University of Toronto

The Investigation of Generative Adversarial Networks for the Generation of Synthetic Computer Tomography Data from Magnetic Resonance Imaging Images for Applications in Adaptive Radiation Therapy for the Central Nervous System

by

F.A. Loo

A thesis submitted to the Faculty of Applied Science and Engineering in partial fulfillment of the requirements of the degree of Bachelor of Applied Science

in

Industrial Engineering

under the guidance of

Dr. Teodor Stanescu

Toronto, Ontario
April 2023

ABSTRACT

This thesis addresses the problem of accurately delivering radiation to the central nervous system while minimizing exposure to healthy organs in cancer radiation therapy (RT). Currently, computed tomography (CT) lacks soft-tissue contrast in the brain and may expose patients to additional radiation, whereas magnetic resonance imaging (MRI) does not contain information for RT treatment dose calculation. To address this, we propose a solution using cycle-consistent GANs (CycleGANs) to synthesize synthetic CT (SynCT) images from MRI scans. We assess nine CycleGAN models with varying parameters using ten image similarity metrics, explore model training trends across all anatomical locations of a validation dataset, and identify the evaluation metrics most sensitive to input and model changes. To enhance model performance, we generated an image mask from segmented MR and CT image sets to indicate anatomical overlap for improved feature extraction.

Our findings suggest that Model 7 with GAN + L_1 regularization and ResNet_3 generator structure, trained with 256 x 256 resolution patches instead of full images, is optimal. We also observe consistent model performance across all anatomical locations within a validation dataset with minor variation in extremities towards the z-direction, which is inconsequential in RT planning. Lastly, Spatial Correlation Coefficient (SCC), Mean Squared Error (MSE), and Relative Global Error (ERGAS) are identified as the most sensitive metrics for evaluating model performance, with Mean Absolute Error (MAE) and Root-Mean-Square Error (RMSE) also acceptable.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to Dr. Teodor Stanescu for his invaluable patience and mentorship, and for generously sharing his time and expertise. I am also thankful to Seyyed Saleh Hosseini for his support and feedback, and to Ryan Kuhn for developing the image similarity analysis pipeline leveraged in this paper.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

In radiation therapy (RT), radiation needs to be delivered accurately to the planning target volume (PTV) to eliminate cancerous tissues while simultaneously delivering minimal radiation to surrounding healthy organs and prevent side effects [1]. Therefore, the accurate distinction between cancerous and normal tissues is essential. Currently, computed tomography (CT) is primarily used for RT planning, including the target localization and organ contouring, because the RT treatment simulation process requires dose calculations which use the tissue electron density information obtained from the CT [2]. However, CT lacks good soft-tissue contrast, especially in sites such as the brain, head and neck, and pelvic anatomical regions, making it extremely difficult to distinguish healthy organs from cancerous target areas [3,4]. In addition, multiple iterations of CT scans expose patients to additional ionizing radiation, which bring potentially harmful effects to patients [2].

Magnetic resonance imaging (MRI) is increasingly used for RT planning due to its superior soft-tissue contrast without non-ionizing irradiation compared to CT [2], which facilitates the delineation of the tumor and healthy surrounding organ. However, radiation dose estimation based solely on MR images is challenging because they do not provide direct information on electron density [5]. As well, compared with CT scans, MRI takes a much longer time and is more expensive [6]. Thus, given the above benefits and drawbacks of each imaging method, it would be optimal to collect an MR image set for delineation of the treatment area and surrounding healthy organs, and convert it into a synthetic CT (SynCT) data set for RT dose calculations and treatment plan simulation.

Recently, there has been much research interest being directed toward the conversion of MRI into SynCT based on Deep Learning (DL) methods (DL). This approach has the advantage of fast SynCT generation and can account for setup changes between multiple imaging modalities [7]. For general image applications, a convolutional neural network (CNN or ConvNet) is a popular class of deep neural networks (DNN) using a set of convolution kernels/filters for detecting image features; however, the drawback of generator-only CNN models is that they may lead to blurry results due to generally misalignment between MR and CT [7,8]. Another approach is the use of Generative Adversarial Networks (GANs) that can synthesize better SynCT images than previous generator-only CNN models, in which the generator and discriminator networks are trained simultaneously through min–max optimization strategy [7,9]. Previous studies noted that GANs which were trained using co-registered MR and CT image pairs with a high degree of pixel-to-pixel anatomical correlation achieved higher accuracy. However, acquiring large datasets of high-quality patient image pairs is challenging due to practical considerations (e.g., different scanners, anatomical deformations, motion artifacts, etc.). Thus, previous research in SynCT generation from MR images have used either small, paired samples or unpaired CT and MR images [1,8,10].

Therefore, we propose a method to synthesize SynCT images from MR images using a cycle-consistent GAN (CycleGAN) framework that allows either paired or unpaired image-to-image translation. The methodology includes the investigation of multiple model parameters to assess the performance of several CycleGAN models. To improve the overall model performance, the raw MR and CT image sets are first segmented to generate an image mask to indicate the anatomical overlap across both images. This pre-processing step enables the

extraction of anatomical features and avoid imaging artifacts and non-anatomical structures, making the DL modelling more accurate and efficient. Since the input data quality is improved, this will ensure less training data is required to generate a high-quality SynCT model.

2. METHODS AND MATERIALS

2.1 Data Acquisition

The dataset used in this study was evaluated and approved by the research ethics board (REB) of the Princess Margaret Cancer Centre. Our dataset consisted of 50 cases of paired CT and 3T MR image series which were previously acquired on patients receiving radiotherapy for brain tumors. The CT and MR image voxel size was 1 mm^3 and 1.2 x 1.2 x 1mm^3, respectively. Of the 50 MRI–CT image pairs, 76% (38 pairs) were used for network training and the remaining 24% (12 pairs) were used for testing and validation.

2.2 Data Preprocessing

The MR and CT datasets were rigidly co-registered in the RayStation v9A (RaySearch, Stockholm, Sweden) treatment planning software as per the brain RT clinical protocol. Since the MR and CT image series were acquired and formatted with different parameters, additional image processing steps were conducted to generate a slice-to-slice rendering of the same anatomy in the two imaging modalities. Specifically, the MR data sets were resampled into the CT frame of reference (i.e., image orientation and resolution) by using scripts based on IronPython in RayStation.

Furthermore, the MR and CT were segmented in RayStation to delineate only the anatomy that was common to each paired dataset. This step was necessary to increase accuracy of the DL model training and reduce computation time.

Once all CT and MR data sets were pre-processed, the DICOM images and corresponding RT structure files containing the anatomical contours were exported from RayStation and used for DL model training and validation.

The DL pipeline consists of three key components: (a) pre-processor, (b) module for DL training and analysis for an arbitrary model, and (c) post-processor. The pre-processor reads in the DICOM file data, assigns a value to all points in the patient contour - 1 for points on or within the anatomical boundary, and 0 for points outside the boundary, and then stores these values in a data array. This represents image masks that embeds the DICOM coordinates of the external body anatomical contours. The binary masks also correspond directly to the MR and CT images as shown in Figure 1. Of note, the area of interest within the contour appears white, whereas the area outside of the boundary appears black. The mask serves as a guide to extract paired data from the MR and CT images, i.e., pixel information within the mask overlaid on both CT and MR image slices. This will ensure that input to the CycleGAN model will include only the anatomical structures that both the CT and MR have in common and excludes non-anatomical information present in the images such as the rendering of the CT diagnostic table, patient immobilization mask (where present) and imaging artifacts, etc.

(a) (b) (c)

Figure 1. Example of data pre-processing step prior to DL model training and validation: (a) reference MR image, (b) binary mask corresponding to the same anatomical area from both MR and CT images, and (c) reference CT image. All images correspond to the same slice within the patient anatomy.

Several flavors of GAN models were investigated as discussed in Section 2.3. The post-processor re-formatted the SynCT image data generated by the GAN models to facilitate testing and validation. The SynCT data was further processed as DICOM files similar to the MR and CT image data series used for training. This allowed DICOM data import and analysis for RT specific applications within RayStation.

2.3 Model Architecture

Generative adversarial networks (GANs) are neural network architectures that consist of two sub-networks: *G*, a generator and *D*, a discriminator. *G* learns a mapping from a latent variable *z* (typically random noise) to an image *y* in a target domain, and *D* learns to discriminate the generated image *G(z)* from the real image *y* [12]. During the training of a GAN, both *G* and *D,* are learned simultaneously, with *G* aiming to generate images that are indistinguishable from the real images, and *D* aiming to tell apart generated and real images [13].

CycleGAN is a type of GAN that consists of two generators (G_1 and G_2) and two discriminators (D_1 and D_2), the architecture applied in this study is as shown in Figure 2.

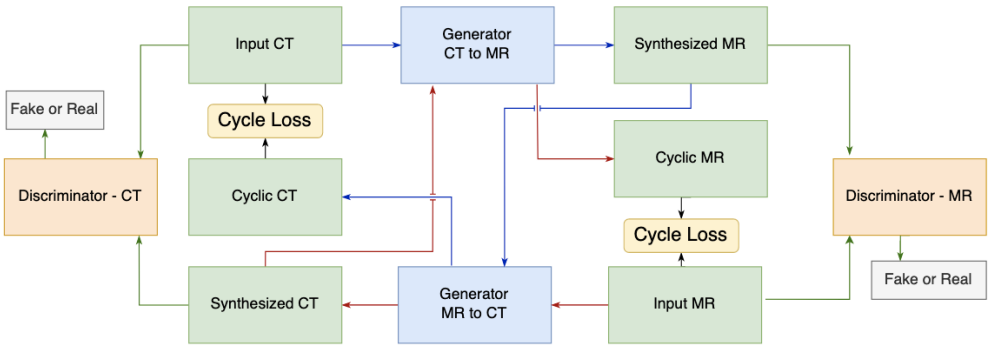


Figure 2. Overall structure of the CycleGAN model.

To train the model, a paired input CT-MR dataset was fed into the architecture with a learning rate of 0.0002 and the split factor of 0. All models were implemented using

TensorFlow and trained on an NVIDIA GeForce RTX 3090 compute unit featuring 64 GB of memory.

The main model architecture design choices investigated in this paper are the generator structure, discriminator structure, objective losses for paired training, and image training size. Firstly, a commonly used generator's architecture is the conventional U-Net; however, ResNet has proved to be more versatile for the SynCT image synthesis problem [14], specifically it is easier to optimize and can gain enhanced accuracy from considerably increased depth. Therefore, in this study, the generator neural architectures of all CycleGAN model investigated were chosen as variations of ResNet in terms of the number of layers. Next, the same discriminator - PatchGAN, described in Section 2.3.2 was used for all experiments. Furthermore, there are two types of losses explored in this paper – (a) GAN + L_1 loss, and (b) L_1 loss. In the following sections, these factors will be further elaborated upon. Lastly, the image size input is varied to be either 256 x 256 or 512 x 512 to assess the model performance across different input sizes, where 512 x 512 refers to data being resampled in patches of 256 x 256 across the entire image.

A total of nine CycleGAN models were investigated in this study, each with varying design parameters that were trained and tested with the same datasets, and their individual performance was evaluated using image similarity metrics. The model specifications are shown in Table 1.

Table 1. Specifications for each of the 9 CycleGAN models to be tested.

Model	Regularization Method (Loss)	Generator Neural Architecture	Training Image Size
1	L_1	ResNet_9	256 x 256
2	L_1	ResNet_6	256 x 256
3	L_1	ResNet_3	256 x 256
4	GAN + L_1	ResNet_12	256 x 256
5	GAN + L_1	ResNet_9	256 x 256
6	GAN + L_1	ResNet_6	256 x 256
7	GAN + L_1	ResNet_3	256 x 256
8	GAN + L_1	ResNet_9	512 x 512 (patches of 256 x 256)
9	L_1	ResNet_9	512 x 512 (patches of 256 x 256)

2.3.1 Generator Structure

To assess the effect of network size on the model performance, different ResNet generator models - ResNet_3, ResNet_6, ResNet_9 and ResNet_12 are investigated in this paper.

ResNet_3

Inspired by [15], the generator is structured as a sequence of a number of convolutional layers, residual blocks and fractionally strided layers. In particular, our network consists of three convolutional layers, three residual blocks, two fractionally strided convolutional layers, and one final convolutional layer. The guideline provided in [16,17] is used to select the scheme of each layer. The detailed structure of the network is as follows:

Relu-INorm-CF64K7S1 → Relu-INorm-CF128K3S2 → Relu-INorm-CF256K3S2 → 3 × RF256K3S1 → Relu-INorm-SCF128K3S2 → Relu-INorm-SCF256K3S2 → Tanh-CF1K7S1

In the above notation, Relu denotes the rectified linear unit activation function, INorm denotes instance normalization, CFαKβSμ represents a convolutional layer with α filters, kernel size of β and stride μ. Similarly, SCFαKβSμ denotes a fractionally strided convolutional layer. RFαKβSμ shows a residual block following the structure recommended in [18]. Note that tangent hyperbolic (Tanh) function is used at the last layer to map the output to the range of [-1, 1]. The generator architecture for ResNet_3 is shown below in Figure 3 above, with the residual blocks being represented in red boxes.

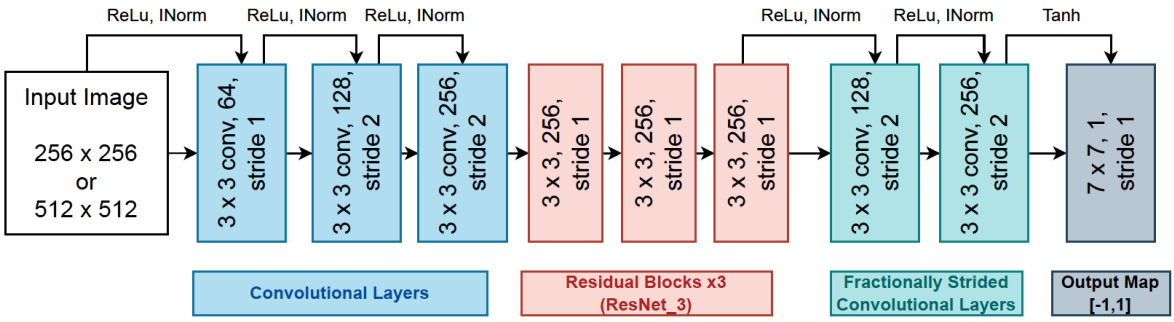


Figure 3. Model architecture of the Generator. Example is for ResNet_3. Input image of 512 x 512 refers to data being resampled in patches of 256 x 256 across the entire image.

ResNet_6, ResNet_9 and ResNet_12

In addition to ResNet_3, we have also explored ResNet_6, ResNet_9, and ResNet_12 as alternative generator structures. The model structures are similar to that of ResNet_3, with the only difference being the number of residual blocks. Instead of having 3 RF256K3S1 residual blocks, ResNet_6, ResNet_9 and ResNet_12 would have six, nine and twelve of such structures each, respectively. In reference to Figure 3, this means the number of residual blocks (coloured in red) would change, depending on the generator structure selected.

2.3.2 Discriminator Structure

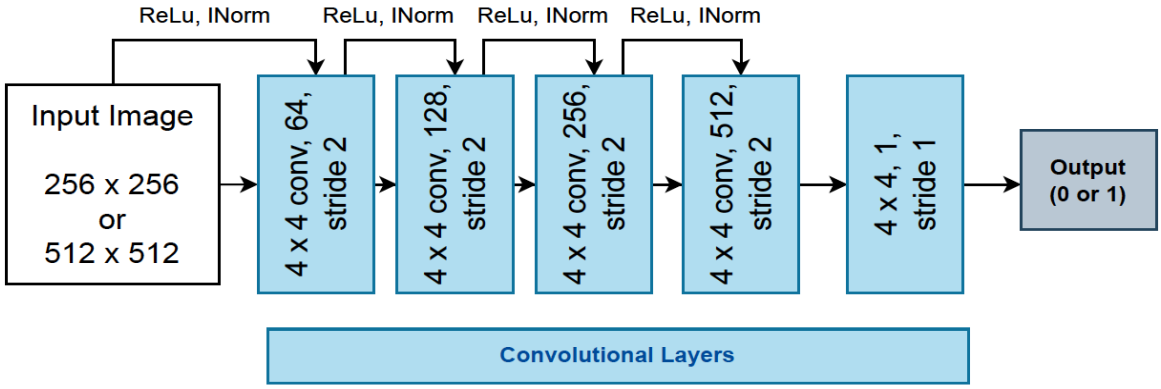


Figure 4. Model architecture of the Discriminator. Input image of 512 x 512 refers to data being resampled in patches of 256 x 256 across entire image.

The discriminator is selected as a PatchGAN [19] to recover the high frequency details of the image and produce sharper images, as the L_2 norm generally cares only about the low-level information. Instead of labelling the whole image with a scalar using a classifier, in a PatchGAN the input image is divided into a number of overlapping patches, and the structure assigns a label to each patch in the output. Comparing real versus fake patches by the PatchGAN discriminator directs the generator to synthesize sharper images with high frequency content similar to the real ones. The discriminator is structured with six convolutional layers as follows:

LRelu-INorm-CF64K4S2 → LRelu-INorm-CF128K4S2 → LRelu-INorm-CF256K4S2 → LRelu-INorm-CF512K4S2 → CF1K4S1

The discriminator architecture is shown in Figure 4.

2.4 Objective Losses for Paired Training

2.4.1 GAN + L_1 Loss

In this structure, the basic assumption is that corresponding CT and MR slice images of the same anatomy are available for network training. Our structure consists of a generator synthesizing images from MR to CT domain (M2C: MR → CT). Moreover, there is a discriminator called D_{CT}, which is trained to classify the fake images against the real ones. The discriminator performance is deployed as a critic and the generator is trained towards synthesizing real-looking CT images and consequently fooling the discriminator.

Mathematically speaking, D_{CT} is trained to minimize the loss:

$$L_{D_{CT}}(D_{CT}) = 1/2\ E_{CT\sim p_{data}(CT)}[\|D_{CT}(CT) - 1\|^2] + 1/2\ E_{MR\sim p_{data}(MR)}[\|D_{CT}(M2C(MR))\|^2] \quad (1)$$

where p_{data} (CT) and p_{data} (MR) denote the distributions of the CT and MR images, respectively. In other words, D_{CT} is trained to label the real CT images as 1 and synthetic images as 0. Note that, as proposed by [20], and in order to have a stable training process, here the adversarial loss given in equation (1) has been defined in the form of L_1 norm, instead of logarithmic formulation. On the other hand, the generator M2C is trained to synthesize images which cannot be distinguished from the real CTs by D_{CT}. More precisely, it must minimize the adversarial loss:

$$L_{adv,CT}\ (M2C) = E_{MR\sim p_{data}(MR)}[\|D_{CT}(M2C(MR)) - 1\|^2] \quad (2)$$

Training using just this adversarial loss, the M2C only tends to generate images with features similar to the CT domain. In fact, this loss is not able to constrain the generator on its own to synthesize a CT image corresponding to the input MR. So, another loss term is required to put such a constraint on the generator training [16,17]. This loss is defined as a pixel-wise L_1 distance between the synthesized CT and its corresponding ground truth. Let P_{CT} : MR → CT denote the operator which returns the corresponding paired CT of the given MR. Then, L_1 loss is defined as:

$$L_{dir,CT}(M2C) = E_{MR\sim p_{data}(MR)}[\|M2C(MR) - P_{CT}(MR)\|] \quad (3)$$

Using the above equations, total generator loss is re-defined as:

$$L_{GAN}(\ M2C) = L_{adv,CT}\ (M2C, D_{CT}) + \lambda_{CT} L_{dir,CT}(M2C) \quad (4)$$

where $\lambda_{CT} \in R^+$ is a factor to balance the losses. So, the discriminator and generator in this paired-data GAN structure are optimized using the following equations:

$$D_{CT} = arg\ min_{\widehat{D}_{CT}}\ L_{D_{CT}}(\widehat{D}_{CT}) \quad (5)$$

$$M2C = arg\ min_{\widehat{M2C}}\ L_{GAN}(\widehat{M2C}) \quad (6)$$

2.4.2 L_1 Loss

In this structure, no discriminator is incorporated in the structure. In other words, generator M2C is trained aiming at minimizing only the pixel-wise L_1 distance between the synthesized CT and its corresponding ground truth, formulated as equation (3). Therefore, the generator is optimized based on the following equation:

$$M2C = arg\ min_{\widehat{M2C}}\ L_{GAN}(\widehat{M2C}) \quad (7)$$

2.5 Evaluation Metrics

Ten indicators have been used to evaluate the SynCT generation models – namely, structural similarity index measure (SSIM), mean absolute error (MAE), mean squared error (MSE), root-mean-square error (RMSE), peak-signal-to-noise-ratio (PSNR), multi-scale structural similarity index measure (MSSSIM), universal quality index (UQI), relative global error (ERGAS), spatial correlation coefficient (SCC), and visual information fidelity (VIF).

2.5.1 Structural Similarity Index Measure (SSIM)

SSIM measures the similarity of two images, in which the picture is highly structured, and there is a strong correlation between adjacent pixels [21,22]. The value of SSIM ranges between 0 and 1, where the two images are exactly alike when SSIM is 1. The formula for SSIM calculation is as follows:

$$SSIM(x,y) = [l(x,y)]^{\alpha}\,[c(x,y)]^{\beta}\,[s(x,y)]^{\gamma} \tag{8}$$

From the formula above, there are three components, where *l(x,y)* compares luminance; *c(x,y)* compares contrast; and *s(x,y)* compares structure. *X* represents the synthesized CT image, and *Y* represents the reference CT image. As well, α, β and γ are parameters that define the relative importance of the three components. *L, C,* and *S* are used to make a comparison of the brightness information, contrast information, and structure information between the reference and the synthesized CT image, respectively.

2.5.2 Mean Absolute Error (MAE)

The methodology to calculate MAE is similar to [10], with the only difference being MRI is being converted to CT in this paper, instead of the converse. MAE measures the average distance between each pixel of the synthesized and the reference MRI image; the smaller the MAE value, the more similar the two images are, and thus the better the SynCT generation model. The reference image, CT and synthesized CT images, M2C(MR), were compared using the mean absolute error (MAE), which is defined as follows:

$$MAE = \frac{1}{N}\sum_{i=0}^{N-1} \| CT(i) - M2C\big(MR(i)\big)\| \tag{9}$$

where *i* is the index of the 2D axial image slice in the aligned voxels and *N* is the number of slices in the reference CT images.

2.5.3 Mean Squared Error (MSE)

MSE measures the average squared distance between each pixel of the synthesized and the reference MRI image. Since the error between the reference and synthesized CT images might be negative, MSE is used to avoid negative numbers and the formula is defined as follows [23]:

$$MSE = \frac{1}{N}\sum_{i=0}^{N-1}\Big(CT\,(i) - M2C\big(MR\,(i)\big)\Big)^2 \tag{10}$$

where *i* is the index of the 2D axial image slice in the aligned voxels and *N* is the number of slices in the reference CT images; the smaller the MSE value, the more similar the two images are.

2.5.4 Root-Mean-Square Error (RMSE)

RMSE is used to measure the difference between the source CT image and the synthesized CT image; the smaller the value of RMSE, the more similar the two images are [24]. The formula for RMSE is as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{M}\sum_{i=1}^{N}\left[\left(M2C(MR)(i,j)\right)-CT(i,j)\right]^2}{MxN}} \tag{11}$$

where *M* and *N* are the size of the image, *i* and *j* are the pixel positions in the image. *M2C(MR)(i,j)* is the synthesized CT image and *CT(i,j)* is the original CT image.

2.5.5 Peak-Signal-to-Noise-Ratio (PSNR)

PSNR is the ratio between the maximum power of a signal to the maximum power of a noise signal and it is an excellent measure of quality for white noise distortion [23,25]. It is mainly used to evaluate the sensitivity error of synthesized CT images' quality, and it is an important indicator to measure the difference between two images [22,26]. The larger the PSNR value, the more similar the two images are. The formula for PSNR is as follows:

$$PSNR = 20\ log_{10}\left(\frac{\mathrm{L}^2}{MSE}\right) \tag{12}$$

MSE represents the mean square error of the reference and synthesized CT images, whereas *L* is the dynamic range of the image pixels [23].

2.5.6 Multi-Scale Structural Similarity Index Measure (MSSSIM)

MSSSIM was introduced as a means for including image details at different scales [27], since the multi-scale method is a convenient way to incorporate image details at different resolutions. Similar to SSIM, the value of MSSSIM ranges between 0 and 1, where the two images are exactly alike when MSSSIM is 1.

MSSSIM has the same components and parameters as SSIM, but the key difference is that MSSSIM evaluation is obtained by combining the measurement at different scales using the following formula:

$$MSSSIM(x,y) = [l_M(x,y)]^{\alpha_M} \cdot \prod_{j=1}^{M}\left[c_j(x,y)\right]^{\beta_j}\left[s_j(x,y)\right]^{\gamma_j} \tag{13}$$

where *M* is the maximum scale, and *j* is the *j*-th scale.

2.5.7 Universal Quality Index (UQI)

UQI is global rather than being local or specially intended to the images being tested or on the individual observers [23]. The value of Q ranges between [−1, 1], where the two images are exactly alike when the quality index, Q is 1. Q is defined as:

$$Q = \frac{4\sigma_{xy}\,\overline{x}\,\overline{y}}{(\sigma_x^2+\sigma_y^2)((\overline{x})^2+(\overline{y})^2)} \tag{14}$$

where $\overline{x}$ and $\overline{y}$ are the mean values of the original and synthesized CT images, respectively; σ_x^2 and σ_y^2 are the variances, and σ_{xy} is the covariance.

2.5.8 Relative Global Error (ERGAS)

ERGAS is used to compute the quality of the synthesized CT image in terms of normalized average error of each band of the processed image [28]. Increase in the value of ERGAS indicates distortion in the synthesized image; the lower the ERGAS value, the more similar the synthesized CT image is to the reference CT image. ERGAS is calculated as follows:

$$ERGAS = 100\frac{R_{M2C(MR)}}{R_{CT}}\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{RMSE(i)^2}{mean(i)^2}\right)} \tag{15}$$

$\frac{R_{M2C(MR)}}{R_{CT}}$ is the ratio of spatial resolution size of the synthesized CT to the reference CT, and *n* refers to the *n*-th spectral band.

2.5.9 Spatial Correlation Coefficient (SCC)

SCC indicates the spatial correlation between the reference and synthesized CT images, which is also the correlation degree between the two images [22,29]. The SCC value ranges between 0 and 1, where the reference and synthesized CT images are exactly alike when SCC is 1. The SCC formula is as follows:

$$SCC\big(CT, M2C(MR)\big) = \frac{\sum_{i=1}^{N}\sum_{i=1}^{N}(CT_i-\mu_{CT})^2\left(M2C(MR)_i-\mu_{M2C(MR)}\right)^2}{\sqrt{\sum_{i=1}^{N}(CT_i-\mu_{CT})^2\ \sum_{i=1}^{N}\left(M2C(MR)_i-\mu_{M2C(MR)}\right)^2}} \tag{16}$$

μ_{CT} denotes the mean value of the reference CT while $\mu_{M2C(MR)}$ denotes the mean value of the synthesized CT image, M2C(MR).

2.5.10 Visual Information Fidelity (VIF)

VIF is an image quality metric that uses information theoretic criterion for image fidelity measurement [30,11]. In an information-theoretic framework, the information that could ideally be extracted by the brain from the reference image and the information lost due to distortion are quantified in the VIF method using natural scene statistics (NSS), the human visual system (HVS), and an image distortion (channel) model [11].

The VIF is derived from a quantification of two mutual information quantities: the mutual information between the input and the output of the HVS channel when no distortion channel is present (called the reference image information) and the mutual information between the input of the distortion channel and the output of the HVS channel for the test image. The VIF value is represented using a value between 0 and 1, where the reference and synthesized CT images are exactly alike when VIF is 1.

3. RESULTS AND DISCUSSION

3.1 Model Training

In support of the usage of CycleGANs in SynCT generation, sample test images of different anatomical locations - including the most challenging areas to be rendered and generated, are included in this section to demonstrate the high-quality SynCT outputs generated from CycleGAN models with varying parameters. Visually, the outputs of these models are virtually indistinguishable from the reference images, which is illustrated by the slice sampling for the validation dataset in both the axial and sagittal views.

Figure 5 illustrates sample test results of the M2C network, with an input MR image size of 256 x 256. Given that (a) represents the raw input MR image, (h) is the real CT image, and (b) to (g) is the SynCT by various models with an increasing number of epochs, we notice the M2C network can generate a high-quality SynCT that closely represents (h) starting at epoch 200. Of note, the model rapidly improves and quality of the SynCT minimally improves after epoch 200. Similarly, Figure 6 illustrates sample test results of the C2M network with the same input size. Given that (a) to (h) are similar to that of Figure 5, we notice that the C2M network is able to generate a high-quality SynMR that closely resembles (h) starting at epoch 200. Hence, it is sufficient to train both the M2C and C2M for approximately 200 epochs.

Figure 7 shows five rows of images with a resolution of 512 x 512 representing test results generated using various M2C models with GAN + L_1 loss that were specified in Table 1. Each row represents a slice, which refers to a different location in the anatomy. For each anatomical slice, there is (a) an original MR image, (b) original CT image, and (c), (d), (e), (f), and (g) are the results of Models 8, 4, 5, 6, and 7, respectively. From Figure 7, Slice 3 (third row from the top) was the most challenging slice to be successfully converted into a SynCT, because it has the highest gradient of bone structure. Thus, it has the most complex rendering of anatomical structures compared to the other slices – especially when it is contrasted with Slice 5 in the last row. And yet, the model output appears almost identical to (b). Where Figure 7 represents the axial view of the brain, Figure 8 illustrates its sagittal view for the same case given the same loss and input image size. Similarly, each row represents a slice, and (a) to (g) are similar to that of Figure 7. In Figure 8, Slice 2 has the most complex anatomical rendering due to the gradient of its bone structure especially when compared to Slice 4, and its SynCT output also closely resembles (b), the reference CT image.

Figure 9 illustrates the axial view of the brain using test results with an image resolution of 512 x 512 that were generated using models trained with both GAN + L_1 and L_1 losses. The model specifications are detailed in Table 1. The same slices in Figure 7 are displayed here, with the key difference being Figure 7 only showcased models with GAN + L_1 loss, whereas Figure 9 displays results for models with both loss types. Similar to Figure 7, each slice has (a) an original MR image and (b) original CT image, but in Figure 9, (c) and (d) are the results of Models 1 and 5, respectively. Visually, (c) and (d) - the images produced by both models, appear to be nearly identical. Next, Figure 10 represents the sagittal view of the slices illustrated in Figure 9. Here, we observe that the image produced by Model 5 (d) visually appears to have better quality anatomical rendering than Model 1 (c). In later sections, the performance of these models will be quantitatively evaluated and compared using the ten image similarity metrics.

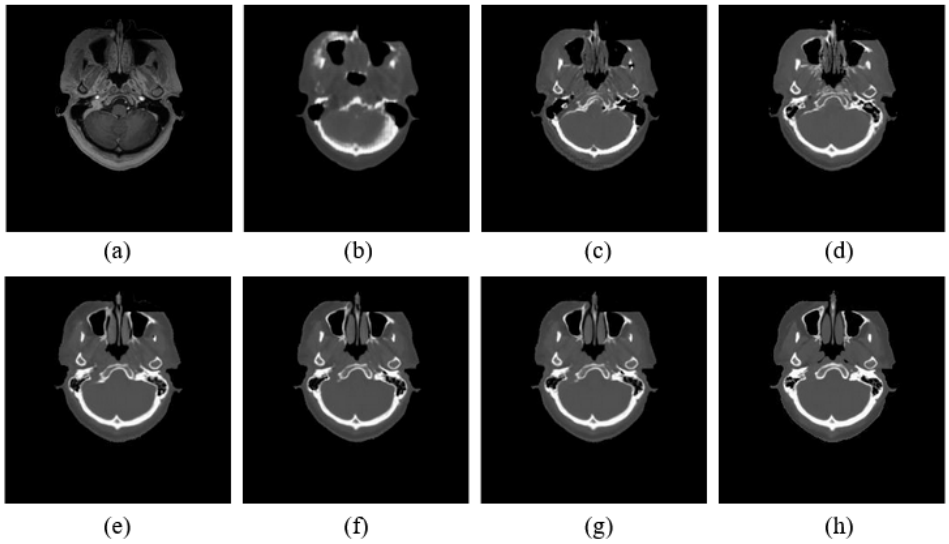


Figure 5. Sample images of 256 x 256 resolution highlighting the M2C progress of the CycleGAN + L_1 model training: (a) MR input image and progress of SynCT images as generated at epochs (b) 1, (c) 20, (d) 50, (e) 200, (f) 300, (g) 400. (h) reference CT image.

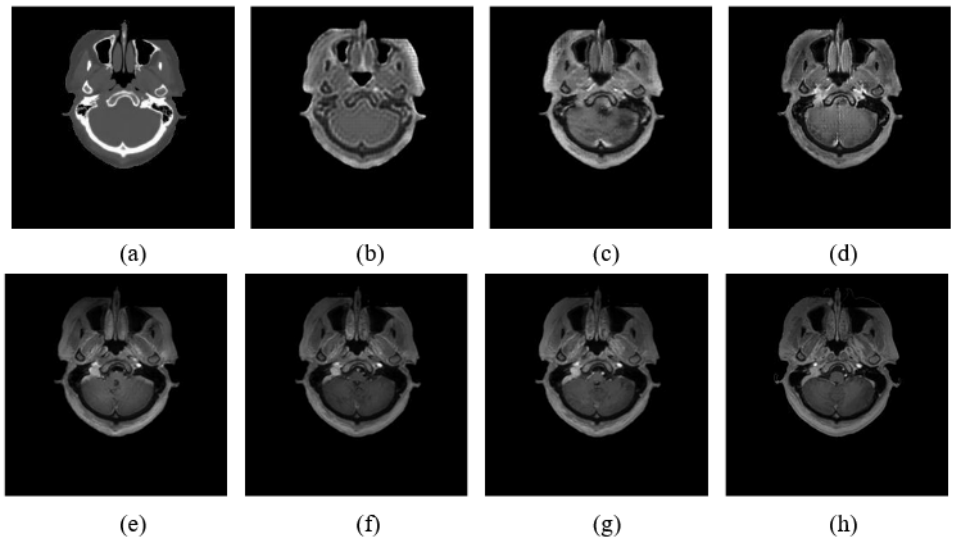


Figure 6. Sample images of 256 x 256 resolution highlighting the C2M progress of the CycleGAN + L_1 model training: (a) CT reference image and SynMR images generated at epochs (b) 1, (c) 20, (d) 50, (e) 200, (f) 300, (g) 400. (h) reference MR image.

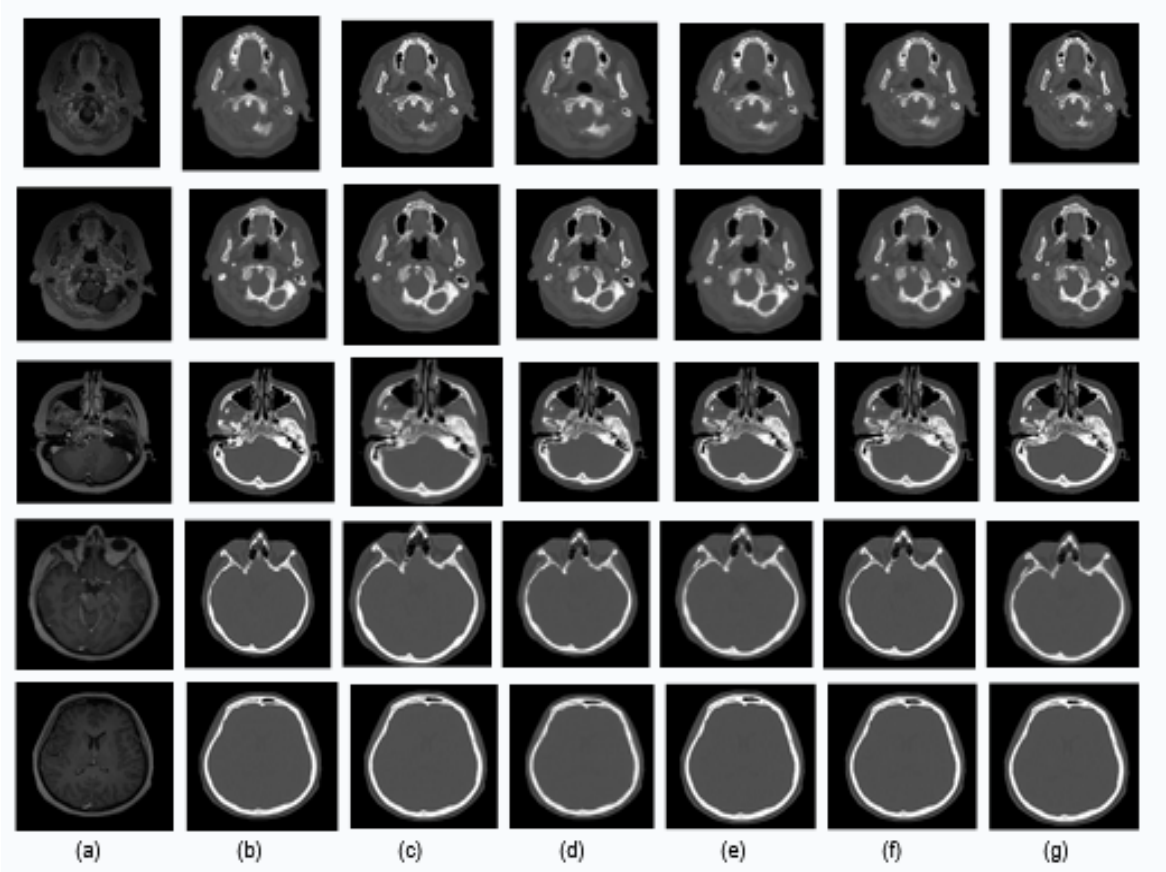


Figure 7. Samples of SynCT results from M2C generators trained with GAN + L_1 methods: (a) Original MR - (b) Original CT - (c) Result of Model 8 - (d) Result of Model 4 - (e) Result of Model 5 - (f) Result of Model 6 - (g) Result of Model 7. See Table 1 for Model specifications.

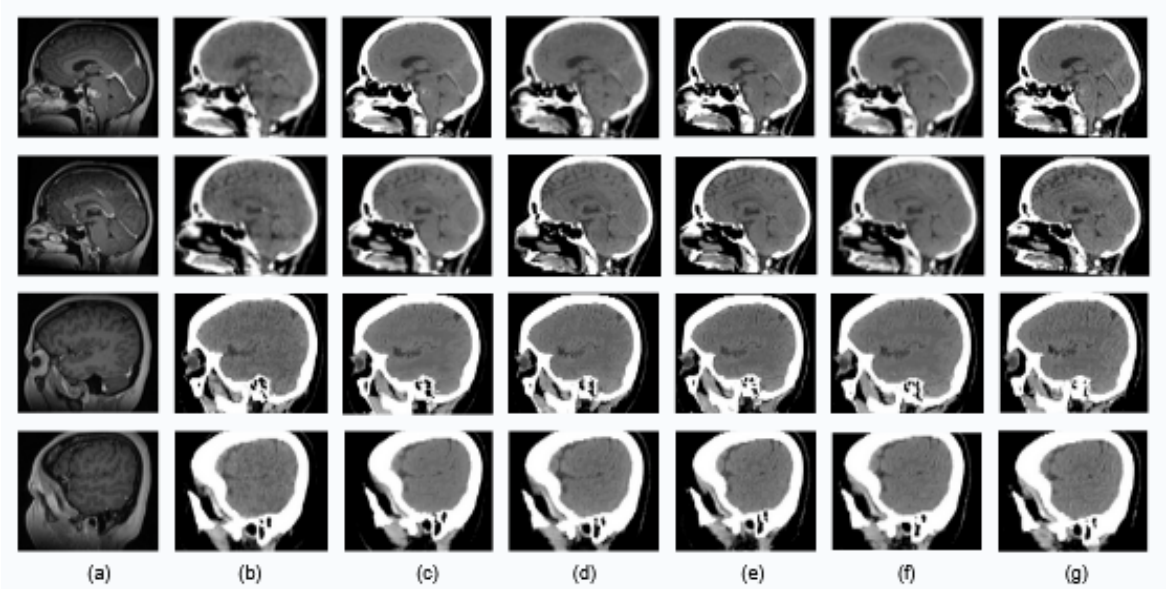


Figure 8. Sagittal view samples of SynCT results from M2C generators trained with GAN + L_1 methods: (a) Original MR - (b) Original CT - (c) Result of Model 8- (d) Result of Model 4 - (e) Result of Model 5 - (f) Result of Model 6 - (g) Result of Model 7. See Table 1 for Model specifications.

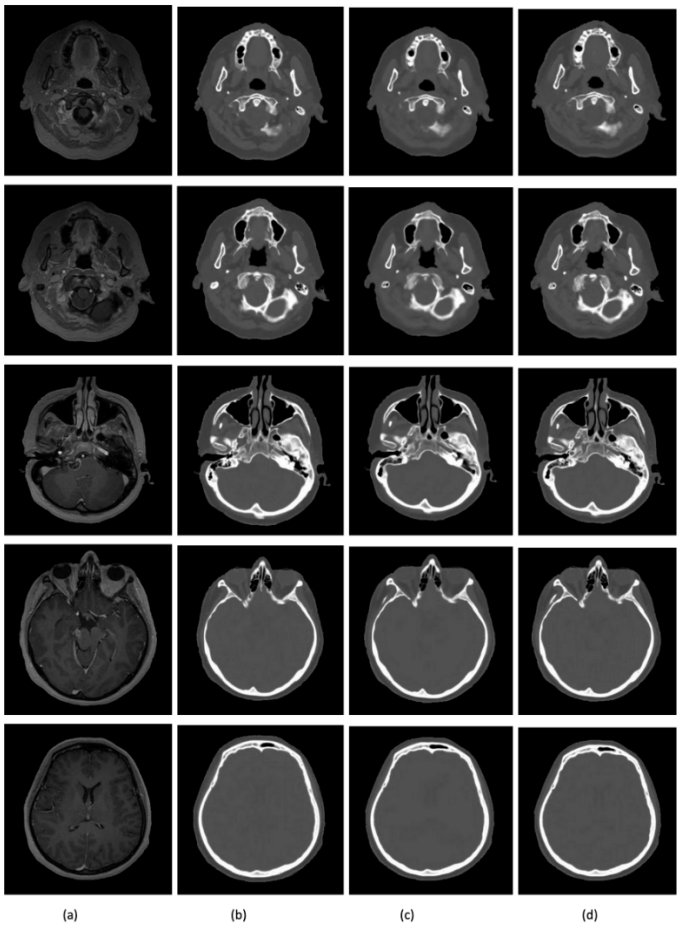


Figure 9. Samples of test results, from generators trained with GAN + L_1 and L_1 methods mentioned in Table 1: (a) Original MR - (b) Original CT - (c) Result of Model 1 – (d) Result of Model 5.

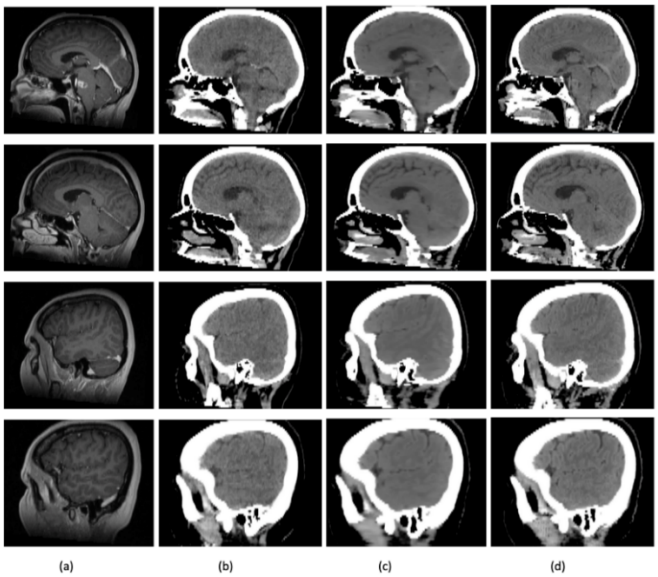


Figure 10. Sagittal view samples of test results, from generators trained with GAN + L_1 and L_1 methods mentioned in Table 1: (a) Original MR - (b) Original CT - (c) Result of Model 1 - (d) Result of Model 5.

3.2 Image Similarity Analysis

3.2.1 Comparison of model performance

Given that each case consists of eighty-eight slices, we obtained the average value for each evaluation metric across all slices in this validation dataset. In doing so, we generated a single data point that represents the model performance for each case to compare the relative performances of the nine models.

Figures 11, 12, and 13 show the results of the image similarity analysis for Case 1 across all nine models listed in Table 1, with all the metric values being normalized relative to the maximum for each metric value in the respective model. This was done to ensure that all data can be displayed in the same plot, and that there can be a fair, relative comparison of model performance. The raw data can be found in Appendix A.1. As per Section 2.3, this thesis aims to investigate the impact of three model parameters on its performance – (a) regularization method, (b) generator architecture, and (c) input image size. Figures 11 and 12 indicate the model performance comparison when the regularization method is varied, whereas Figure 13a and 13b illustrates the model performance when the ResNet structure and input image size is varied, respectively – with the remaining two factors kept constant.

Per Section 3, the ten evaluation metrics investigated in this thesis can be decomposed into three categories – (1) there are five metrics whereby where the SynCT is identical to the reference CT image when the metric value is 1, these include (a) SSIM, (b) MSSSIM, (c) UQI, (d) VIF and (e) SCC. (2) There are four error-related metrics, (a) MAE, (b) MSE, (c) RMSE and (d) ERGAS, where the smaller metric value indicates better model performance. (3) The larger the PSNR value, the better the model performance.

Regularization Method

From Figures 11 and 12, both GAN + L_1 and L_1 losses exhibit similar performances for SSIM, MSSSIM, UQI and VIF, but GAN + L_1 slightly outperforms L_1 based on SCC. Next, GAN + L_1 outperforms L_1 based on all four error-related metrics, except for the case where Model 2 (L_1 loss) outperforms Model 6 (GAN + L_1 loss) in Figure 11b. For PSNR, there is minimal variation between the two types of loss.

Hence, the evidence suggests GAN + L_1 is the superior regularization method. Even though the images produced by Models 1 and 5 appeared to be nearly identical in Figure 9, the quantitative data from Figure 12a suggests that Model 5's performance is superior to that of Model 1. This aligns with the observation based on Figure 10 in Section 3.1.

Generator Architecture

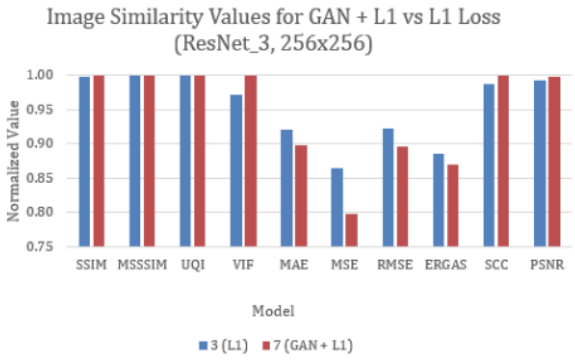
Comparing ResNet_3, ResNet_6, ResNet_9, and ResNet_12 for an input size of 256 x 256 with GAN + L_1 loss in Figure 13a, we discovered that ResNet_3 outperforms all other ResNet variations on all ten metrics. For SSIM, MSSSIM, UQI, VIF and SCC, ResNet_3 produces values that are closest to 1 – which means that the SynCT is nearly identical to the reference CT image. Next, for MAE MSE, RMSE, and ERGAS, ResNet_3 produces the smallest error value, and it produces the largest value for PSNR.

Hence, ResNet_3 is the recommended generator architecture. This finding is significant, because this means that increasing the size of the ResNet structure is unlikely to yield any additional advantages. Thus, a deeper neural network is not necessary to produce a high-performing model that generate high-quality SynCT outputs.

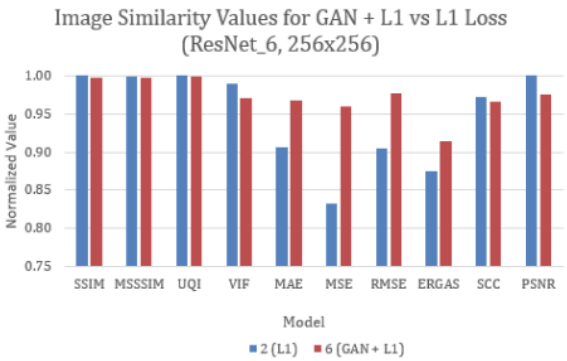
Input Image Size

As demonstrated in Figure 13b, the input image size had minimal impact on SSIM, MSSSIM, UQI, SCC, PSNR, and MAE for two models with an identical ResNet structure (ResNet_9) and regularization method (GAN + L_1). However, there were some performance differences based on image size for four metrics – the model with an input image size of 256 x 256 performs better for VIF and ERGAS, but the model with an input size of 512 x 512 performs better for MSE and RMSE.

Since MSE is one of the most sensitive metrics as per Section 3.2.3 and the relative performance gap between Model 5 and Model 9 based on MSE is more significant compared to that of the other metrics, we infer that using input images with a resolution of 512 x 512 (patches of 256 x 256) tends to improve the model performance metrics.

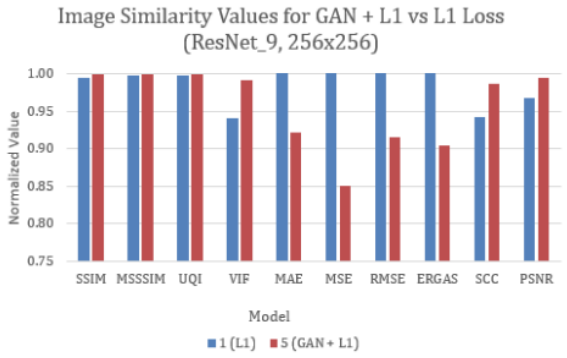


(a)

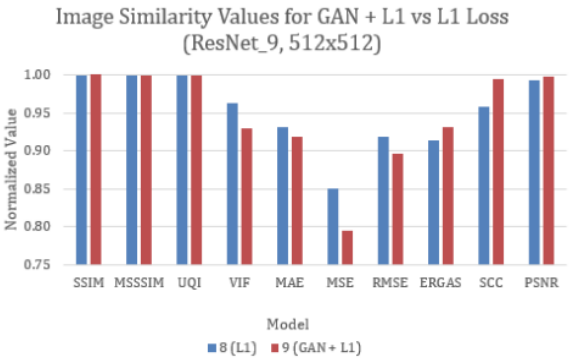


(b)

Figure 11. Image similarity results for GAN + L_1 vs L_1 loss: (a) Model 3 (L_1) and Model 7 (GAN + L_1), ResNet_3, input size of 256 x 256 – (b) Model 2 (L_1) and Model 6 (GAN + L_1) ResNet_6, input size of 256 x 256.

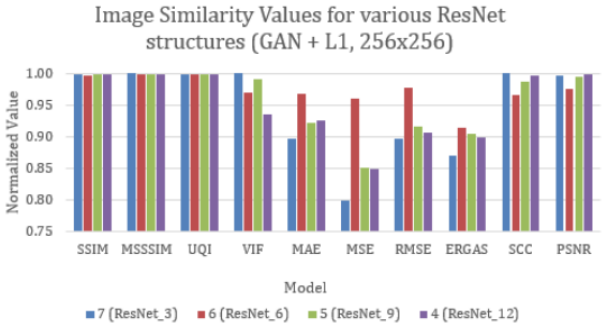


(a)

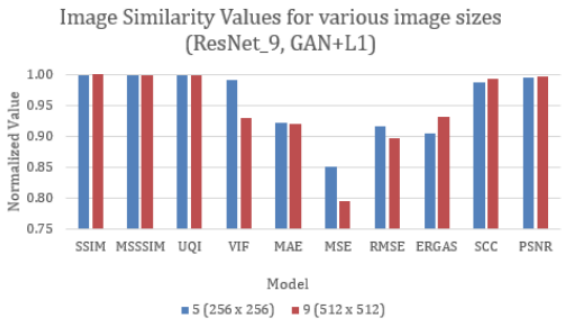


(b)

Figure 12. Image similarity results for GAN + L_1 vs L_1 loss: (a) Model 1 (L_1) and Model 5 (GAN + L_1), ResNet_9, input size of 256 x 256 – (b) Model 8 (L_1) and Model 9 (GAN + L_1) ResNet_9, input size of 512 x 512.



(a)



(b)

Figure 13. Image similarity results for ResNet structure and input image size: (a) Models 7 (ResNet_3), 6 (ResNet_6), 5 (ResNet_9), and 4 (ResNet_12), GAN + L_1 loss, input size of 256 x 256 – (b) Models 5 (256 x 256) and 9 (512 x 512), ResNet_9, GAN + L_1 loss.

3.2.2 Image similarity analysis for multiple slices

Next, using the same validation dataset (Case 1) as in Section 3.2.1, we explored the model performance across all eighty-eight slices of the dataset, which was generated using Model 5. Since the anticipated trend in metrics across the slices within a dataset is expected to remain consistent across all models, it is sufficient to analyse one model and the choice in the example model is immaterial. The raw data for each of the ten image similarity metrics were represented in Figure 14.

From Figure 14, we noticed that the SSIM, MSSSIM, VIF, UQI, and SCC values were stable, which indicates that the model performed equally well across all slices. However, there exists a large variation in the MAE, MSE, RMSE, and ERGAS values, with little variation in the PSNR value, which indicate inconsistent model performance across slices in this validation dataset based on these metrics. Based on the four error metrics (a) MAE, (b) MSE, (c) RMSE and (d) ERGAS, the error values tend to decrease towards the end of the dataset, which indicates that there is an increase in image quality, starting at around Slice 45 for most of these metrics – which is the middle of the dataset.

Overall, the model demonstrates consistent performance across all slices within the dataset, with some variations in the extremities towards the z-direction. Given that most metrics indicate that there is high-quality data in the middle of the dataset as illustrated by the flatter trends in the middle, it is likely that the values at the extremities of the dataset change as the patient position changes. A possible explanation for this is that Slice 0 corresponds to the vortex, which has less anatomical information, whereas Slice 88 corresponds to an area with potentially incomplete information between all the datasets that are used for training. Hence, the image similarity analysis data may be skewed for the anatomical extremities. A potential area of improvement is to increase the size of the dataset to include more information for these extreme areas. However, for the purposes of radiation therapy applications, these areas do not impact the treatment plan dose calculations as they are located away from the target area. Thus, further increasing the dataset may not be crucial for practical applications of the model.

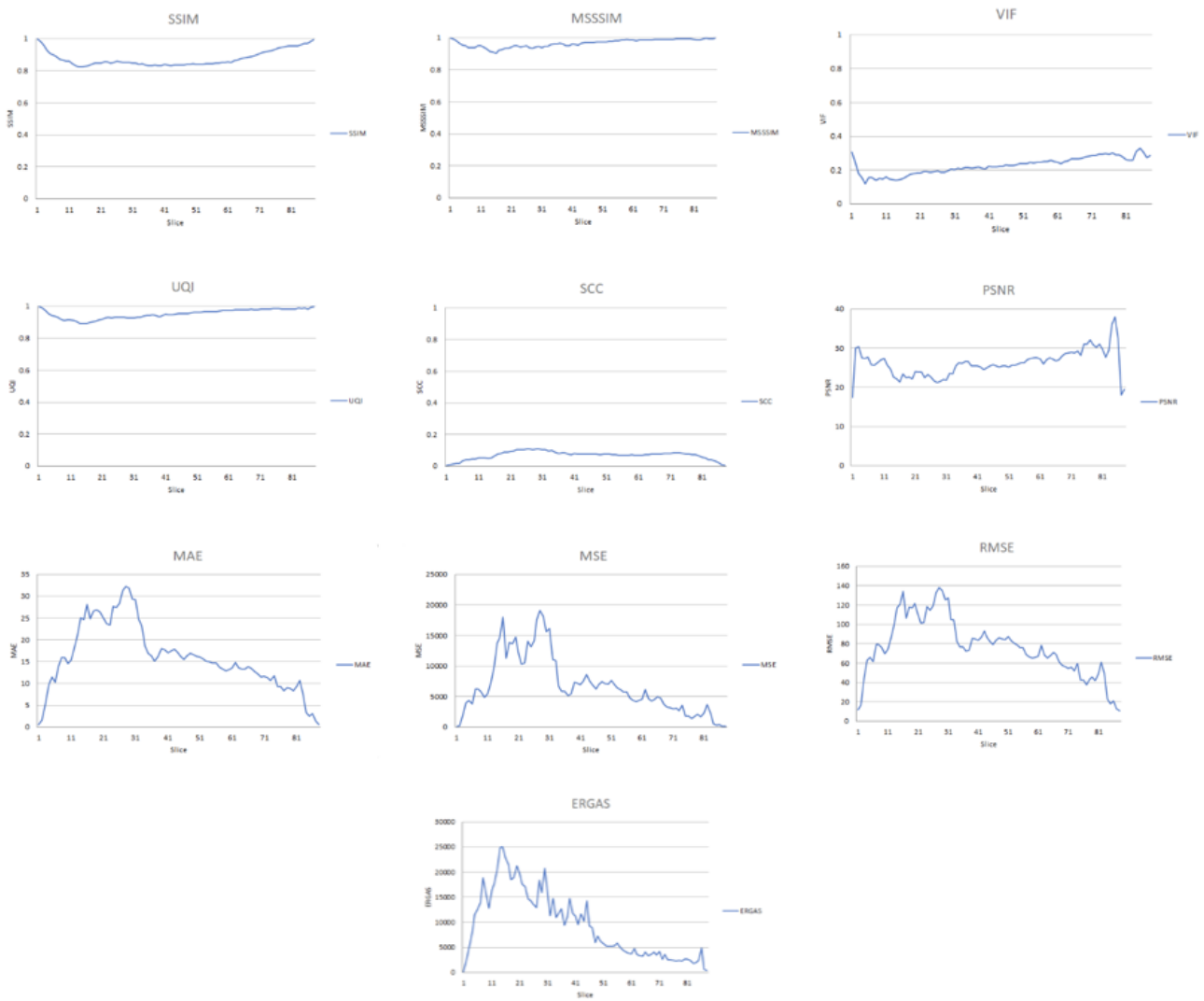


Figure 14. Image similarity results across all slices for Case 1 using Model 5

3.2.3 Image similarity analysis for multiple validation cases

Moreover, we investigated the image similarity results generated using Model 5 to rank the metrics in terms of their sensitivity towards changes in the model, based on a sample population of twelve cases. As per the sample test image visualizations in Section 3.1, using only thirty-eight training datasets is sufficient to generate a high-performing model. This indicates that a dataset size in the tens is adequate to generate high-quality SynCT outputs, thus a large training dataset is not required to produce a good SynCT model.

To ensure a fair and clear comparison between the metrics, we visualized the metric values on the same plot in Figure 15 – with five metrics each in Figures 15a and 15b. The metric values are normalized relative to that of Case 1 to enable this single-plot representation because the data points varied from 0.6 to 10,000 across the twelve validation datasets. The raw data can be found in Appendix A.2.

From Figure 15, we observed that SCC showed the largest variation, which means that it is the most sensitive to changes in the models compared to the other metrics like SSIM, MSSSIM, UQI, and VIF from Figure 15a, as well as PSNR, MAE, and RMSE from Figure 15b that have the smallest relative spread. When combined with the finding from Section 3.2.2; it is noted that SCC has minimal variation across all slices within a single validation dataset, but it is very sensitive to the changes across multiple datasets. From Figure 15b, we noticed that MSE and ERGAS has a moderate relative spread that is only less significant than that of SCC. Figure 16 illustrates the results for SCC, MSE, and ERGAS to compare their sensitivity.

Overall, the three metrics demonstrated similar trends across various test cases in Figure 16, however is interesting to note that when compared against the other metrics, SCC has the opposite trajectory for cases 4 and 5 whereas MSE had a different trend for cases 2 and 3.

When evaluating the performance of machine learning models, metrics that demonstrate significant variation are particularly useful, as they have a greater ability to detect minute changes in the input data or the model itself. In this case, metrics that vary greatly across all test cases are said to be sensitive to even subtle changes within the model, and this sensitivity can make them more effective in comparing and evaluating the relative performance of different models. Hence, SCC appears to be the best metric for model evaluation, followed by MSE and ERGAS. Next, MAE and RMSE are acceptable besides the top three metrics, whereas SSIM, MSSSIM, UQI, VIF, and PSNR are not recommended. Based on the results for all test cases generated from Model 5, we assume the same conclusion can be drawn for all other models.

Hence, we reached the following conclusions regarding the comparison of model performance in Section 3.2.1 based on the top three most sensitive metrics – based on (a) SCC, GAN + L_1 is the superior regularization method in Figures 11 and 12, and ResNet_3 performs equally well as ResNet_12 in Figure 13a. Based on (b) MSE however, ResNet_3 is concluded to be the optimal generator architecture in Figure 13a. Moreover, the MSE trend confirms GAN + L_1 as the recommended loss and suggests that an input resolution of 512 x 512 (patches of 256 x 256 images) improves model performance in Figure 13b. Lastly, based on (c) ERGAS, we observe a similar trend as SCC. Based on this conclusion, Model 7 has the optimal parameter configuration, however its input image size needs to be modified to be 512 x 512 to improve its performance.

Aside from SCC, MSE, and ERGAS, Figure 13a shows that MAE and RMSE are also sensitive metrics that can detect changes in the model, which can be used to compare relative model performance.

(a) *(b)*

Figure 15. Image similarity results across twelve cases using Model 5: (a) SSIM, MSSSIM, UQI, VIF, SCC – (b) MAE, MSE, RMSE, ERGAS, PSNR.

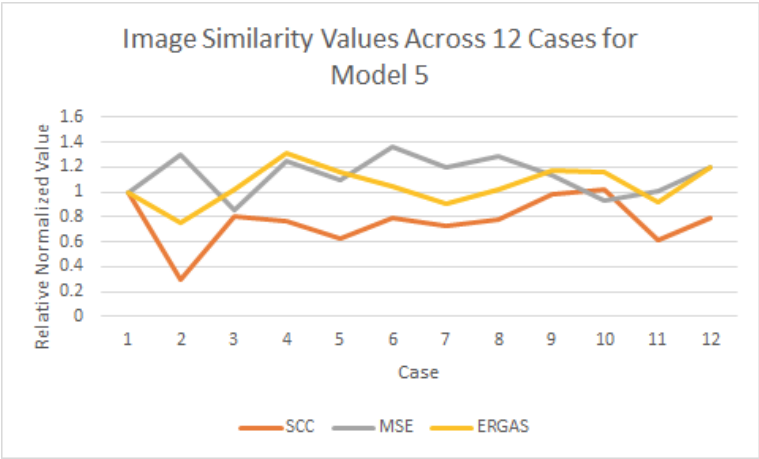


Figure 16. Re-plotting SCC, MSE, and ERGAS image similarity metrics as per Figure 15 to highlight the trending and cross-correlation.

4. CONCLUSIONS

In conclusion, we identified the optimal model parameter configuration for generating SynCTs of the central nervous system. The best-performing model is Model 7 with a ResNet_3 generator architecture and GAN + L_1 loss. However, its performance can be optimized if it was trained with images of 512 x 512 resolution that are patches of 256 x 256 resolution instead of full images with a 256 x 256 resolution, since this tends to improve model performance.

Next, we discovered that the model performance across a single validation dataset is a function of slice, since the performance is dependent on the anatomical location in the dataset. The model performance tends to stabilize in the midsections of the dataset, whereas it varies towards the extremities of the dataset in the z-direction either due to lack of anatomical information in the vortex at the first slice, or incomplete information between training datasets at the final slice. In theory, the dataset can be enhanced by including more information for these extreme areas; however, this may not be necessary for practical applications in radiation therapy as these areas do not impact the treatment plan dose calculation due to their distance from the target area.

Lastly, the metrics were ranked in order of their sensitivity in detecting changes to the input or model. The first tier of the most sensitive metrics includes SCC, MSE and ERGAS, followed by MAE and RMSE in the second tier, and the third tier includes SSIM, MSSSIM, UQI, VIF, and PSNR. The first and second tiers of metrics are recommended to be used for comparing the relative performance of models.

Potential future work from this thesis includes evaluating all nine CycleGAN models based on their performance across the twelve validation cases, with the addition of an upgraded version of Model 7 that is trained on patches of 256 x 256 resolution images. Next, the study on SynCT generation models can be expanded by comparing the performance of CycleGANs against other types of GANs including conditional GANs, for both the central nervous system and other anatomical sites. Lastly, the study on the image similarity metrics in this thesis can be extended by investigating statistical correlations among them to potentially develop a novel metric as a combination of the ten metrics.

5. BIBLIOGRAPHY

[1] S. K. Kang *et al.*, "Synthetic CT generation from weakly paired MR images using cycle-consistent GAN for MR-guided radiotherapy," *Biomedical Engineering Letters*, vol. 11, no. 3, pp. 263–271, Jun. 2021, doi: 10.1007/s13534-021-00195-8.

[2] X. Han, "MR-based synthetic CT generation using a deep convolutional neural network method," *Medical Physics*, vol. 44, no. 4, pp. 1408–1419, Mar. 2017, doi: 10.1002/mp.12155.

[3] C. Rasch, R. Steenbakkers, and M. van Herk, "Target Definition in Prostate, Head, and Neck," *Seminars in Radiation Oncology*, vol. 15, no. 3, pp. 136–145, Jul. 2005, doi: 10.1016/j.semradonc.2005.01.005.

[4] Y. Liu *et al.*, "CT synthesis from MRI using multi-cycle GAN for head-and-neck radiation therapy," *Computerized Medical Imaging and Graphics*, vol. 91, p. 101953, Jul. 2021, doi: 10.1016/j.compmedimag.2021.101953.

[5] H. Arabi et al., "Comparative study of algorithms for synthetic CT generation from MRI : Consequences for MRI -guided radiation planning in the pelvic region," Medical Physics, vol. 45, no. 11, pp. 5218–5233, Oct. 2018, doi: https://doi.org/10.1002/mp.13187.

[6] G. Antoch and A. Bockisch, "Combined PET/MRI: a new dimension in whole-body oncology imaging?," European Journal of Nuclear Medicine and Molecular Imaging, vol. 36, no. S1, pp. 113–120, Dec. 2008, doi: https://doi.org/10.1007/s00259-008-0951-6.

[7] M. Boulanger *et al.*, "Deep learning methods to generate synthetic CT from MRI in radiotherapy: A literature review," *Physica Medica: European Journal of Medical Physics*, vol. 89, pp. 265–281, Sep. 2021, doi: 10.1016/j.ejmp.2021.07.027.

[8] J. M. Wolterink, A. M. Dinkla, M. H. F. Savenije, P. R. Seevinck, C. A. T. van den Berg, and I. Išgum, "Deep MR to CT Synthesis Using Unpaired Data," *Simulation and Synthesis in Medical Imaging*, pp. 14–23, 2017, doi: 10.1007/978-3-319-68127-6_2.

[9] I. J. Goodfellow *et al.*, "Generative Adversarial Networks," *arXiv.org*, 2014. https://arxiv.org/abs/1406.2661

[10] C.-B. Jin *et al.*, "Deep CT to MR Synthesis Using Paired and Unpaired Data," *Sensors*, vol. 19, no. 10, p. 2361, May 2019, doi: 10.3390/s19102361.

[11] S. Mahmoudpour and M. Kim, "Chapter 10 - A study on the relationship between depth map quality and stereoscopic image quality using upsampled depth maps☆," *ScienceDirect*, Jan. 01, 2015. https://www.sciencedirect.com/science/article/pii/B9780128020456000107 (accessed Mar. 20, 2023).

[12]S. U. H. Dar, M. Yurt, L. Karacan, A. Erdem, E. Erdem, and T. Cukur, "Image synthesis in multi-contrast MRI with conditional generative adversarial networks," *IEEE Transactions on Medical Imaging*, vol. 38, no. 10, pp. 2375–2388, 2018.

[13] K. Kurach, M. Lucic, X. Zhai, M. Michalski, and S. Gelly, "A Large-Scale Study on Regularization and Normalization in GANs," *arxiv.org*, Jul. 2018, [Online]. Available: https://arxiv.org/abs/1807.04720

[14] H. Emami, M. Dong, S. P. Nejad-Davarani, and C. K. Glide-Hurst, "Generating synthetic CTs from magnetic resonance images using generative adversarial networks," *Medical Physics*, vol. 45, no. 8, pp. 3627–3636, Jul. 2018, doi: 10.1002/mp.13047.

[15] J. Johnson, A. Alahi, and L. Fei-Fei “Perceptual Losses for Real-Time Style Transfer and Super-Resolution”
[16] A. Radford, L. Metz, and S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”
[17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks”, CVPR, 2017.
[18] S. Gross and M. Wilber, “Training and investigating residual nets,” *Torch*, 04-Feb-2016. [Online]. Available: http://torch.ch/blog/2016/02/04/resnets.html. [Accessed: 20-Mar-2023].
[19] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, “Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks”, Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp.1-10.
[20] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. “Least squares generative adversarial networks”, CVPR. IEEE, 2017.
[21] R. Bhatt, N. Naik, and V. K. Subramanian, “SSIM compliant modeling framework with denoising and deblurring applications,” IEEE transactions on image processing, vol. 30, pp. 2611–2626, 2021.
[22] M. Huang *et al.*, “Remote Sensing Image Fusion Algorithm Based on Two-Stream Fusion Network and Residual Channel Attention Mechanism,” *Wireless Communications and Mobile Computing*, vol. 2022, p. e8476000, Jan. 2022, doi: https://doi.org/10.1155/2022/8476000.
[23] T. Samajdar and Md. I. Quraishi, “Analysis and Evaluation of Image Quality Metrics,” *Advances in Intelligent Systems and Computing*, pp. 369–378, 2015, doi: https://doi.org/10.1007/978-81-322-2247-7_38.
[24] K. S. Tamilselvan and G. Murugesan, “Image Segmentation,” Medical and Biological Image Analysis, Jul. 2018, doi: 10.5772/intechopen.76428.
[25] Sheikh, H.R., Sabir, M.F., Bovik, A.C.: A statistical evaluation of recent full reference image quality assessment algorithms. IEEE Trans. Image Process. 15(11), 3440–3451 (2006)
[26] G. S. Reddy, R. Nanmaran, and G. Paramasivam, “Image Restoration Using Lucy Richardson Algorithm for Deblurring Images with Improved PSNR, SSIM, NC in Comparison with Wiener Filter,” *Alinteri Journal of Agriculture Sciences*, vol. 36, no. 1, pp. 642–649, Jun. 2021, doi: https://doi.org/10.47059/alinteri/v36i1/ajas21091.
[27] Z. Wang, E. P. Simoncelli and A. C. Bovik, "Multiscale structural similarity for image quality assessment," *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, Pacific Grove, CA, USA, 2003, pp. 1398-1402 Vol.2, doi: 10.1109/ACSSC.2003.1292216.
[28] P. Jagalingam and A. V. Hegde, “A Review of Quality Metrics for Fused Image,” *Aquatic Procedia*, vol. 4, pp. 133–142, 2015, doi: https://doi.org/10.1016/j.aqpro.2015.02.019.
[29] K. Wang, G. Qi, Z. Zhu, and Y. Chai, “A Novel Geometric Dictionary Construction Approach for Sparse Representation Based Image Fusion,” *Entropy*, vol. 19, no. 7, p. 306, Jun. 2017, doi: 10.3390/e19070306.
[30] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," in *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430-444, Feb. 2006, doi: 10.1109/TIP.2005.859378.

APPENDIX

A.1 Raw data for Case 1 across 9 models as per Table 1

Model	SSIM	MSSSIM	VIF	UQI	SCC	MAE	MSE	RMSE	ERGAS	PSNR	MAE
1	0.90	0.99	0.33	0.97	0.08	8.12	2528.72	44.62	6680.53	29.63	8.12
2	0.90	0.99	0.32	0.97	0.08	8.33	2740.64	45.93	6808.00	29.47	8.33
3	0.90	0.99	0.32	0.97	0.08	8.75	3041.55	48.61	7030.13	28.97	8.75
4	0.90	0.99	0.32	0.97	0.08	8.19	2636.97	44.98	6717.87	29.71	8.19
5	0.90	0.99	0.32	0.97	0.08	8.33	2691.06	45.57	6950.64	29.55	8.33
6	0.90	0.99	0.31	0.97	0.08	8.42	2693.29	45.70	7020.16	29.49	8.42
7	0.90	0.99	0.31	0.97	0.08	9.04	3166.92	49.74	7683.40	28.76	9.04
8	0.90	0.99	0.30	0.97	0.08	8.31	2518.84	44.57	7154.96	29.63	8.31
9	0.90	0.99	0.30	0.97	0.08	8.37	2690.09	45.08	6897.49	29.69	8.37

A.2 Raw data for Model 5 across 12 test cases (average across all slices for each case)

Model	SSIM	MSSSIM	VIF	UQI	SCC	MAE	MSE	RMSE	ERGAS	PSNR	MAE
1	0.88	0.97	0.23	0.95	0.07	15.84	6738.53	76.42	9366.71	26.15	15.84
2	0.94	0.96	0.15	0.96	0.02	13.52	8750.99	89.68	7039.09	24.69	13.52
3	0.90	0.97	0.22	0.96	0.06	13.76	5729.73	71.30	9582.46	26.56	13.76
4	0.89	0.96	0.18	0.95	0.05	17.97	8419.17	89.08	12312.87	25.25	17.97
5	0.91	0.96	0.19	0.96	0.04	14.46	7370.70	84.48	10809.15	25.59	14.46
6	0.89	0.96	0.20	0.95	0.06	17.37	9217.81	93.55	9749.35	24.48	17.37
7	0.91	0.96	0.20	0.96	0.05	15.69	8075.59	87.99	8519.26	24.90	15.69
8	0.89	0.96	0.20	0.95	0.05	17.37	8625.18	90.92	9553.11	24.73	17.37
9	0.87	0.96	0.22	0.95	0.07	16.51	7599.07	83.81	10978.16	25.63	16.51
10	0.87	0.96	0.21	0.95	0.07	15.86	6309.18	76.56	10861.73	26.02	15.86
11	0.90	0.96	0.18	0.96	0.04	14.35	6745.25	79.00	8610.23	25.66	14.35
12	0.89	0.96	0.20	0.95	0.06	17.15	8028.76	86.77	11261.66	25.46	17.15

(this page is intentionally left blank)