Reconstructing temporally high resolution pediatric MRI scans from low resolution images

Ryan Li
Stanford University
ryanli8@stanford.edu

Sreya Halder
Stanford University
halders@stanford.edu

Divya Nagaraj
Stanford University
dnagaraj@stanford.edu

1 Abstract

In this work, we leverage a deep learning pipeline for training a network to improve the resolution of MRI images, leading to shorter image acquisition times. Using a Generative Adversarial Network, we extend previous approaches and focus specifically on super-resolution of the MRIs of pediatric patients, which tend to have more artifacts. While our results show promise, further research into solving vanishing gradients and decreasing pixelation is needed.

2 Introduction

High-resolution magnetic resonance images (MRI) are vital in diagnosing numerous conditions, including brain tumors, traumatic brain injuries, developmental anomalies, strokes, dementia, and infections and are preferred in diagnosis due to its diverse range of applications and low risk posed to the patient [4]. However, a typical scan requires patients to be in the scanner for prolonged periods of time (up to 60-90 minutes). Faster image acquisition could dramatically drive down the amount of time the patient has to be still in the scanner and could help create a better patient experience and cut down the amount of noise from the slight movements of the patient. This is especially true of pediatric patients who tend to be less able to stay still for these extended periods. Thus, our project focuses on MRI super-resolution of pediatric patients specifically. We chose to pursue this project because we all have interests in leveraging algorithms to improve healthcare.

This paper provides a pathway for faster image acquisition, followed by reconstruction of the low resolution images into higher resolution ones that can expedite clinical diagnoses. By using a Generative Adversarial Network (GAN), we transform an input of a low-resolution MRI image to a higher-resolution MRI image. Uniquely, our project chooses to focus on improving temporal resolution in pediatric patients, two factors that are not commonly seen in the field of medical image super-resolution.

3 Related work

Research on image super-resolution, specifically in the realm of medicine, has exploded in the last couple of years, with CNNs and SRGANs primarily used to improve image super-resolution. Additional architectures commonly used to tackle this problem include ResNets, U-Nets, and ensemble learning (or a combination of the former). However, in our literature search, we did not find a model that catered specifically to pediatric patients, which is the novelty of our project. For instance, Ghodrati et al. found very promising results using CNNs with ResNet that was comparable to U-Nets, despite requiring less than 10% of the number of parameters needed for the ResNet [5]. However, the number of parameters needed for both ResNets and U-Nets is extremely large (100,000 parameters and 1.3 million parameters respectively). The other approach considered more

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

state-of-the-art involves using super-resolution GANs (SRGANs). Since Ledig et al published their photo-realistic image super-resolution paper [6], using SRGANs has become state-of-the-art in the medical domain. After reading papers such as Chen et al, Lyu et al, and Wang et al that achieved impressive results on MRI superresolution of scans inspired by Ledig et al's technique, we decided to use this approach of using an SRGAN [7, 8, 9].

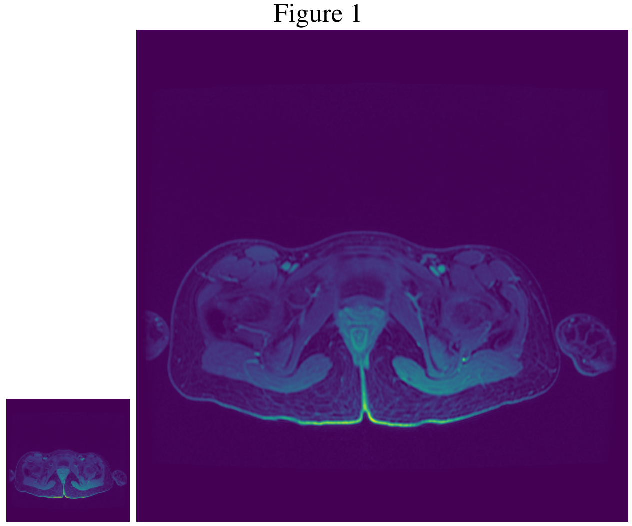
It is also important to note that due to the difficulty in acquiring patient data that medical datasets tend to be a lot smaller than datasets for similar tasks in other domains. Past literature supports that a GAN would be better for smaller datasets [10].

4 Dataset and Features

Courtesy of the Stanford School of Medicine and the Laboratory of Quantitative Imaging and Artificial Intelligence, we are utilizing a dataset of 10 patients, each with about 1,300 full-body high-resolution MRI slices. A slice, in this case, is a DICOM image of size 512x512, at 72 ppi resolution, that begins at the top of the head and scans the entire body. Because the dataset was provided to us from Stanford University Neuroscience researchers, we do not posses information about the backgrounds of these patients. We were also given another dataset of 10 patients, each with about 1,300 low-resolution slices of a 3D fMRI scan of the patient's brain. However, because this specific dataset only contained low-resolution images and for the purposes of our research we required the paired high-resolution images, we were unable to use this dataset to train our model.

Instead, we used a scaling function to create our low resolution images, resizing the images to be 256x256 instead of 512x512 (so scaling down by a factor of 1/2). It's difficult to provide other quantitative data (exploratory data analysis) because of the format of our data, namely that we have more than a thousand cross-sections each for 10 patients. This makes it difficult to perform comprehensive analyses, as we do not want to want to, for example, compare or equate a cross-section of the kidney with one of the thorax. However, we have examples of visualizations of our data below. In order to create our dataset from the DICOM images we are given, we wrote a Python script using Pydicom to convert the DICOM images to NumPy arrays, and then storing each of the arrays as NumPy files. We then created a text file storing paths to each of the stored NumPy arrays, which are loaded into the dataloader. Only the high resolutions images are loaded, and then we scale down the NumPy arrays for each high resolution set of data. We then set aside a set of 64 images to utilize as our test set. An example of our high-resolution patient data can be seen below in Fig. 1, and our generated low-resolution counterpart is seen in Fig. 2. These visualizations were a result of using a custom PyTorch data-loader and scaling factor.

Figure 1



5 Methods

For this project, we used a super-resolution generative adversarial network (SRGAN). An SRGAN is comprised of two major parts - a generator and a discriminator. The two parts work hand-in-hand - the generator tries to create fake images in an attempt to fool the discriminator while the discriminator tries to distinguish between real and generated images.

Most image super-resolution papers utilize the mean-squared error loss function. However, we chose to use a Binary Cross-Entropy with Logits Loss (BCEwithLogitsLoss) function and Smooth L1 Loss. We utilize BCE with Logits loss (which is essentially a combination of a Sigmoid layer and BCE Loss) for content criterion, used to measure our generator content loss. Smooth L1 loss is used for our adversarial criterion, used to measure our generator adversarial loss and discriminator loss (since it is less sensitive to outliers than MSE Loss and also prevents exploding gradients). The generator tries to minimize the loss while the discriminator tries to maximize loss.

BCE with Logits Loss Function

$$l(x,y) = L = \{l_1, ..., l_N\}^T, l_n = -w_n[y_n \cdot log\sigma(x_n) + (1-y_n) \cdot log(1-\sigma(x_n))] \tag{1}$$

Smooth L1 Loss

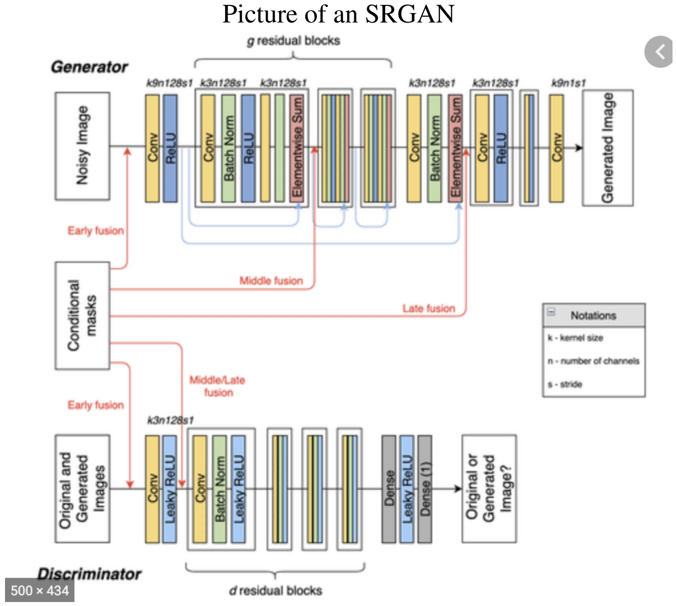
$$loss(x,y) = \frac{1}{n}\sum_i z_i \tag{2}$$

$$z_i = 0.5(x_i - y_i)^2 if |x_i - y_i| < 1 \tag{3}$$

$$z_i = |x_i - y_i| - 0.5, otherwise \tag{4}$$

The typical architecture of a SRGAN looks like the below:

Picture of an SRGAN



We decided to use a BCELoss function because we wanted to start developing with a very simple deep convolutional GAN, and BCELoss is a simple way to measure the binary cross entropy between a target and output (since we have our target to be the high-resolution images and our output to be the generated high-resolution images). In our discriminator and generator we utilize the Leaky ReLU activation function to initialize, and for the forward propagation portion of the discriminator, we use a sigmoid activation function, while for the generator we use a tanh activation function. The main reason for using the various kinds of activation functions was to test out the effects of each, so we plan to tinker with this a bit more as we move forward. We fully acknowledge that our current parameters and functions are not the most ideal for the task, but we did not want to complicate things before we had a good grasp of a more basic version of the model.

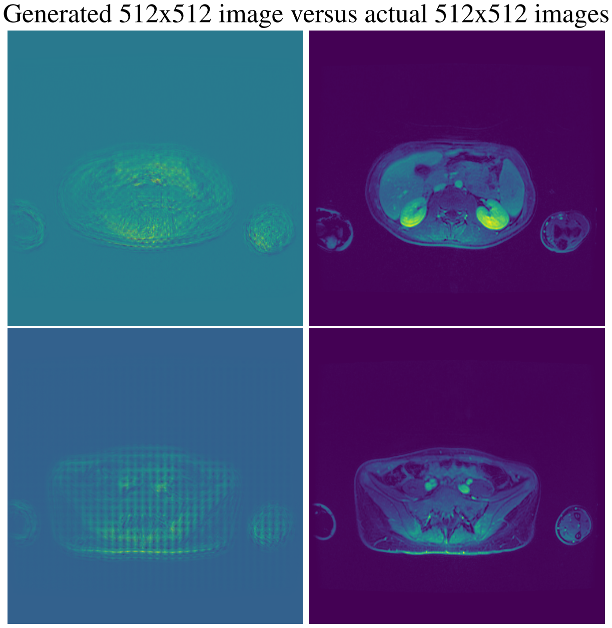
Our upsample factor was 2, so we were conducting super-resolution from a 256x256 image to a 512x512 image. We also implemented dropout regularization on our discriminator to help with the large contrast between the discriminator's training and the generator's training.

6 Experiments

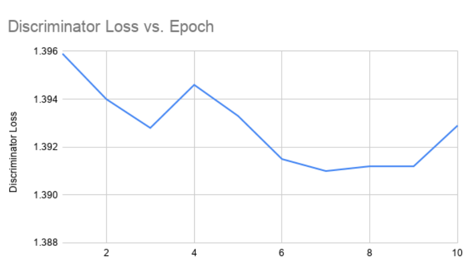
We used a generator learning rate of 0.002, a discriminator learning rate of 0.0002, an Adam optimizer for the generator, and a stochastic gradient optimizer for the discriminator. Online sources disagree over whether having different learning rates for the generator and discriminator is acceptable, but we found this to be a band-aid for the problem of our generator loss being higher than our discriminator loss. We initially tried using an Adam optimizer for the discriminator, but the discriminator was too good and caused the generator to be unable to train. In fact, we still ran into similar problems with vanishing gradients, but making this change significantly improved our results. Another change we made was to have the generator take four steps for every one that the discriminator took, which also helped us reduce our generator loss. We used a batch size of 8 for our roughly 13,000 images (approximately 1,300 per patient), in large part because our GPU was unable to store any larger batch size.

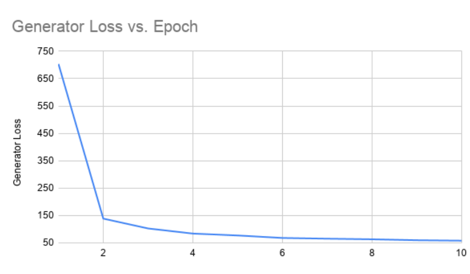
The below results come from training our model over 10 epochs on our dataset.

Generated 512x512 image versus actual 512x512 images



On training, our discriminator and generator losses decreased as follows:





Our generator loss seems to plateau around 50, which is higher than what is optimal, but it is already a significant improvement over previous iterations of our model.

7 Conclusion

While our results show a lot of promise, they are still quite pixelated. We ran into problems where our loss while testing was 10 to 20 times higher than our losses in training. The resulting images were not as high-quality as expected, as we experienced both vanishing and exploding gradients as well as a discriminator that was too good for our generator. Further avenues for investigation would be to change our loss function to a Wasserstein loss or another suitable loss function, training for more epochs than we had the computational capability to do, etc. This architecture, however, shows a lot of promise and solving for our plateauing loss will go a long ways to improve the imaging quality.

8 Contributions

We all worked on this project jointly, contributing equally, and would like to thank Dr. Heike Daldrup-Link, Dr. Joyce Wang, and Dr. Michael Zhang of Stanford School of Medicine for providing us the data we used in this project as well as our TA Shubhang Desai for this help. Our code is located in the Gradescope submission link for the final project.

9 References

[1] Wang, Jiancong, et al. “Enhanced Generative Adversarial Network for 3D Brain MRI Super-Resolution.” ArXiv, 10 July 2019.

[2] https://github.com/alvayliu/Fashion-GAN

[3] https://github.com/wiseodd/generative-models/blob/master/GAN

[4] https://www.nationalmssociety.org/Symptoms-Diagnosis/Diagnosing-Tools/MRI

[5] Ghodrati, Vahid, et al. "MR image reconstruction using deep learning: evaluation of network structure and loss functions." Quantitative Imaging in Medicine and Surgery [Online], 9.9 (2019): 1516-1527. Web. 26 May. 2020

[6] Ledig, Christian, et al. "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network." ArXiv, 15 September 2016.

[7] Chen, Yuhua, et al. "MRI Super-Resolution with GAN and 3D Multi-Level DenseNet: Smaller, Faster, and Better." ArXiv, 2 March 2020.

[8] Qing Lyu, Chenyu You, Hongming Shan, Yi Zhang, and Ge Wang "Super-resolution MRI and CT through GAN-CIRCLE", Proc. SPIE 11113, Developments in X-Ray Tomography XII, 111130X (10 September 2019); https://doi.org/10.1117/12.2530592

[9] Wang, Yun-Heng, et al. “Sparse Representation-Based MRI Super-Resolution Reconstruction.” Measurement, vol. 47, 2014, pp. 946–953., doi:10.1016/j.measurement.2013.10.026.

[10] Noguchi, et al. “Image Generation From Small Datasets via Batch Statistics Adaptation.” ArXiv.org, 23 Oct. 2019, arxiv.org/abs/1904.01774.