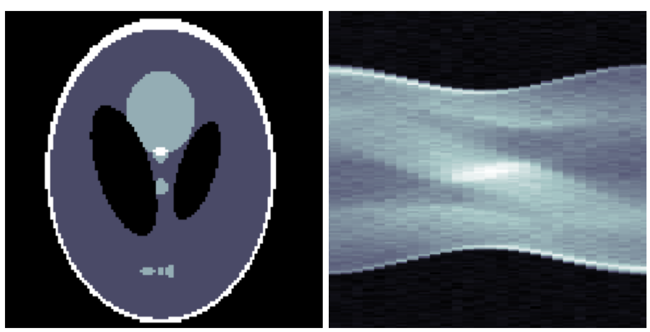
ECE 228 - Milestone Report - Team 22

Chris Light
UCSD - ECE
cdlight@ucsd.edu

Michael Ingerman
UCSD - MAE
mingerman@ucsd.edu

1 Introduction

Computed Tomography is one of the most prevalent forms of medical imaging available. Albeit invaluable to doctors in medical decision making, CT scanning still holds a risk to patients as the use of harmful X-ray radiation poses health concerns. There is a delicate balance of not exposing the patient with too much radiation while still producing a meaningful picture. Traditional signal processing methods in CT imaging is bounded by the Nyquist rate, which is a lower limit to the number of projection angles necessary to perform image construction. Reducing the number of angles necessary for a meaningful image means less radiation exposure - and a necessity to break the Nyquist criteria by use of a Deep Neural Network.
In this paper we adapt a method called Deep Back Projection [1] for the purposes of reconstructing CT chest images of Covid19 patients using sparse imaging with a deep convolution neural network (CNN) framework. The advantages of using a CNN is that it reduces the number of parameters of deep neural network by imposing spatial invariance on image data. Using the Harvard's "COVID19-CT-Dataset"[2] we first process each CT image by producing a set of back projected sinogram slices to form a stack of images of parallel lines at different orientations. We then feed this stack as input into our model during training in an attempt to employ learning to compute a fully reconstructed sinogram.



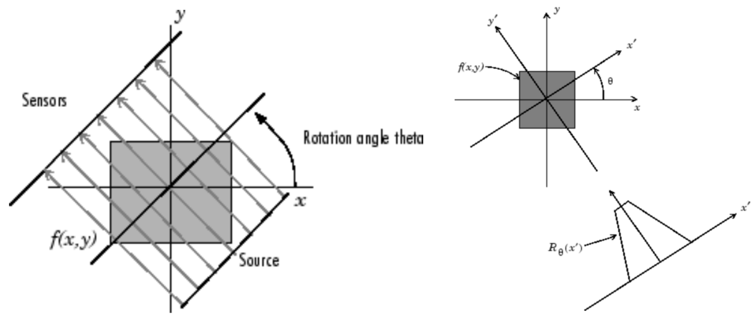
(a) Shepp-Logan phanton (b) Shepp-Logan sinogram

Figure 1: Left image is a Shepp-Logan Phantom. Right image is a Corresponding sinogram.

2 Background

The Radon transform function computes the projections of an image matrix along a specified direction. For a 2D image, the projections of the image data found via radon transform can be considered a set of line integrals of pixel intensities along parallel paths as can seen in fig. 2. By sampling axes at various orientations, a set of random transforms can be combined to form a sinogram image like in fig 1. Furthermore, for each orientation of sinogram projection computed via radon transform, the signal intensities for each projection can be cast as parallel lines of uniform intensities along the direction

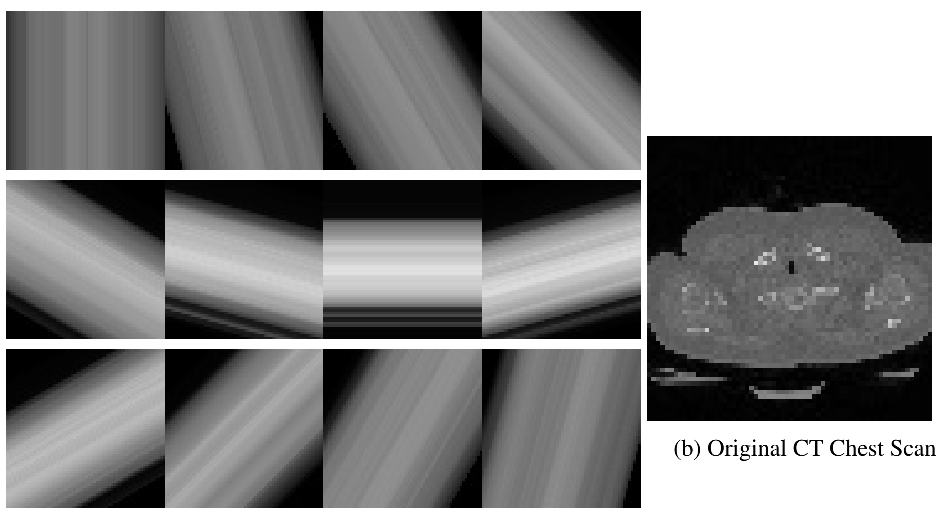
Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.



(a) Parallel-Beam Projection (b) Radon Transform on axis

Figure 2: Left image showcases how parallel signals are sent at a given orientation of a source. Right image shows how signals are interpreted through Radon Transform to yield the integral intensities of a signals along an axis.

of the original signal as can be seen in fig. 3. The resulting images with parallel lines of uniform intensities are called back-projected slice sinogram images. Collectively, these back-projected images are useful since they represent spatial-based information in a non-spatially dependent way. This makes back-project images a convenient input for a convolution neural network style model.



(b) Original CT Chest Scan

(a) 12 Generated Sinogram Back Projections

Figure 3: Radon transform is applied to a CT chest scan(truth image) using 12 angles to create a CT sinogram with fixed step size between $0° - 180°$. Single-view back projections are stacked to form the input for the convolution neural network.

3 Dataset and Framework

3.1 Dataset Processing

Truth images for our model were gathered from Harvard's "COVID19-CT-Dataset"[2]. Each truth image was converted to a set of sinogram projections at different orientations θ_i. Every set of sinogram projection images y_i was found by first computing a sinogram y from a given truth image with a set of n angles. The number of slices n corresponding to the range of angles was variably selected in order to gauge how the model preformed as a function of data sparsity. Then each set of sinogram projection $y_i \in R^n$ were generated by extracting projections along the set of angles $\theta_i \in R^n$ for a given sinogram image y.

All datasets used to obtain current results contained 568 truth images with 568×12 sinogram slice projections. Future model training will be done with a dataset of $200,000$ truth images and a variable number of slice projections. No image preprocessing was applied to truth images in our initial attempts with the exception of image downsizing 128×128 pixels from the original truth image size of 512×512 pixels. The rational for doing so was to observe how well our model dealt with minimally prepossessed data.

3.2 Framework Architecture

Following the work of [3], we construct a similar architecture to what is seen in fig. 4

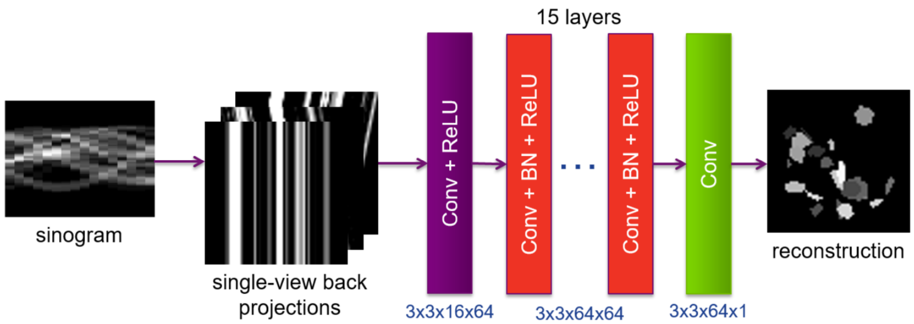


Figure 4: Network Architecture

From the dataloader, a torch tensor of size [b,n+1,h,w] is fed into the network; For b = batch size, n = number of angles, (h,w) = dimension of the pictures. The first layer of the model conforms to n input channels (the number of slice projections) and 64 output channels with ReLU activation. 15 layers of Conv > batch normalization > ReLU with a consistent 64 channels follows and finally compressed into one channel at the last layer as the resulting image is produced. The kernel size is consistent at (3,3) and padding is set to preserve the original CT image resolution.

4 Initial Results

At the project's current state, we have successfully managed to perform training and produce some observable results in fig 5.

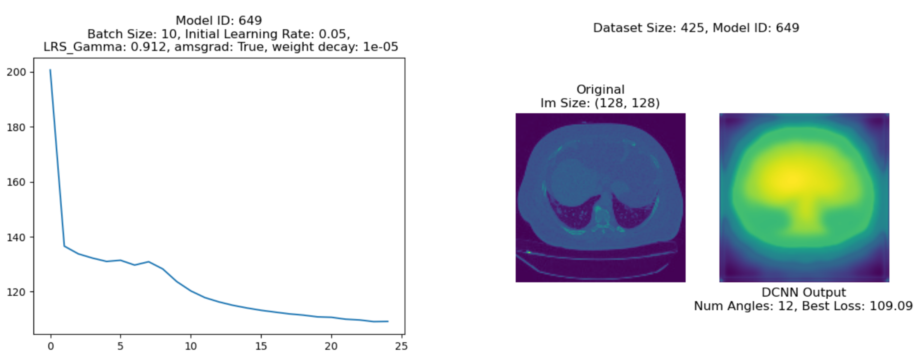


Figure 5: Initial Results

These results are without any hyperparameter tuning, model reconfigurations, or data preprocessing. One observation in the loss plot is we don't have validation testing set up yet.
For reference, we are using a batch size of 10, learning rate 5e-2, weight decay 1e-5, and MSE loss. The optimizer is AMSGrad, which is an extension of the Adam Optimizer. We are also using ExponentialLR scheduler with gamma = .912.

5 Future Improvements

5.1 Improved Model/training

We will consider changes to the model, however this might not be the forefront of our considerations since [3] is a testament to the model's optimality for our purpose. We will be looking more at the hyperparameters such as learning rate, weight decay, batch size. We might consider other learning rate schedulers aswell as a different optimizer, however the Adams optimizer is probably already good enough.

5.2 Improved Data Preprocessing

An observation of our experimental results showcased a lack of confidence in our models reconstruction of truth images. While the main culprit of this is likely due to dataset size, we theorize that preprocessing methods may improve learning of our models. Therefore future dataset will be preprocessed to improve contrast using methods such as Contrast Limited Adaptive Histogram Equalization from the OpenCV library. By improving the contrast in truth images, we hope to imbue our models with higher confidence for image reconstruction.

References

[1] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, 37(6):1322–1332, 2018.

[2] Sayyed Mostafa Mostafavi. COVID19-CT-Dataset: An Open-Access Chest CT Image Repository of 1000+ Patients with Confirmed COVID-19 Diagnosis, 2021.

[3] Dong Hye Ye, Gregery T. Buzzard, Max Ruby, and Charles A. Bouman. Deep back projection for sparse-view ct reconstruction, 2018.