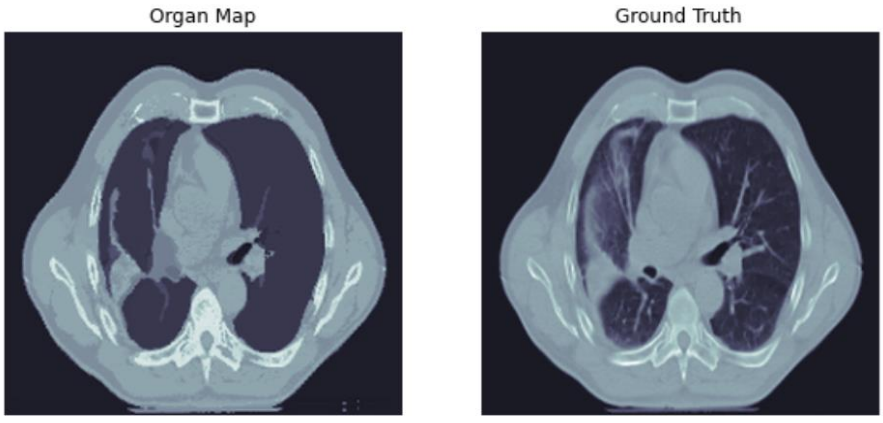

Figure 8: A sample pair with an organ map (left) and its CT image (right)

2.3 Training Methods and Loss Function

We used the same CGAN structure as the previous work to train on 330 sample pairs leaving the remaining 10 samples pairs for validation. One of the main differences between the previous work and this work is in the loss function. In the previous work, the generator loss function was a combination of binary cross-entropy and L1 loss (mean absolute error). In this work, binary cross-entropy is substituted by sigmoid cross-entropy and a new metric, the L1 loss between the real parts of the Fourier transform of the generated image and the target plus the L1 loss between the imaginary parts of the Fourier transform of the generated image and the target, is also incorporated to the generator loss function. We call this new metrics "frequency L1 loss" or "L1 in frequency space" in this thesis. We added the frequency L1 loss because the finer structures in the image correspond to higher frequencies in the frequency space in theory and minimizing frequency L1 loss during the training may produce a better generator. The loss functions of the generator and the discriminator are:

L_G = -[log(sigmoid(fake_logit)) + log(1-sigmoid(real_logit))] + L1 + (L1 in frequency space)

L_D = -[log(sigmoid(real_logit)) + log(1-sigmoid(fake_logit))]

fake_logit = (disc_generated_output - mean(disc_real_output))

real_logit = (disc_real_output - mean(disc_generated_output))

Sigmoid(x) = 1 / (1 + exp(x))

where disc_generated_output is the discriminator's output for the generated image and organ map pair (fake pair) and disc_real_output is the discriminator's output for the CT image and organ map pair (real pair).

We used a 3D U-net to build the generator and a 3D convolutional neural network to build the discriminator (Figures 9 and 10).

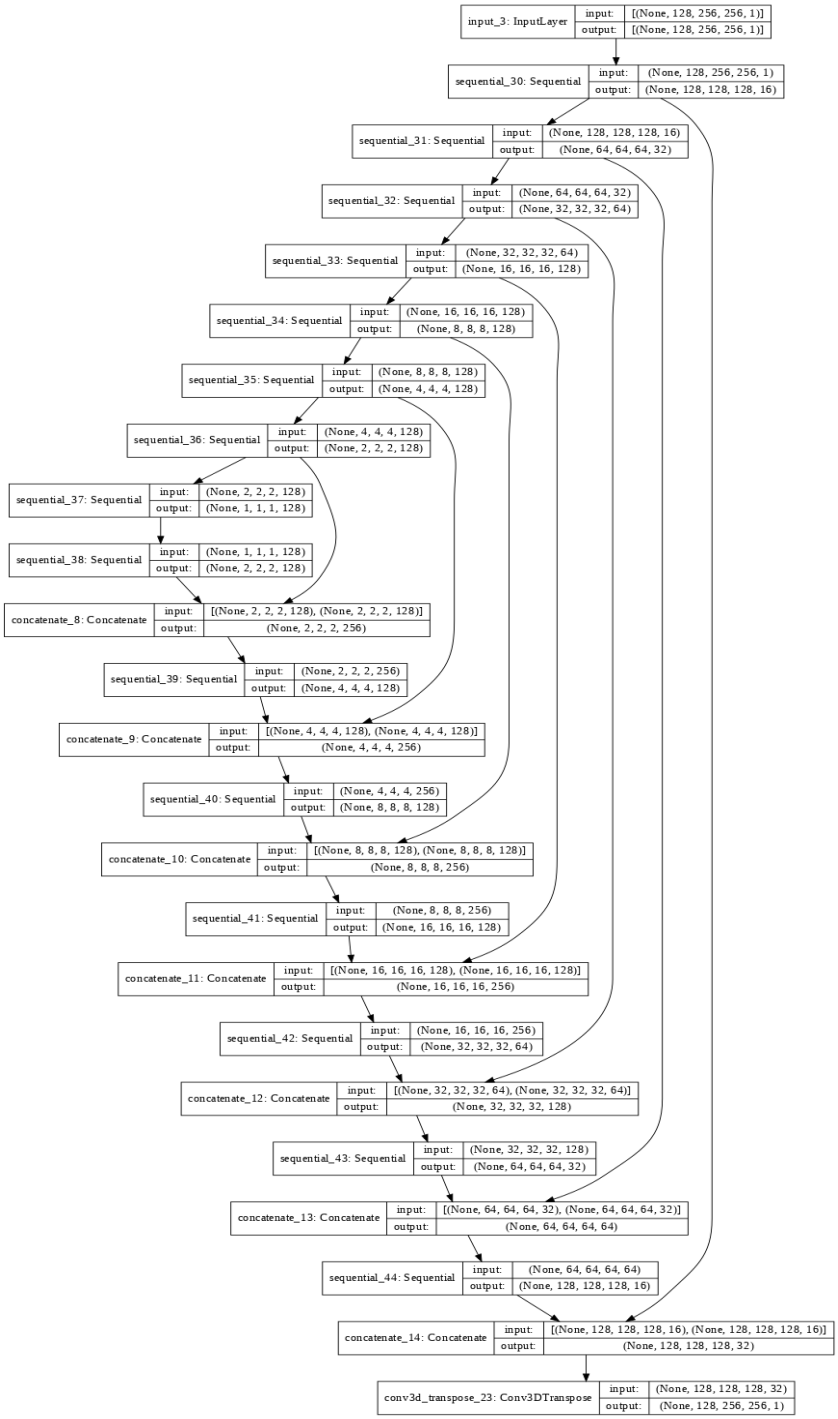

Figure 9: Generator structure

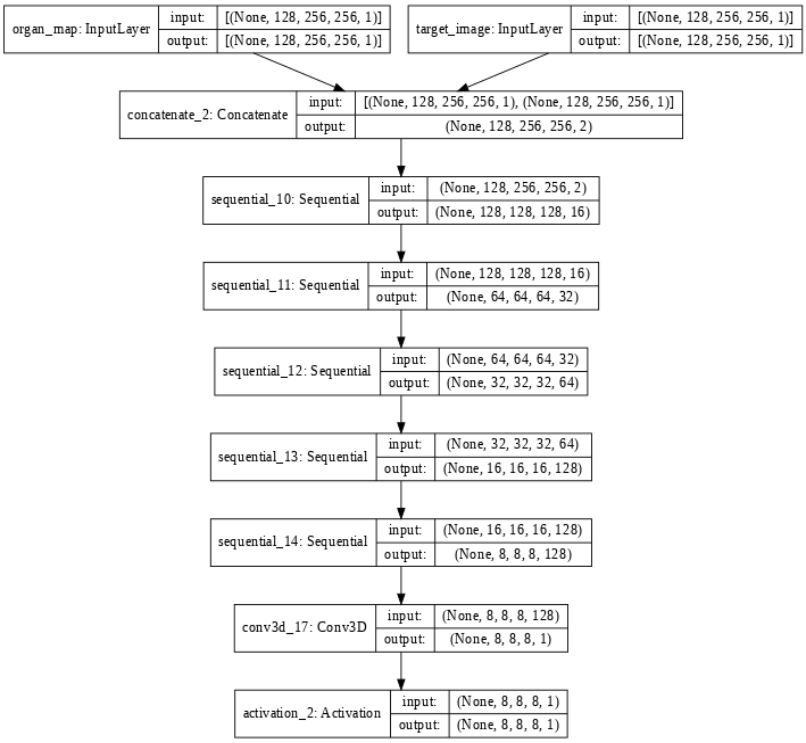

Figure 10: Discriminator structure

2.4 Repeat of the Previous Work

We repeated the previous work of Yuan et al to see if we could get similar results independently before trying to optimize the method. Specifically, we utilized binary cross-entropy plus L1 loss for the generator loss function and binary cross-entropy for the discriminator loss function in the CGAN framework. We basically completed two training trials: one utilizing Yuan's 2-generator method (mentioned in chapter 1) and, another training 1 generator to generate the whole phantom.

2.5 Evaluation Methods

With the patient organ maps as input, the trained generator was used to fill them with textures generating fake CT images. The structural similarity index (SSIM), peak signal-to-noise ratio (PSNR), and L1 and L2 loss metrics were used to evaluate the generated images. The trained generator was also used to define textures within a 3D XCAT phantom to test the generator's performance on an unknown target. The XCAT phantom for a standard male adult was voxelized into a 3D organ map image setting the organs and structures to the gray scale values defined in the patient organ maps. The XCAT organ map was input to the generator which filled the structures with textures.

2.6 Results

Figure 11 shows an example of a generated fake CT image and its corresponding original CT. Images generated in this fashion were found to be blurred and did not contain much detail in the lungs as can be seen in the example.

Figure 12 shows a comparison of the original XCAT adult male phantom and its textured version using the generator. The generated textures within the XCAT were found to be similarly blurry and lacking detail.

Figure 13 shows the Loss vs epochs and SSIM vs epochs curves of the training for our CGAN framework with sigmoid cross-entropy. As can be seen in the figure, the CGAN framework is very stable. For comparison, Figure 14 shows an example of loss curves for

the generator in a CGAN framework with binary cross-entropy. In this case, the model starts to collapse at about the 100th epoch.

During the training, our generator's sigmoid cross-entropy (gen_gan_loss) and discriminator's sigmoid cross-entropy (disc_loss) vary at the first few epochs but converge to the same value quickly. Hence, the framework is mainly minimizing the frequency L1 loss (the L1 loss is very small thus not shown in Figure 13).

Table 1 shows the quantitative evaluation results in the validation group for the previous work by Yuan, repeated results and this work. The generated images using our method were found to be better in SSIM than the previous work, but slightly worse in measurements for PSNR and L1 and L2 losses. For our repeat of the previous work, our results were also found to be worse since the SSIM, PSNR, L1 and L2 values were worse than the data in the previous work.

For SSIM, our framework can improve SSIM to over 96% in the training dataset and about 91% in the validation dataset. We randomly chose 9 organ maps to test the generator, the average SSIM was 92.28% and standard deviation was 0.0152 as shown in Table 1.

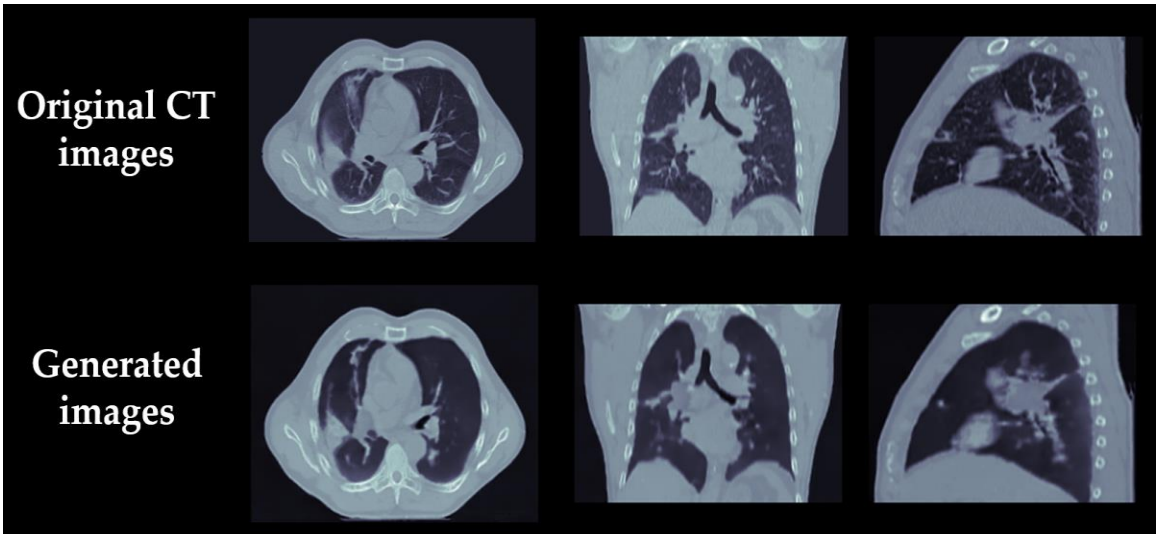

Figure 11: Original CT image versus generated image (left: transverse plane, middle: coronal plane, right: sagittal plane).

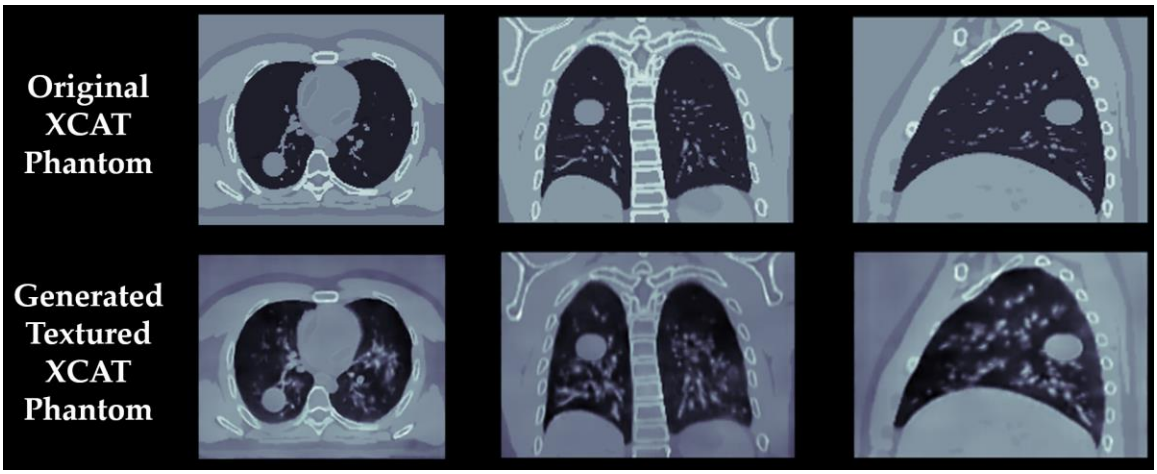

Figure 12: Original XCAT phantom versus a generated textured 3D XCAT phantom (left: transverse plane, middle: coronal plane, right: sagittal plane)

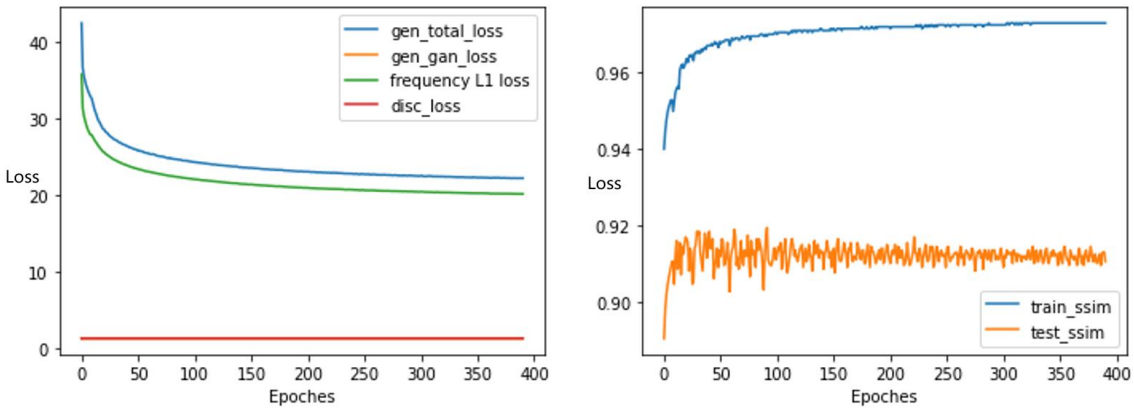

Figure 13: The Loss vs epochs (left) and SSIM vs epochs (right) curves of the training

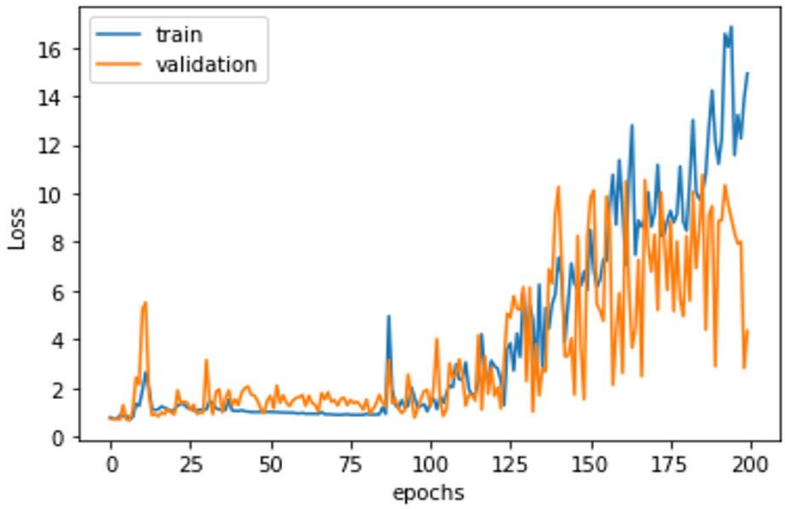

Figure 14: An example of loss curves for the generator in a CGAN framework with binary cross-entropy.

Table 1: Quantitative evaluation results.

	SSIM	PSNR	L1	L2
Previous work	0.8793 ± 0.0266	28.35 ± 1.11	0.015 ± 0.003	0.0015 ± 0.0004
Repeated previous work (2-generator)	0.8349 ± 0.0669	22.80 ± 5.88	0.078 ± 0.043	0.0099 ± 0.0083
Repeated previous work (1-generator)	0.8744 ± 0.0467	23.10 ± 5.90	0.074 ± 0.038	0.0089 ± 0.0070
This work	0.9228 ± 0.0152	25.29 ± 2.32	0.049 ± 0.011	0.0034 ± 0.0018

2.7 Discussion

Qualitatively, for structures other than the lungs, such as the ribs, spine and muscles, the trained generator was found to model them well. For the lungs, the vessels and airways were found to be blurry lacking the necessary detail.

Our CGAN model was found to be stable and produced a better result than the previous method of Yuan with respect to SSIM. However, the result's quality was not improved when evaluated by other metrics such as PSNR, L1 and L2. Moreover, comparing Figure 5 with Figure 11, it can be seen that the new model's ability to generate detailed lung textures is poor. The main reason is that the 3D textures' variability is very high. On the other hand, the number of generator variables is not enough compared to the high

variability of the textures. In addition, the number of samples (340) is not enough for training and validation. The new metric, frequency L1 loss, was incorporated to tackle the problem mentioned above. However, the model could not minimize the frequency L1 loss to an ideal value.

2.8 Conclusion

For the project, replacing the binary cross-entropy with sigmoid cross-entropy and bringing in frequency L1 loss in the generator loss function made the training more stable and slightly improved the results with respect to SSIM. However, the resulting PSNR, L1 and L2 were worse than Yuan's previous work. Consequently, we utilize the loss functions of this prior method for the second project.

3. Project 2: Generate High-resolution Lung Textures

3.1 Dataset

3D micro-CT data of the whole left and right lungs (Figure 15) was obtained from the Ravin Advanced Imaging Laboratories at Duke University for the modeling of lung background textures. The data had an array size of 1550x1090 and consisted of 2128 slices. The resolution was 0.11 mm isotropic.

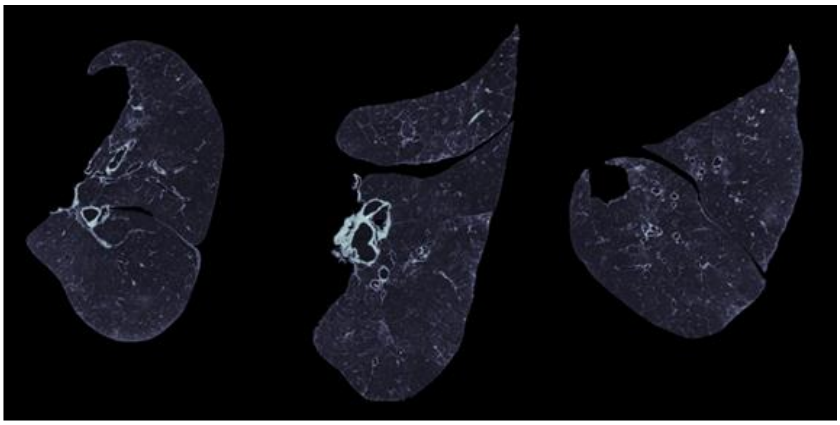

Figure 15: Lung 3D micro-CT data (left: transverse plane, middle: coronal plane, right: sagittal plane).

3.2 Data Processing

The micro-CT data was previously segmented labeling the different tissues with integer IDs ranging from 0-27. Table 2 shows the values assigned to different lung tissues in the data.

Table 2: Values of different lung structures in the 3D micro CT image.

	Non-lung	Air	Parenchyma	Cartilage	Calcification
Pixel values	0	1	2-14	15-22	23-27

For this project, the generator will be trained to generate background textures within a given lung segmented mask image. It will not be used to generate vessels or airways; those structures will already exist. Therefore, the micro-CT data was processed to remove vessels and airways. Pixel values for the parenchyma were kept while the others were set to 0. Remaining small vessels and vasculature were then eliminated. This was done by a MATLAB function: "bwconncomp". We utilized this function to find the connected components (lines) in the image and then remove the components with large volumes while keeping the small ones. The detailed information can be found at: https://www.mathworks.com/help/images/ref/bwconncomp.html.

The processed micro-CT data was rescaled into a 2128x1536x1024 array to divide it into 1,000 sample pairs with a size of 16x128x128. The size of the samples was chosen factoring in available computer memory. Each sample pair consisted of 1 target image and its corresponding mask. Target images contain the integer intensities for the lung background tissue in the patch while the mask image is binary setting all the lung background tissue to 1 and everything else to 0. Figure 16 shows an example sample pair from the dataset.

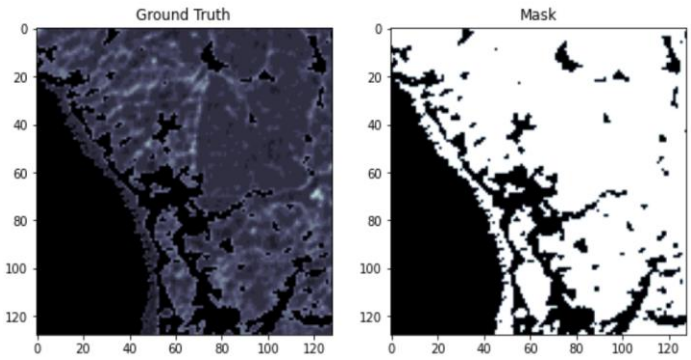

Figure 16: One sample pair from the dataset (left: target image, right: mask image).

3.3 Training Methods

We used a CGAN framework like the one in Project 1 to train on the micro-CT datasets. The size of the micro-CT samples is 16x128x128, which is smaller than the size of the samples in Project 1. As a result, the number of layers in the generator and the discriminator will be less than those in Project 1 if we use the same CGAN framework. If we just use the 3D U-net for the generator, the generator will be weakened more than the discriminator so the generator will not be well trained. In this case, there are basically two methods to resolve this problem. One is to weaken the discriminator, another is to strengthen the generator. To make the generator produce realistic images, the discriminator should not be weakened. Hence, we strengthen the generator based on the 3D U-net in Project 1. The 3D U-net in Project 1 took a 128x256x256 image and output an image with the same size. In this project, we first up-sample the input to 16x256x256, then pass it through the 3D U-net. After that, the U-net's output is down-sampled to the

input's size. Moreover, the discriminator is also strengthened by up-sampling the input to 16x256x256 at the beginning while keeping the output size 8x8x8.

With this modified framework, 800 samples were used for training and 200 samples were used for validation. To give our method the ability to generate random textures to simulate anatomical variability, several Dropout layers were added to the trained generator. The Dropout layer originally serves to randomly set input units to 0 with a specified frequency at each step during training time, which helps prevent overfitting. However, in practice, we do not add Dropout layers during training time since the CGAN framework is unstable with Dropout layers. Since the Dropout layer does not have trainable variables, we can add Dropout layers to the generator after training to make it randomly produce textures. The generator's output size is also 16x128x128.

3.4 Evaluation Methods

In addition to the metrics used in Project 1, we evaluated the distributions of entropy, mean, standard deviation and correlation in 1000 generated image pieces and their original target images which provide the ground truth. The correlation was calculated based on: https://ww2.mathworks.cn/help/images/derive-statistics-from-glcm-and-plot-correlation.html. We also used the trained generator to generate lung background textures within the standard male adult XCAT phantom and compared them to the

original micro-CT data as well as the previous work of Abadi et al (Abadi, 2017). Since the size of the mask or phantom is much bigger than the generator's output size, the textures on the mask or phantom are not generated once. We broke the mask or phantom to several 16x128x128 patches and filled in textures on each patch.

3.5 Results

Figure 17 shows an example of a generated image compared to its ground truth (original segmented micro-CT) as well as the mask used to generate the texture. The generated image is visually similar to the original micro-CT.

Using Dropout layers, the generator was capable of producing variable textures. Table 3 shows the 100xL1 values between different generated images with the same input by the generator with different numbers of Dropout layers. As indicated by the table, the numbers of Dropout layers can affect the difference between generated images with the same input. More specifically, the L1 is smaller with less Dropout layers, which means the generated images are closer to each other with less Dropout layers. For the same number of Dropout layers, the difference between generated images with the same input are close.

Figure 18 shows the generated results by the model with 6 Dropout layers. Table 4 shows the quantitative evaluation results with different number of Dropout layers. Table 4 shows that the number of Dropout layers does not substantially affect the similarity

between the ground truth and the generated image. The SSIM and PSNR are generally growing with less Dropout layers, which indicates that the generated image has a tendency to be closer to the ground truth with less Dropout layers.

In Tables 3 and 4, the Dropout layers are in the 7th down-sample module and 1st up-sample module for the 2 Dropout layers case. They are in the 6th, 7th down-sample modules and the 1st, 2nd up-sample modules for the 4 Dropout layer case, and they are in the 5th, 6th, 7th down-sample modules and the 1st, 2nd, 3rd up-sample modules for the 6 Dropout layer case. The conclusions above may be less apparent with different placement of the Dropout layers.

Figures 19 and 20 show the background textures generated for the whole left and right lungs. Some faint grids appear in the generated whole lung images because they consist of thousands of 16x128x128 pieces produced by the generator that are stitched together. Hence, some discontinuities appear on the boundaries between generated pieces.

Figures 21-28 show the entropy, mean, standard deviation, and correlation distribution and histogram respectively for 1000 sample pairs. Quantitatively, the generated results compare very favorably with the ground truth of the micro-CT data.

Figure 29 shows an XCAT phantom textured using the method of Abadi et al (Abadi, 2017) and the same phantom textured using the trained generator (the parenchyma region in the original phantom were replaced with the generated textures). Figure 30

compares the detailed textures between the original phantom and the generated one among several slices. Comparing the background textures in Figure 30 with that in Figures 17 and 18, the original phantom can be seen to have better resolution and more higher intensities (brighter spots) while the generated phantom is more similar to the real micro-CT data. The reason why the generated phantom has fewer high intensities is that some big numbers (8-14), brighter speckles, were eliminated while processing the micro-CT data to remove the pulmonary vessels and airways.

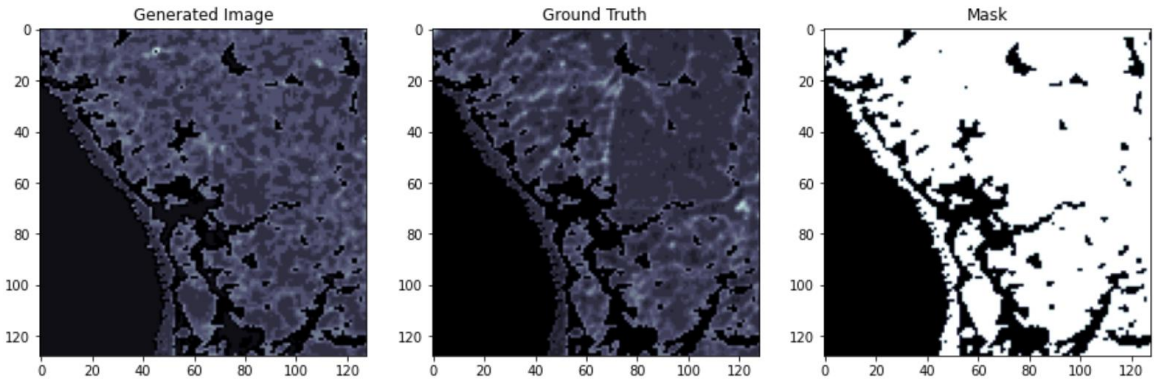

Figure 17: Generated image, ground truth and mask

Table 3: The 100xL1 values between different generated images (gen, gen1, gen2) with the same input by the generator with different numbers of Dropout layers

Numbers of Dropout layers	100x|gen-gen1|	100x|gen1-gen2|	100x|gen2-gen|
0	0	0	0
2	1.7213	1.7745	1.7670
4	2.7533	2.8406	2.8235
6	3.1930	3.2919	3.2892

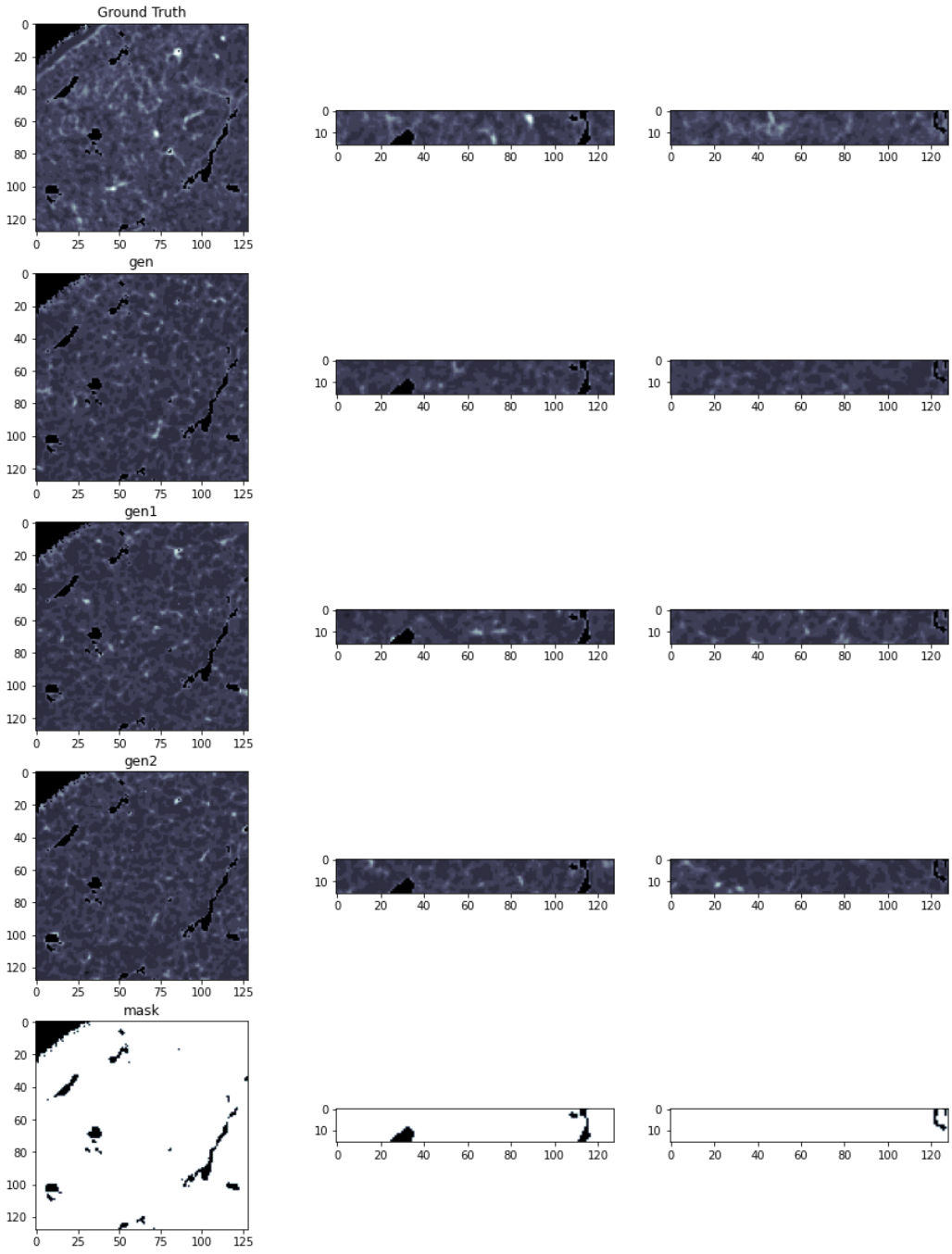

Figure 18: The generated results by the model with 6 Dropout layers (left: transverse plane, middle: frontal plane, right: sagittal plane; gen, gen1 and gen2 are 3 different generated image pieces with the same mask)

Table 4: Quantitative evaluation results with different number of Dropout layers.

Number of Dropout layers	SSIM	PSNR	L1	L2
0	0.7000 ± 0.2712	27.63 ± 8.53	0.031 ± 0.026	0.0045 ± 0.0038
2	0.7011 ± 0.2700	27.72 ± 8.62	0.031 ± 0.026	0.0045 ± 0.0038
4	0.6972 ± 0.2717	27.79 ± 8.97	0.032 ± 0.027	0.0046 ± 0.0039
6	0.6961 ±0.2722	26.74 ± 7.32	0.032 ± 0.027	0.0048 ± 0.0040

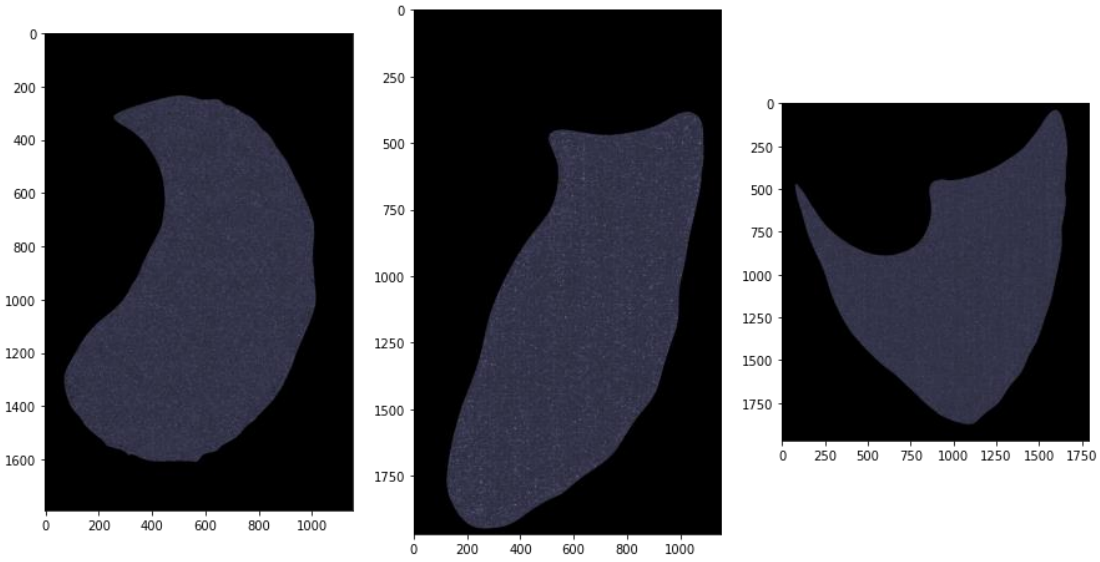

Figure 19: The generated whole left lung (1968x1792x1152) with only background textures (left: transverse plane, middle: frontal plane, right: sagittal plane)

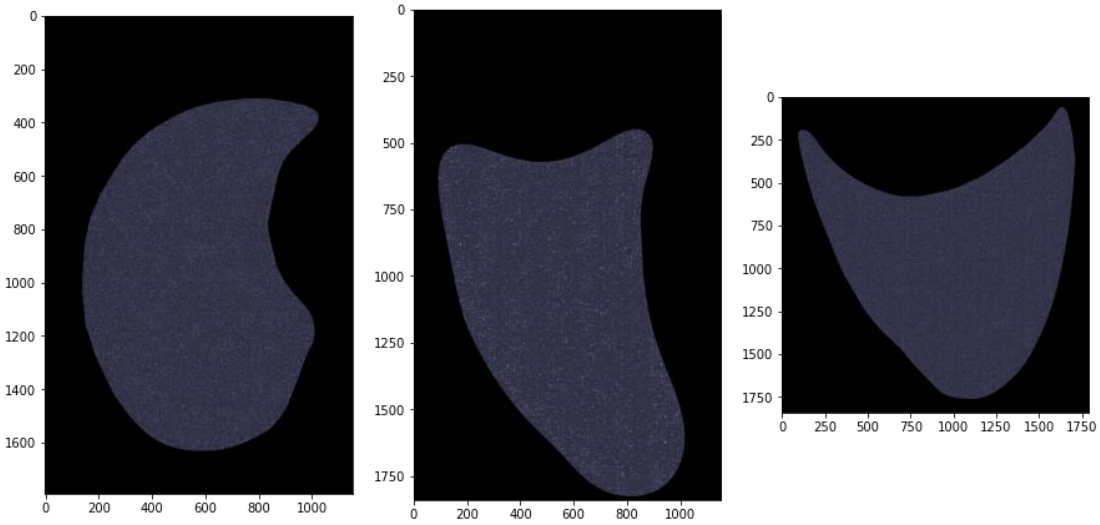

Figure 20: The generated whole right lung (1840x1792x1152) with only background textures (left: transverse plane, middle: frontal plane, right: sagittal plane)

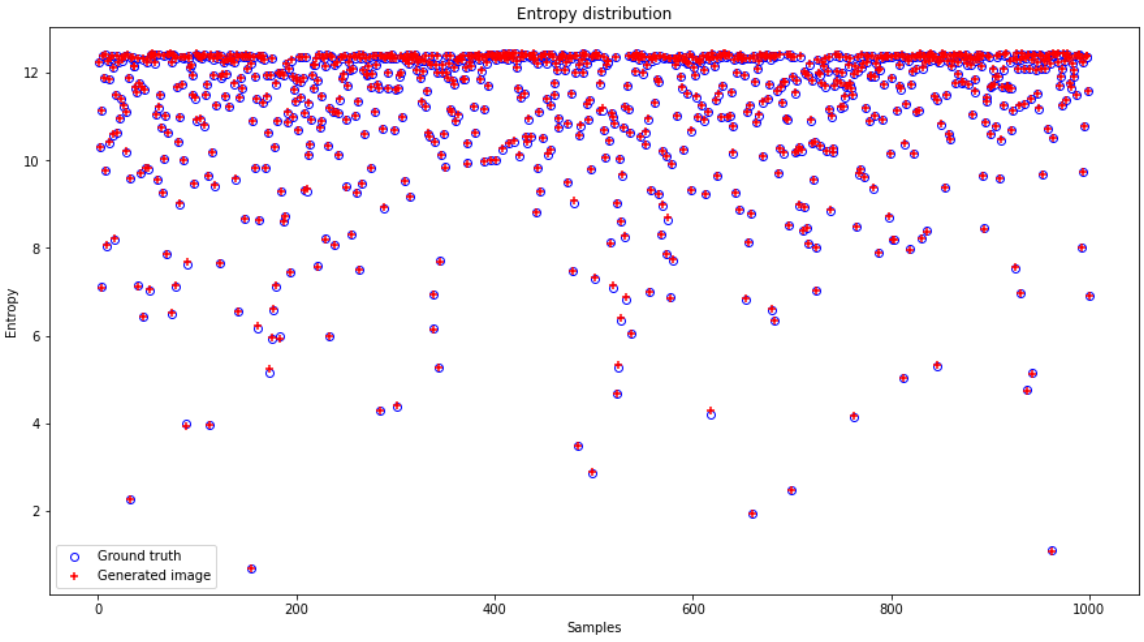

Figure 21: The entropy distribution for 1000 sample pairs

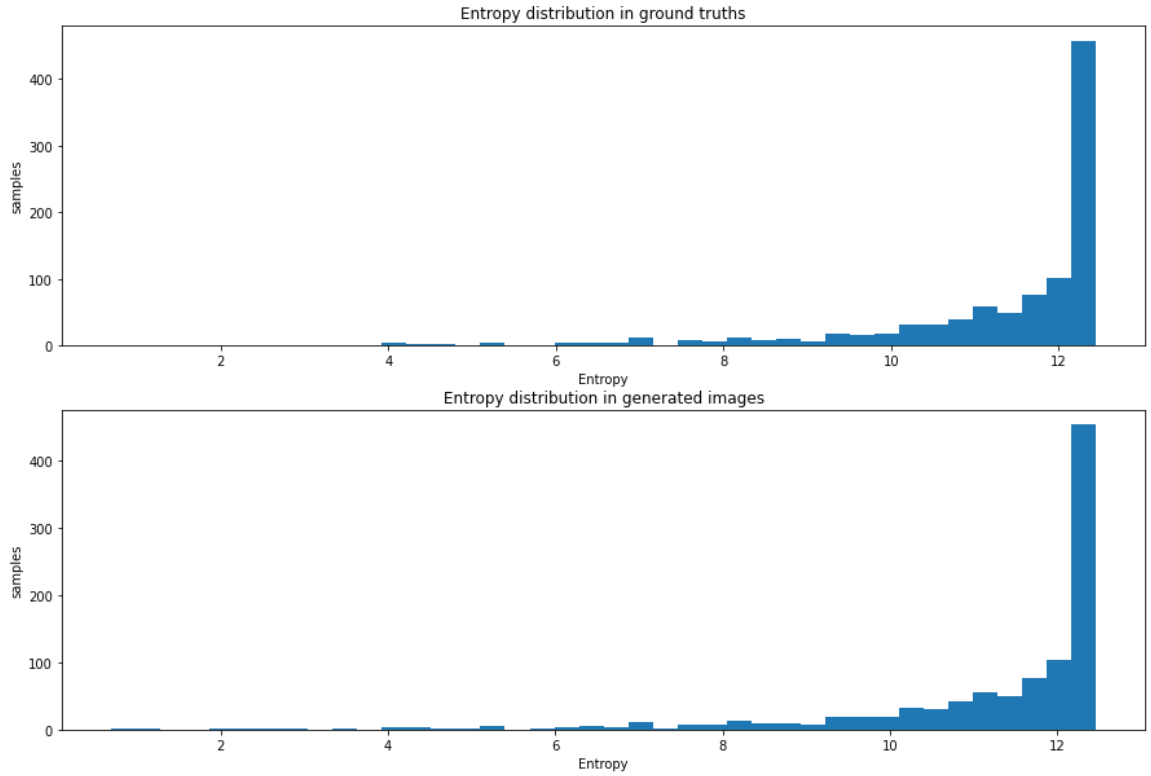

Figure 22: The entropy histogram for 1000 sample pairs

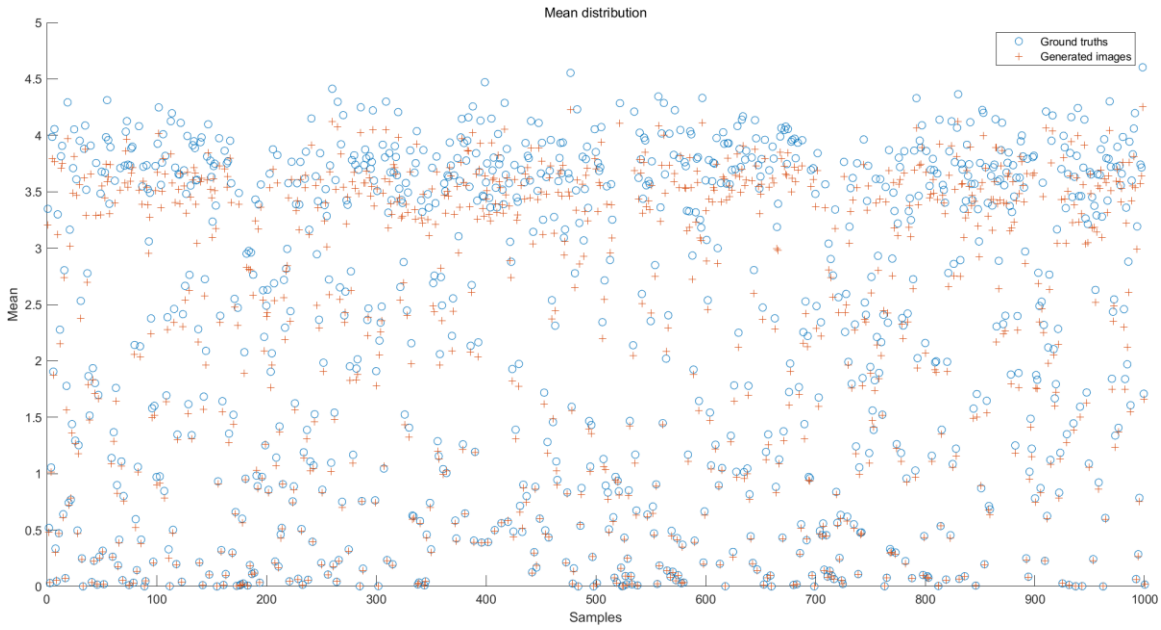

Figure 23: The mean distribution for 1000 sample pairs

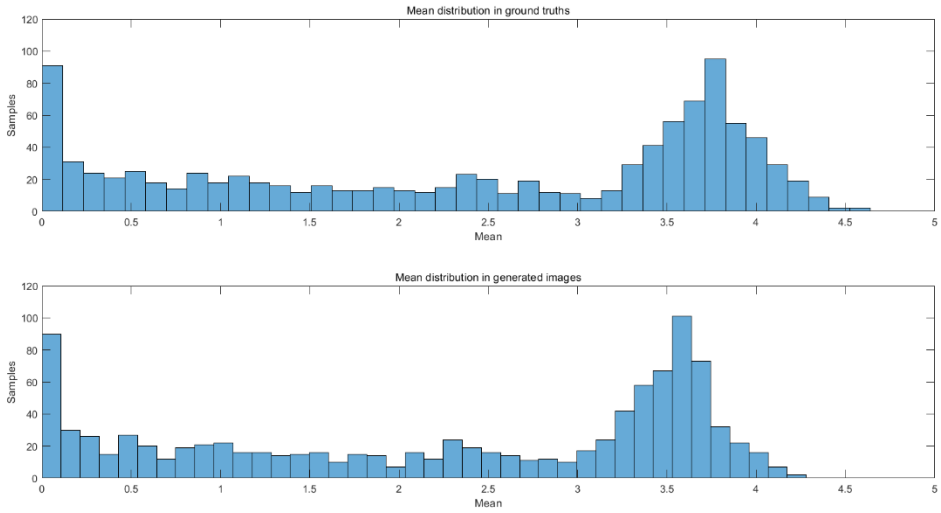

Figure 24: The mean histogram for 1000 sample pairs

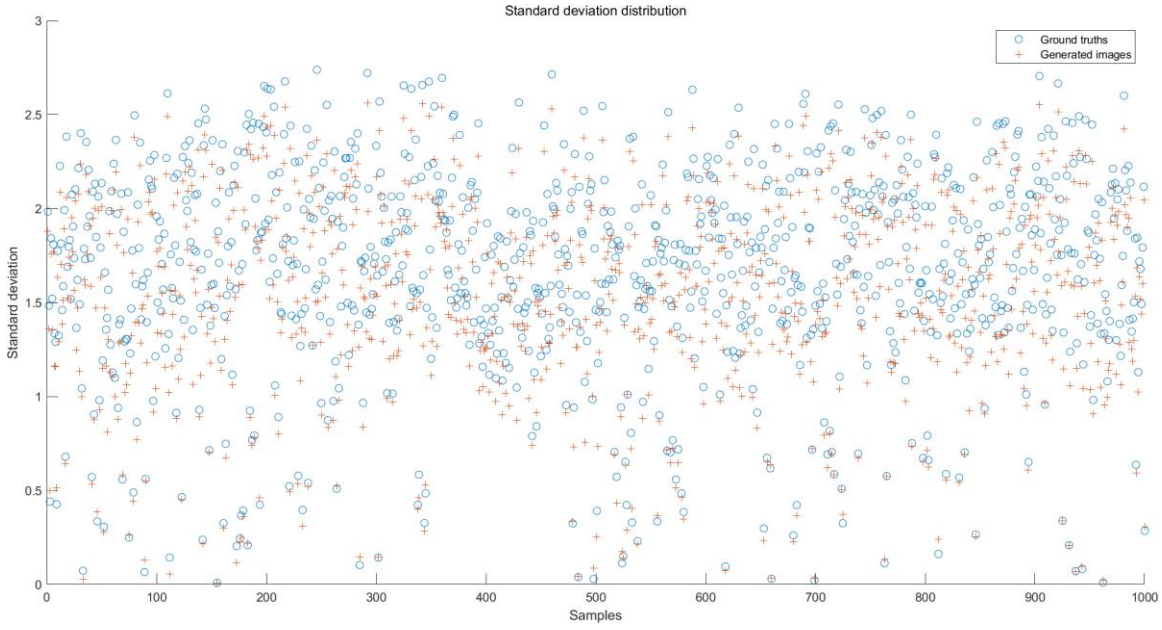

Figure 25: The standard deviation distribution for 1000 sample pairs

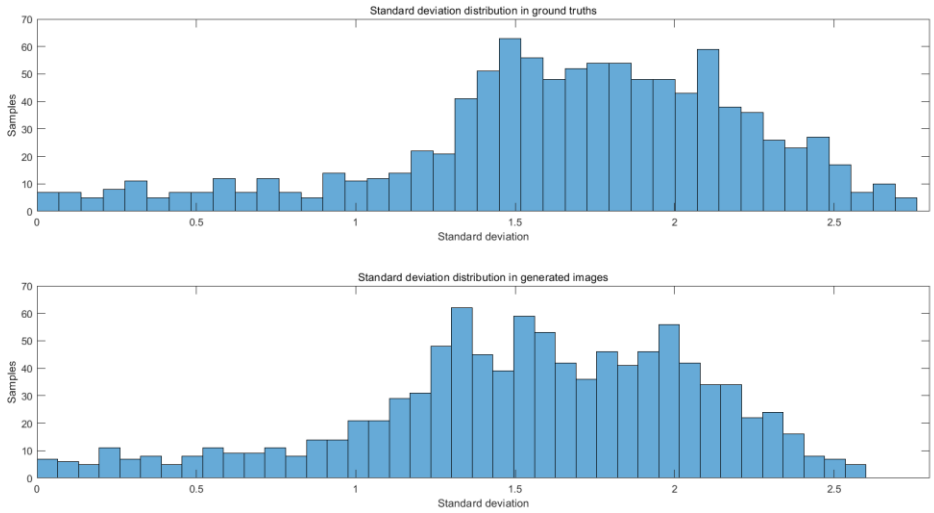

Figure 26: The standard deviation histogram for 1000 sample pairs

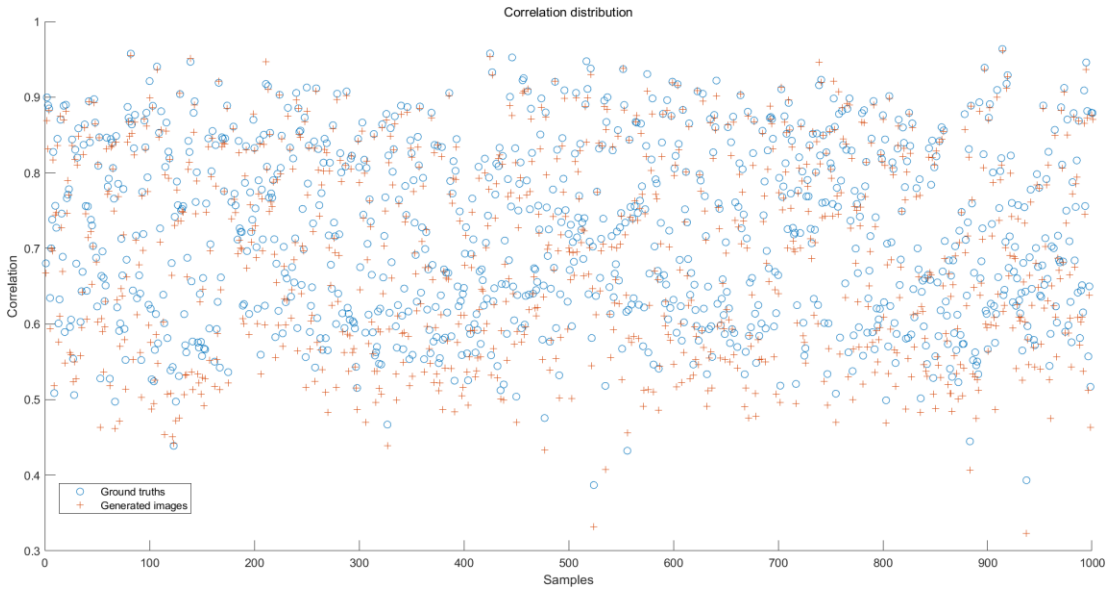

Figure 27: The correlation distribution for 1000 sample pairs

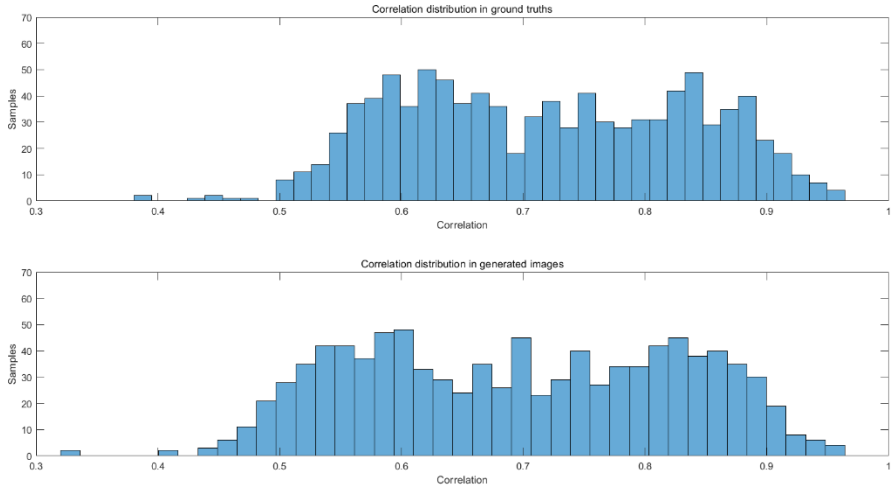

Figure 28: The correlation histogram for 1000 sample pairs

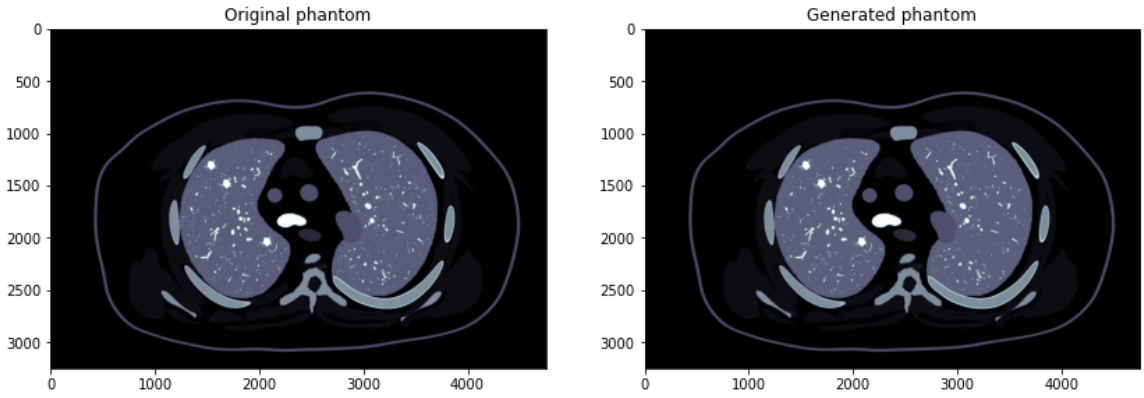

Figure 29: Original XCAT phantom textured using the method of Abadi (Left) and the same phantom with a generated texture from the CGAN framework (Right).

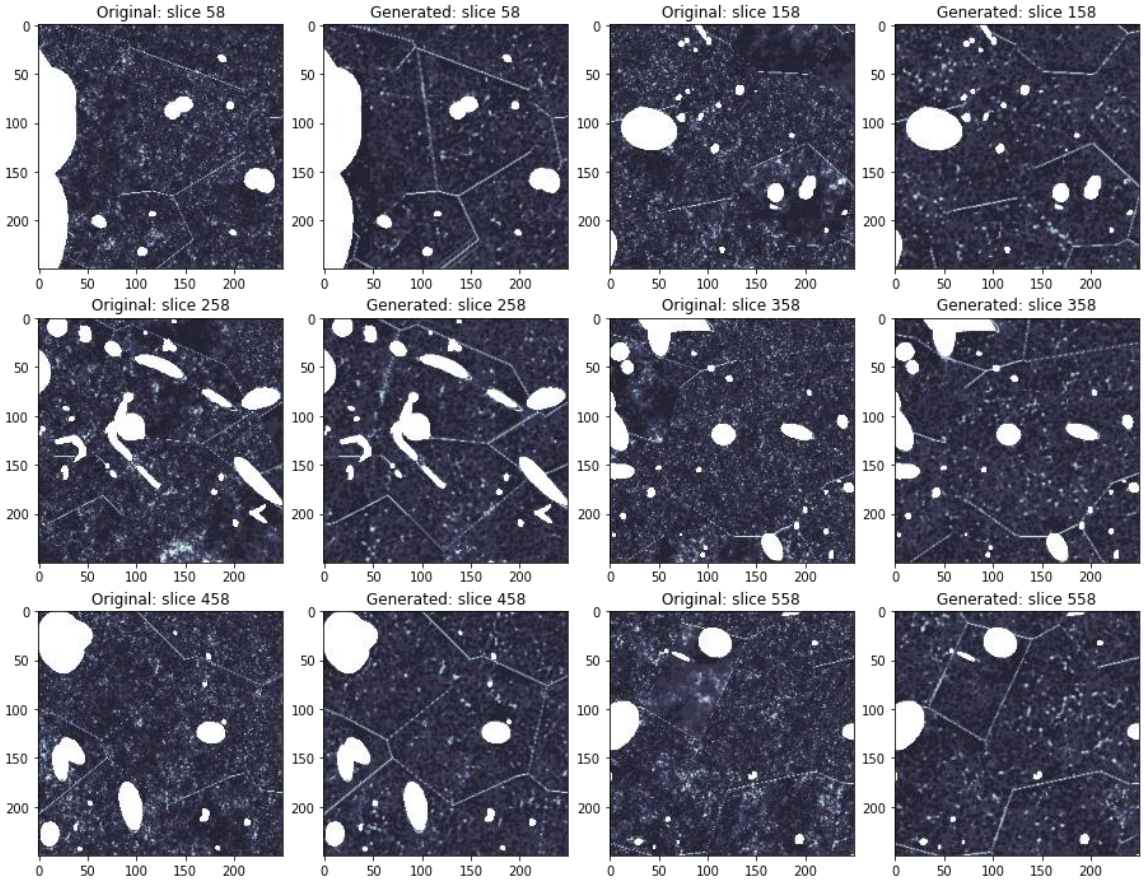

Figure 30: The detailed textures of the original Abadi phantom and the generated phantom using our method among several slices.

3.6 Discussion

The CGAN framework was found to produce textures visually and quantitatively similar to the ground truth of the micro-CT. Moreover, the generator can also randomly produce textures with Dropout layers. If we add Dropout layers in the trained generator, the generator can randomly generate textures on the mask. In other words, if we input the same mask to the generator multiple times, the output will be different each time

resulting in anatomically variable textures.

Assessing the texture generation in the whole lungs and in the XCAT, the quality of the textures was not perfect. Since the generator's output size was small, the generated whole lung is composed of thousands of small pieces. Consequently, there are some faint grids caused by discontinuities on the boundary. To alleviate this, future work will investigate training the generator to produce textures that are 256x256x256 with a resolution of 0.1 mm. This roughly corresponds to the size of a large pulmonary lobule (25x25x25 mm). Instead of filling an entire lung, the generator will fill it lobule by lobule minimizing any grid patterns. Discontinuities will also be hidden by the walls of the lobules.

The generated textures also did not contain as many high intensities as the original micro-CT data. This was due to overprocessing the images to remove vessels and airways. More careful processing will be done to include these higher intensity speckles.

3.7 Conclusion

Building upon Project 1, we developed a CGAN method, based on micro-CT data, capable of generating 3D realistic, high-resolution lung textures. These textures can be added to computational phantoms such as the XCAT to greatly improve their level of detail and realism. With this addition, the lungs no longer appear as a uniform object; they appear heterogeneous and have detail down to the resolution of micro-CT. Imaging

simulations of the phantoms will be more realistic enabling them to assess image quality and to evaluate emerging, higher-resolution imaging methods such as photon-counting CT.

References

1. Abadi, E., Segars, W. P., Tsui, B., Kinahan, P. E., Bottenus, N., Frangi, A. F., Maidment, A., Lo, J., & Samei, E. (2020). *Virtual clinical trials in medical imaging: a review*. *Journal of medical imaging (Bellingham,Wash.)*, *7*(4), 042805. https://doi.org/10.1117/1.JMI.7.4.042805

2. Segars, W.P., Mahesh, M., Beck, T.J., Frey, E.C. and Tsui, B.M.W. (2008), *Realistic CT simulation using the 4D XCAT phantom*. Med. Phys., 35: 3800-3808.

3. Segars, W.P., Sturgeon, G., Mendonca, S., Grimes, J. and Tsui, B.M.W. (2010), *4D XCAT phantom for multimodality imaging research*. Med. Phys., 37: 4902-4915.

4. Segars, W.P., Bond, J., Frush, J., Hon, S., Eckersley, C., Williams, C.H., Feng, J., Tward, D.J., Ratnanather, J.T., Miller, M.I., Frush, D. and Samei, E. (2013), *Population of anatomically variable 4D XCAT adult phantoms for imaging research and optimization*. Med. Phys., 40: 043701.

5. Segars, W.P., Tsui, B.M.W., Cai, J., Yin, F.-F., Fung, G.S.K. and Samei, E., *Application of the 4-D XCAT Phantoms in Biomedical Imaging and Beyond*, in IEEE Transactions on Medical Imaging, vol. 37, no. 3, pp. 680-692, March 2018, doi: 10.1109/TMI.2017.2738448.

6. Ehsan Abadi, Gregory M. Sturgeon, Greeshma Agasthya, Brian Harrawood, Christoph Hoeschen, Anuj Kapadia, W. Paul Segars, Ehsan Samei, *Airways, vasculature, and interstitial tissue: anatomically informed computational modeling of human lungs for virtual clinical trials*, Proc. SPIE 10132, Medical Imaging 2017: Physics of Medical Imaging, 101321Q (9 March 2017); doi:10.1117/12.2254739

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

8. Yuan, Y. *Synthesize 3D realistic CT textures and anatomy in the XCAT phantom using Generative Adversarial Network (GAN)*. Thesis of the Master of Science in Graduate Program in Medical Physics in the Graduate School of Duke University (2021).

9. Goodfellow, I.J., et al. *Generative adversarial networks*. arXiv preprint arXiv:1406.2661 (2014).

10. Mao, Xudong, Li, Qing. *Generative Adversarial Networks for Image Generation*. Singapore: Springer Singapore: Imprint: Springer, 2021.

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

12. Ronneberger, O., Fischer, P., Brox, T. (2015) *U-Net: convolutional networks for biomedical image segmentation*. In: Medical image computing and computer-assisted intervention (MICCAI), pp 234–241