Paired MR-to-sCT Translation using Conditional GANs – an Application to MR-guided Radiotherapy

Alexandra Alain-Beaudoin[1], Laurence Savard[1], and Silvain Bériault[1]

[1] Advanced Development Group, Elekta Ltd,
2050 de Bleury, Montréal, QC, Canada, H3A 2J5
silvain.beriault@elekta.com

Abstract. We present a method for MR-to-sCT image translation using paired training data. The method is based on the Pix2Pix conditional GAN architecture. A multi-channel (2.5D) approach is used to improve translation results thru-plane in comparison to applying a 2D model independently on each slice, while keeping inference time small in comparison to a full 3D approach. Separate models were trained for both brain (T1-weighted) and pelvis (T1- and T2-weighted) using already paired data as provided by SynthRAD2023 challenge. Models were validated using 60 validation subjects provided by the challenge. Image similarity metrics obtained during the validation phase are: mean absolute error (MAE) of 64.27 ±14.15, peak signal-to-noise ratio (PSNR) of 28.64 ± 1.77, structure similarity index (SSIM) of 0.872 ± 0.032.

Keywords: Image Translation, Generative Adversarial Network, Deep Learning, Pix2Pix.

1 Introduction

MR-guided external beam radiotherapy (MRgRT) is one application that can strongly benefit from AI-powered image-to-image translation, for generating synthetic CT (sCT) contrast from MR images – a process we refer as MR-to-sCT. MR-to-sCT is an enabler for MR-only radiotherapy workflows. Such workflow could avoid the use of ionizing radiation imaging, while providing radiation-oncologists (RO) with enhanced soft-tissue visualization (with MRI) alongside co-registered sCT contrast with proper Hounsfield Units (HU) to infer electron density (ED) values for RT dose calculation.

Moreover, MR-to-sCT can significantly enhance current online plan adaptation technique in MRgRT. For example, current online adaptative workflows with Unity MR-Linac (Elekta AB, Stockholm, Sweden) require manual or semi-automatic contouring of different structures on the daily MRI, including all bony anatomy. Bulk electron density values are then assigned to each contoured structure as part of the plan adaptation process. MR-to-sCT could provide more realistic HU and ED values, while significantly reducing the burden of re-contouring all bony anatomy on each daily MRI.

One method for generating realistic sCT is to use deformable image registration of the patient planning CT with the daily MRI. However, deformable registration across

different modalities is limited in accuracy, especially for highly elastic structures like the bladder. Alternatively, AI-powered image translation can generate sCT contrast directly from the daily MRI without the need for deformable registration. One known challenge with this approach is to generate a sCT images with precise HUs.

AI methods based on Generative Adversarial Networks (GANs) are commonly used for image-to-image translation in medical imaging applications. For example, Conditional GANs [1] are commonly used to train models with paired and accurately registered training images. Alternatively, CycleGANs [2] are often used when training data is unpaired. As such, they can be trained using unmatched training samples that are only available in one modality (MR or CT).

This paper describes a method to solve the MR-to-sCT translation problem posed by the SynthRad2023 challenge, for both pelvis and brain tumor sites. Given that all training samples provided for this challenge are paired, we have chosen a paired method based on conditional GAN architecture. Section 2 provides implementation details about the method. Section 3 describes strategies for hyperparameter optimization and summarizes the results obtained during the challenge's validation phase. Section 4 discusses specific design decisions, as well as limitations of the proposed method and potential areas of future work.

2 Method

2.1 SynthRAD2023 Data

For task 1 of the SynthRAD2023 challenge, fully anonymized data was collected from 3 different sites: Radboud University Medical Center, University Medical Center Utrecht, University Medical Center Groningen. The challenge consists of 3 phases: training phase, validation phase and test phase. For the training phase, a set of 180 brain MRI/CT pairs and 180 pelvis MRI/CT pairs was provided to the participants. The MR and CT data were rigidly aligned by the organizers. An evaluation mask was also provided by the organizers. All brain MRIs were T1-weighted. 120 pelvis MRIs were T1-weighted. The remaining 60 pelvis MRIs were T2-weighted. For the validation phase, a set of 30 brain MRI images and 30 pelvis MRI images was provided to the participants. The corresponding CT images were not provided to the participants. For the test phase, data is not provided to the participants.

2.2 2D Conditional GAN

We implemented a 2D conditional GAN (Pix2Pix) network [1] as follows. Let $G: X \rightarrow Y$ be a generator that translates images from domain X to domain Y, and D be a discriminator network trained to distinguish between true and synthetized images in domain Y. The original Pix2Pix objective is:

$$\mathcal{L}(G, D) = \mathcal{L}_{GAN}(G, D) + \lambda \mathcal{L}_{R}(G), \quad (1)$$

where $\mathcal{L}_{GAN}$ is the GAN loss and $\mathcal{L}_R$ is a regression loss measuring the difference between the output translation and the (aligned) ground truth. In our implementation, three possible objective functions can be used for $\mathcal{L}_{GAN}$: classic [3], least squares [4], and Hinge [5]. A L1 loss term is used for $\mathcal{L}_R$:

$$\mathcal{L}_R = \mathbb{E}_{x,y}\|G(x) - y\|_1 \tag{2}$$

The generator network is implemented with a ResUnet [6]. The number of layers and filters at each layer is fully parametrizable. The discriminator network is implemented similarly to the encoding part of the ResUnet. Both the generator and discriminator network can optionally use spectral normalization [7] following each convolutional layer, and instance normalization is used in place of group normalization.

Data augmentation can optionally be used and is randomly run on-the-fly during the training loop. Supported data augmentation is loosely inspired from BigAug [8] and includes: affine transformation, synthetic multiplicative bias field, blurring, sharpening, Gamma contrast change, linear intensity transform. Note that the affine transformation is applied to both MR and CT images. All other forms of data augmentation are applied to the MR images only.

The model is implemented in Python using the PyTorch library. Optimization is carried out using Adam with $\beta1 = 0.5$ and $\beta2 = 0.999$. A slow-moving exponential moving average (EMA) [9] of the generator parameters is tracked during training (with $\alpha = 0.999$, weights updated at every batch) and used as the final model for inference. A complete list of network hyperparameters is provided in Table 1.

2.3 Image Pre-Processing

The voxel resolution for model training was chosen as 1x1x1 mm for brain model and 1x1x2.5 mm for the pelvis. If necessary, MR and CT images are resampled to the model resolution and back to native resolution (during inference). MR and CT intensities are linearly rescaled to a range of [-1, +1], with a source range determined using percentiles for MR and a fixed source range of [-1000, +2200] HU or [-1000, +3000] HU to support metal artifacts. In our experiments, models trained with the full range [-1000, +3000] were slightly less accurate because 1/4 of the intensity range [+2000, +3000] is reserved for unusual intensities. One strategy is to train one network with the full range and one with a narrower range, and use the content of the first network for intensities > 2200 (if any) and the content of the latter network otherwise.

During training, patches with a predefined sample size are randomly drawn from the training set images and randomly augmented. During inference, the test image is subdivided into overlapping patches and the model output combined via a weighted average – when combining the output of multiple overlapping patches, higher weight is given if a pixel is near the center of the patch and lower weight if the pixel is close to the edge of a patch.

Table 1. List of model hyperparameters

Hyperparameter	Description	Recommended value
$\mathcal{L}_{GAN}$	Type of adversarial loss function to use (either Classic, Least Squares or Hinge)	Least Squares
Learning rate	Learning rate for the generator (G) and discriminator(D). The learning rate will linearly decrease to zero starting when half the number of iterations has been completed.	G: 1e-4 D: 5e-5
Num filters	Number of filters per layer in the generator (G) and discriminator (D)	G: 64,128,256,512 D: 64,128, 256,512,512
Num disc updates	Number of discriminator updates per generator update	2
Spectral norm	Specify if spectral normalization layers are used in both the discriminator and generator networks	True
L1 weight (λ)	Weight of L1 loss term	50
Voxel size	Voxel size in mm	Brain: 1x1x1 mm Pelvis: 1x1x2.5 mm
Sample size	The size of a sample (in voxels) used during training and inference. For a 2D network, the 3rd dimension is the channel dimension (c.f. Section 2.4)	192x192x5
Batch size	Number of samples per batch	16
Num batches per iteration	Number of batches per iteration	300
Num iterations	Total number of iterations	2500
Data augmentation	Option to turn on data augmentation	True

2.4 Multi-Channel Implementation

One problem with applying a 2D neural network independently on each axial slice of a 3D image is the so-called "staircase effect" that can appear in other orientations (sagittal and coronal). This is shown in the left image of Fig. 1. One possible mitigation is to train a 3D network. However, this can lead to larger training and inference time, and higher GPU memory consumption in comparison to a 2D approach. Instead, we used a 2.5D approach where five consecutive slices of data are supplied to the network, using the channel dimension. As illustrated in the middle image of Fig. 1, it solves the staircase effect without the drawbacks of a full 3D approach [10].

At training time, no change is required other than supplying randomly sampled groups of 5 consecutive slices of data. At inference time, the test image is subdivided

into patches that overlaps both in-plane and thru-plane. The model output is averaged as previously described in section 2.3.

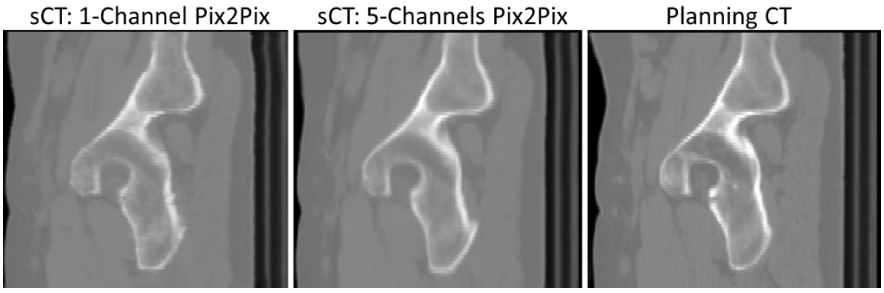

Fig. 1. Use of multiple channels to eliminate "staircase effect" in 2D pix2pix.

3 Results

3.1 Strategy for hyperparameter tuning

During the training phase, we split the data for each anatomy (brain and pelvis) into a training set (75% of data) and a tuning set (25% of data). The training set is used to train the network and the tuning set is used for computing the image similarity metrics – mean absolute error (MAE), peak signal-to-noise (PSNR) and structure-similarity index (SSIM) – at the end of each training run. We started with an initial set of hyperparameters and then launched several trainings for the brain anatomy while varying one parameter at a time. Once a new set of best hyperparameters was found, this procedure was repeated a second time to obtain a final set of hyperparameters. Finally, we did a small hyperparameter search on the pelvis anatomy, but no changes were found necessary other than the voxel size for keeping the training time reasonable.

3.2 Validation phase results

Once the validation phase started, we trained brain and pelvis models independently, using 100% of the training data and using best found hyperparameters. Table 2 shows the best results we obtained during validation. The MAE is 64.27 ± 14.15 the PSNR is 28.64 ± 1.77 and the SSIM is 0.872 ± 0.032. Representative examples of brain and pelvis translations are shown in Fig. 2. For each model, 6 networks were trained with the hyperparameters described in Table 1. The output of all 6 networks were combined with ensemble averaging. The average inference time for one network is approximately 20 seconds for brain images and 30 seconds for pelvis images on a workstation equipped with a NVIDIA V100 16GB GPU card. With ensemble averaging, the inference time scales linearly with the number of networks composing the model.

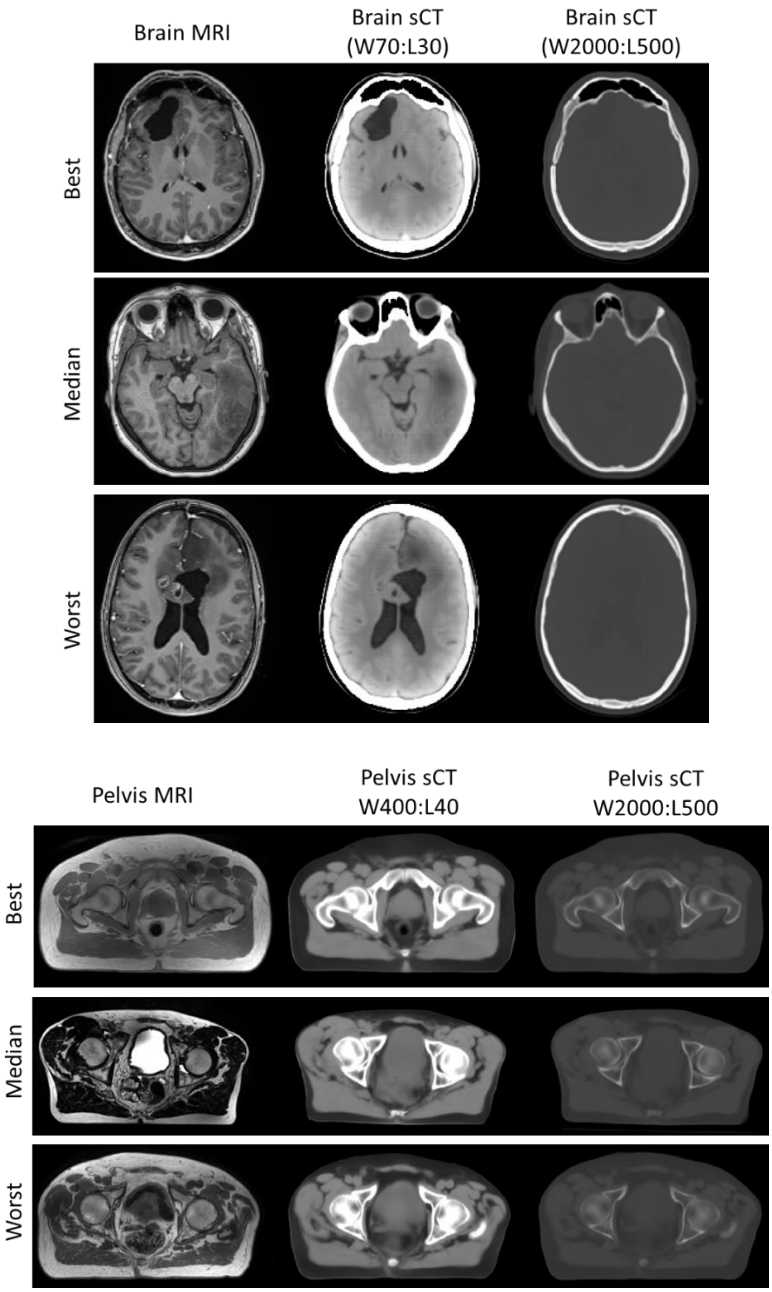

Fig. 2. Representative examples of MR-to-sCT translation on the validation set. Images with best, median and worst MAE are shown. Synthetic CT images are displayed with different window/level settings to emphasis brain and/or soft tissue (middle image) and bone (right image).

Table 2. Final results of validation phase

	MAE	PSNR	SSIM
Mean	64.27 ±14.15	28.64 ± 1.77	0.872 ± 0.032
Min	32.75	24.60	0.789
25pc	55.73	27.56	0.855
50pc	62.28	28.66	0.873
75pc	72.58	29.38	0.890
Max	109.88	34.58	0.969

4 Discussion

Results presented in section 3 indicate that a model based on the pix2pix architecture is suitable for MR-to-sCT image translation. Given that this is a paired method, accurate image registration is necessary between each MR/CT pairs. We used rigid data alignment as provided by the challenge organizers, but we noticed some images could have benefited from deformable image registration (for example, some MR-CT pairs exhibited different bladder size). However, the use of deformable image registration caused degradation of our image similarity metrics. This is either due to difficulties obtaining accurate deformable image registration inter-modality, or because the ground truth validation data was also rigidly registered.

Different pulse sequences were used for the pelvis MRI (2/3 of the training data was T1-weighted, 1/3 was T2-weighted). Additional accuracy could be obtained by training different models for T1-weighted and T2-weighted MRI. This could not be explored since no information about the MR contrast is provided at inference time.

For this challenge, the maximum inference time was set to 15 minutes per image, which is acceptable for offline use, but not suitable for online adaptive workflows. In this work, ensemble averaging leads to small improvements on the overall image similarity metrics. However, the computational cost increases linearly with the number of networks in the model, with possibly very limited impact on the dosimetry. As future work, knowledge distillation [11] could be explored to reduce computational costs.

For the pelvis tumor site, gas pockets in the rectum may not translate well into the sCT. This is because transient gas pocket information is not paired in the training data (i.e. MR and CT scans were not taken simultaneously). Moreover, they are difficult to visualize in MRI. Consequently, gas pockets may be hallucinated or missed out in the sCT. One mitigation could have been to mask all gas pockets in the training CTs with an average HU value. This could prevent the model from hallucinating air pockets during inference and, to some extent, would be consistent with some clinical workflows (large gas pockets are sometimes manually segmented and assigned an average HU/ED). However, during online imaging, gas pockets can provide useful information and radiation therapists may decide to wait for it to pass before starting the treatment.

In conclusion, we presented a 2.5D (multi-channel) pix2pix network for training MR-to-sCT models and demonstrated results for both brain (T1-weighted) and pelvis (multi-contrast) anatomies. Separate models were provided for the different tumor sites as it led to superior image similarity metrics.

References

1. Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A.: Image-to-Image Translation with Conditional Adversarial Networks, (2018).
2. Zhu, J.-Y., Park, T., Isola, P., Efros, A.A.: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. (2017). https://doi.org/10.48550/ARXIV.1703.10593.
3. Goodfellow, I.: NIPS 2016 Tutorial: Generative Adversarial Networks, http://arxiv.org/abs/1701.00160, (2017).
4. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z., Smolley, S.P.: Least Squares Generative Adversarial Networks, http://arxiv.org/abs/1611.04076, (2017).
5. Lim, J.H., Ye, J.C.: Geometric GAN, http://arxiv.org/abs/1705.02894, (2017).
6. Zhang, Z., Liu, Q., Wang, Y.: Road Extraction by Deep Residual U-Net. IEEE Geosci. Remote Sensing Lett. 15, 749–753 (2018). https://doi.org/10.1109/LGRS.2018.2802944.
7. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral Normalization for Generative Adversarial Networks, http://arxiv.org/abs/1802.05957, (2018).
8. Zhang, L., Wang, X., Yang, D., Sanford, T., Harmon, S., Turkbey, B., Wood, B.J., Roth, H., Myronenko, A., Xu, D., Xu, Z.: Generalizing Deep Learning for Medical Image Segmentation to Unseen Domains via Deep Stacked Transformation. IEEE Trans. Med. Imaging. 39, 2531–2540 (2020). https://doi.org/10.1109/TMI.2020.2973595.
9. Brock, A., Donahue, J., Simonyan, K.: Large Scale GAN Training for High Fidelity Natural Image Synthesis, http://arxiv.org/abs/1809.11096, (2019).
10. Avesta, A., Hossain, S., Lin, M., Aboian, M., Krumholz, H.M., Aneja, S.: Comparing 3D, 2.5D, and 2D Approaches to Brain Image Segmentation. Radiology and Imaging (2022). https://doi.org/10.1101/2022.11.03.22281923.
11. Hinton, G., Vinyals, O., Dean, J.: Distilling the Knowledge in a Neural Network, http://arxiv.org/abs/1503.02531, (2015).