MGE51101 Final Project
Low dose CT denoising with Deep Learning

Iskakov Alisher

June 20, 2020

Abstract

CT scans expose patients to large amounts of radiation, thus it is important to reduce the X-ray dose. However, reducing dose leads to more noise. There are different methods for denoising CT scans, and deep learning methods have been showing promising results. In this project, I apply deep convolutional neural network to the problem denoising using a publicly available dataset of CT images and evaluate the results based on several metrics.

1 Introduction

The goal of this project is to experiment with deep neural network applied to denoising of computed tomography (CT) scans. I want to compare the performance of deep learning models with traditional methods and evaluate them based on some set of metrics.

CT is an X-ray based imaging technique, which is widely used in clinical practice. Compared with plane radiography, CT scanner produces multiple projections that are then reconstructed with specific algorithms. This results in a large exposure to radiation by patient, which can increase a risk of cancer and limits the number of examinations. In the recent decades there has been an active research in reducing radiation dose in CT. However, as we reduce the dose the quality of the images also degrades. This is called quantum noise, which happens because there are less photons reaching the detector and it leads to weakening of the signal.

It is important to understand how CT images are formed. In CT, multiple projections of an object are taken, which are then used to create cross-sectional images. This is called back-projection, and it is performed using Radon transform. Radon transform takes projections data from so-called sinogram domain and transforms it into image domain. The algorithm is very fast, and it can also remove some noise if filtering is applied. However, there are different algorithms which utilize the prior information about physical details and iteratively improve the quality of images. These are called iterative reconstruction (IR) algorithms, and they are usually taking more computation and thus requires more time.

With the recent success of CNNs applied in image recognition, they started to gain popularity in other tasks too, particularly in medical imaging.

2 Problem description

The data I am using to train the networks is provided by a hospital, which we are collaborating with on this project. Due to this, I cannot provide sample images from the dataset. The samples are given in .jpg format, so it is convenient to handle them. However, CT scans are usually provided in DICOM format and the values are not normalized in [0, 1] range. The values are given in Hounsfield unit, which shows a relative coefficient of X-ray attenuation of given pixel.

The images are of size 512x512 pixels, but only 400 central pixels are most important to us, since they contain the subject of interest. Therefore, I apply center cropping when preprocessing the images. Each data sample is given as a pair: (low dose image, standard dose image). The goal is to find the correspondence between these two images. Low dose image has been acquired at a lower dose, correspondigly, and reconstructed with simple filtered backprojection algorithm (FBP). Standard dose image used standard dose and was reconstructed with a better IR algorithm. When visually inspecting, there are visible noise patterns in the low dose image, and this noise can hide some textures in CT scans.

There are 5152 image pairs used for training the models. After each epoch, I validate the models on 106 image pairs which were randomly selected and not used in training. I recorded validation results on each epoch, and then selected best results as representative for a given model. I did not use a test set though, since validation data did not impact model parameters in any way, and it could be viewed as an independent subset of data.

2.1 Evaluation

To evaluate the performance of the models, I have used 4 metrics.

- Mean-squared error (MSE) - pixel-wise difference between two images
- Root mean-squared error (RMSE) - square root of MSE; shows the standard deviation of MSE
- Peak signal-to-noise ratio - shows how strong is noise in the low dose image
- Structural similarity index (SSIM) - perceptual metric which shows the difference in luminance, contrast and structure of low dose image, when compared to reference image (standard dose image). SSIM is more reflective of perceptual similarity (as human perceives quality degradation) than MSE.

3 Method

First of all, I decided to conduct a literature review to see what architectures have already been applied to this problem. The first idea was to re-implement the architecture described in [1]. It contains 5 convolutional and 5 deconvolutional layers with 96 feature maps in each. The important part of this architecture is the utilization of skip-connections between corresponding convolutional and deconvolutional layers. This improves training as it acts as a residual connection. Moreover, it helps recover spatial information which could be lost in the encoder.

After that, I decided to try a simpler autoencoder model but with a different loss function. Autoencoders are very suitable for this task due to the symmetric nature of the architecture. When the input is given to a network, the encoder learns a representation of noise and the decoder then reconstructs this image. My implementation has only 7 convolutional and 8 deconvolutional layers, but it achieves the same results as the architectures proposed by [1].

However, I have also changed the loss function used in the training. Usually, in image restoration tasks L_2 loss is used, which calculates pixel-wise difference between images and tries to minimize it. However, several approaches used perceptual loss which was generated by the output of a pretrained VGG network [2]. For example, the authors in [3] used only such VGG features and achieved good results. In my approach, I combined L_2 loss with perceptual VGG loss.

4 Experiments

I trained both of the models for 100 epochs and conducted validation after each epoch. Loss functions were optimized with Adam optimizer [4]. For each of the networks, I set learning rate to 0.0001, and the betas were 0.5 and 0.999, respectively.

During each epoch, I recorded validation results (for each metric) and then selected the best performing epoch. The evaluation metrics are given in Table 1. The first row shows the metrics for a low dose image (reconstructed with FBP algorithm) and standard dose image (reconstructed with IR algorithm). RED-CNN [1] showed the best results on validation set, but autoencoder model was pretty close in quantitative comparison. In addition, the visual results of both models were also very similar.

Model	MSE	RMSE	PSNR	SSIM
Low-dose vs standard-dose	0.00255	0.04878	26.45900	0.84892
RED-CNN [1]	0.00110	0.03205	30.08272	0.92181
Autoencoder	0.00113	0.03293	29.80322	0.91768

Table 1: Validation metrics

The loss curves for the training and validation of both models are presented below. It can be seen, that the loss becomes quite small very fast (almost from the start) and then decreases more in a steady manner. This can be an indication that models ”view” the problem as easy, but it also can mean that there is more room for improvement.

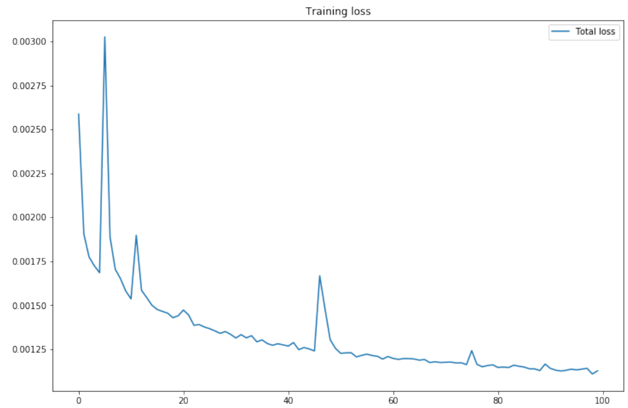

Figure 1: Training loss of RED-CNN

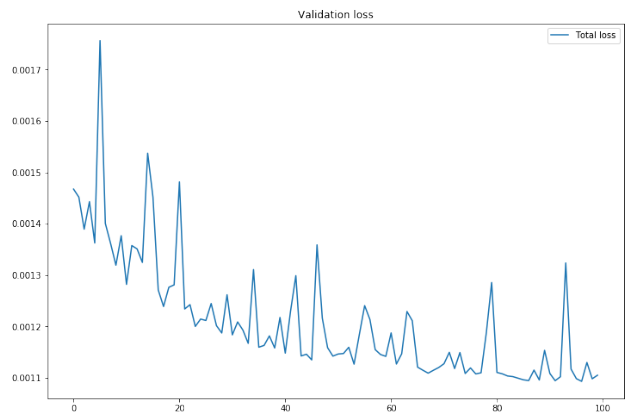

Figure 2: Validation loss of RED-CNN

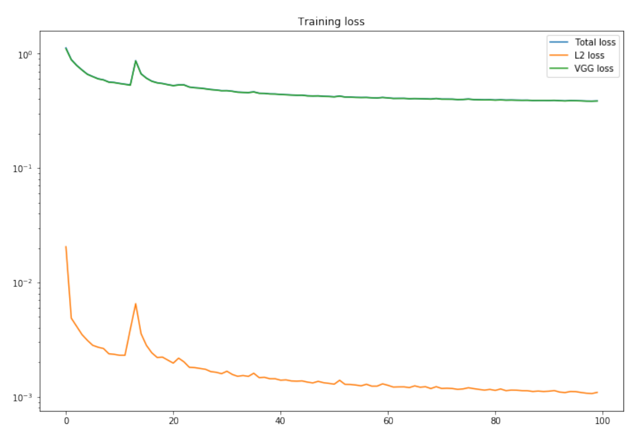

Figure 3: Training loss of autoencoder (logarithmic scale)

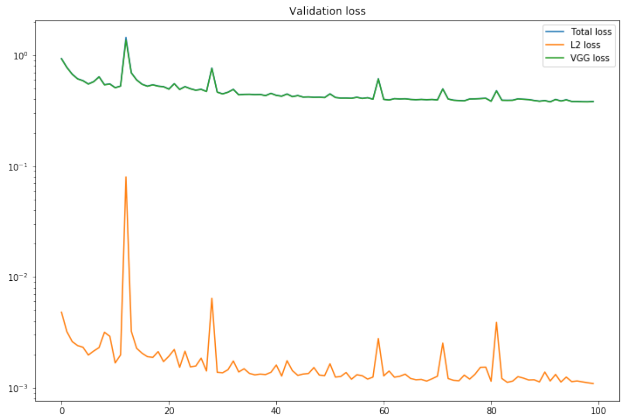

Figure 4: Validation loss of autoencoder (logarithmic scale)

Despite that I experimented with only two architectures, I have found several directions for future improvement. For example, according to [5], SSIM can be also used as a loss function (in combination with L_1 loss to produce better quality images. In addition, it is always possible to go deeper with the current architectures. This could be not a best option but it is also important to experiment in that direction. Adding a discriminator (as in GANs) could also improve the performance of denoising network, though it may be hard to properly tune and train a GAN architecture.

References

[1] Hu Chen, Yi Zhang, Mannudeep K Kalra, Feng Lin, Yang Chen, Peixi Liao, Jiliu Zhou, and Ge Wang. Low-dose ct with a residual encoder-decoder convolutional neural network. *IEEE transactions on medical imaging*, 36(12):2524–2535, 2017.

[2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[3] Qingsong Yang, Pingkun Yan, Mannudeep K Kalra, and Ge Wang. Ct image denoising with perceptive deep neural networks. *arXiv preprint arXiv:1702.07019*, 2017.

[4] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[5] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on computational imaging*, 3(1):47–57, 2016.