Final Project Report: Low-Dose CT Image Denoising

Joel Mbouwe[1], Abderrahim Mehdaoui[1], and Adnan Zeddoun[1]

CentraleSupélec, Gif sur Yvette 91190, France
{joel.mbouwe,abderrahim.mehdaoui,adnan.zeddoun}@student.ecp.fr

Keywords: Low Dose Computed Tomography (LDCT) · Image Denoising · GANs

1 Abstract

Computed tomography methods have continuously improved following the growing of technology. With it, a public concern about the radiation dose administrated to the patient has raised. In fact, it could lead to serious medical damages if the patient is too much exposed. However, reducing the dose could bring some noise and artifacts on the image impacting the radiologists' judgment and confidence at the same time. So, the development of advanced reconstruction methods from low-dose computed tomography (LDCT) images seems to be a good approach to solve this problem. In fact, it has bring impressive results. However, most of these methods are based on the minimisation of the MSE (Mean Square Error) between two images which can compromise the visibility of important structural details after aggressive denoising. We propose here to implement and try to improve a CT image denoising method proposed by [1] which is based on generative adversarial networks associated to the Wasserstein distance and perceptual similarity. This methods tries to take into account the preservation of the the critical information present in a CT-image.

2 Introduction

X-ray computed tomography (CT) is one of the most important imaging method used in modern hospitals and clinics. But, there is a public concern regarding the radiation dose administrated to the patient, since X-rays could cause genetic damage and induce cancer in a proportion related to the radiation dose. So, it appears that reducing the radiation dose may increase the noise and artifacts in reconstructed images, which can impact diagnostic information and the non-detection of some pathologies as well. Hence, extensive efforts have been made to design better image reconstruction or image processing methods for LDCT.

The recent hype produced by deep neural networks brings new ways of thinking and huge potential for the field of medical imaging. However, these networks usually try to minimise the MSE between the network output and the ground truth. However, according to [1], the use of the MSE as a loss function is usually associated to blurring effects, artifacts and a lose of details in the final results. It can lead to a lose of critical information that can impact the diagnostic. To tackle the above problems, [1] proposes to use a generative adversarial network (WGAN) associated to the Wasserstein distance as the discrepancy measure between distributions and a perceptual loss. The perceptual loss computes the difference between images in an established feature space extract from the well known VGG19 [4] to compare the denoised output against the ground truth. We propose here to implement their method and try to show the relevance of their loss function.

The report is organized as follows: Section 3 gives a more formal and shorter definition of the problem, section 4 describes related works about medical images denoising using deep learning methods, Section 5 details the general methodology we followed, Section 6 explains how we evaluate our works and some experimental results, then Section 7 presents the final conclusions.

3 Motivation and Problem Definition

Here we seek to study a new computed tomography (CT) image denoising method based on the generative adversarial network (GAN) with a loss based on perceptual similarity. X-ray (CT) is one of the most important imaging modalities in modern hospitals and clinics. However, considering that X-rays could cause genetic damage and induce cancer, there is a potential radiation risk to the patient, in a proportion related to the radiation dose. So, the problem we aim to study has major concerns regarding the safety of the patients, since improving the dose might lead to future medical issues. However, with lower dose, we might increase the noise and artifacts in reconstructed images, which can compromise diagnostic information that might lead to medical complications for the patient.

Here we aim to use the method proposed by [1] to study the denoising on low dose computed tomography (LDCT).

4 Related works

Finding a good paired dataset easily accessible to perform denoising is not easy. To tackle this issue, [7] proposes a two-step framework that aims at removing any type of noise. First, a GAN is used to estimate the noise distribution over the input to generate noise samples. Then, the sampled patches are used to construct a paired training dataset. Which will be then used to train a CNN so as to perform denoising. [5] proposes also denoising method for NDCT images in the absence of paired training data using GANs. It aims at minimizing a weighted sum of two losses: the Kullback-Leibler divergence between an NDCT data distribution and a generated distribution, and the L_2 loss between the LDCT image and the corresponding generated images (or denoised image). It appears to perform pretty well compared to methods that use paired datasets.

[6] has addressed the problem of denoising in a more general way by using deep convolutional generative adversarial networks (DCGAN) as a uniform architecture to perform image processing tasks including of course denoising. Their architecture gives competitive results and can generate images that are more appealing compared to conventional methods.

5 Methodology

To denoise LDCT Images, we will use a generative adversarial network WGAN with the Wasserstein distance [1] as the discrepancy measure between the real data distribution and the generated data distribution and a perceptual loss that computes the difference between images in an established feature space. To do so, we based our work on this Github repository. [2]

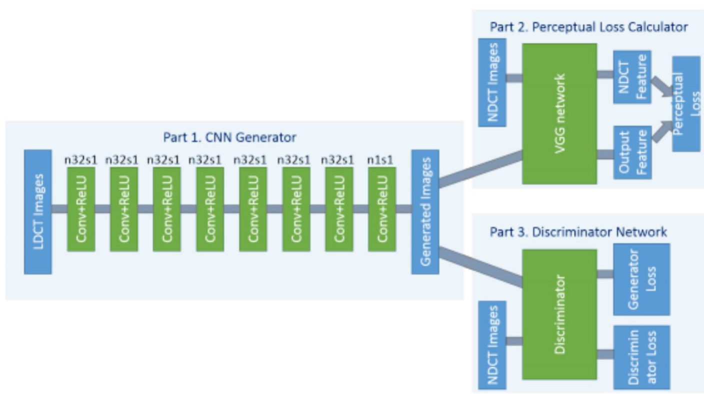

Fig. 1. The overall structure of the proposed WGAN-VGG network. In Part 1, n stands for the number of convolutional kernels and s for convolutional stride. So, n 32s 1 means the convolutional layer has 32 kernels with stride 1.

5.1 Noise Reduction Model

We define a Noise Reduction Model **G** with a CNN network. The goal of the denoising process is to seek a function G that maps LDCT image z to NDCT (Normal Dose Computed Tomography) image x. G will act as the generator of our WGAN model. **G** is composed of 8 convolutional layers and each layer is followed by a Rectified Linear Unit (ReLU) as the activation function.

5.2 Discriminator

The discriminator network **D** aims to distinguish candidates produced by the generator from the true data distribution. **D** is modeled by a CNN network composed of 6 convolutional layers. The activation functions here are Leaky Rectified Linear Unit (LeakyReLu).

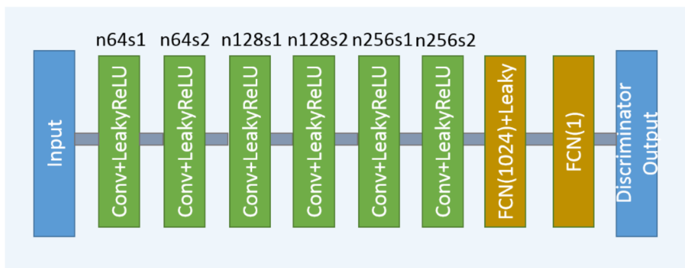

Fig. 2. The discriminator: n 64s 1 means the convolutional layer has 64 kernels with stride 1.

5.3 Objective functions

The overall goal of a GAN network is that the generator produces candidates that the discriminator cannot tell the difference with a real candidate. For doing so, **G** & **D** play a minimax game in which **G** try to generate candidates $G(z)$ so that $D(G(z)) = 1$ and on the contrary **D** tries to outputs $D(G(z)) = 0$ and $D(x) = 1$ with x a real candidate. The WGAN [11] model solves this game by solving the optimization problem :

$$\begin{aligned}\min_G \max_D L_{\mathrm{WGAN}}(D, G) &= -E_{\boldsymbol{x}}[D(\boldsymbol{x})] + E_{\boldsymbol{z}}[D(G(\boldsymbol{z}))] \\ &\quad + \lambda E_{\hat{\boldsymbol{x}}}\left[\left(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1\right)^2\right]\end{aligned}$$

The first two terms perform a Wasserstein distance estimation between the distribution $D(x)$ (or $D(G(z))$) and the distribution of the real data. The last term called gradient penalty plays the role of the regulation parameter of the discriminator and $\hat{\boldsymbol{x}}$ is uniformly sampled along straight lines connecting pairs of generated and real samples.

The article [1] proposes to add another term to the loss function in order to learn and keep details on the images : **The perceptual loss**. They proposed to use the VGG [4] to extract features on both generated and real images and compute the *Frobenius norm* between those features extracted. The result will be add in the previous loss function. They thought about this perception because the MSE loss can lead the generator to produce smooth or blurry images. The perceptual loss is defined by

$$L_{\text{VGG}}(G) = E(\boldsymbol{x}, \boldsymbol{z}) \left[\frac{1}{whd} \| VGG(G(\boldsymbol{z})) - VGG(\boldsymbol{x}) \|_F^2 \right]$$

The overall loss function is therefore :

$$\min_G \max_D L_{\text{WGAN}}(D, G) + \lambda_1 L_{\text{VGG}}(G)$$

with λ_1 and λ two constants. The training process consists in two steps where at each step either **D** or **G** will be updated, the other being idle :

- **Discriminator**: **D** is trained on real data for a few epochs, and see if it can correctly predict them as real. **D**is also trained on the fake generated data from the Generator and see if it can correctly predict them as fake. The loss function associated is :

$$\lambda E_{\hat{\boldsymbol{x}}} \left[\left(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1 \right)^2 \right] - E_z \left[D(x) \right] + E_z \left[D(G(z)) \right]$$

- **Generator**: The previous output of the discriminator on the generated image is used for training the Generator and get better results from the previous state to try and fool the Discriminator. The loss function is

$$\lambda_1 L_{\text{VGG}}(G) - E_z \left[D(G(z)) \right]$$

We train three others networks to estimate the real impacts of WGAN and perceptual loss on a denoising problems. The summary of the trained models are in the table 1

Network	Loss
Generator-MSE	$\min_G L_{\text{MSE}}(G)$
WGAN-MSE	$\min_G \max_G L_{\text{WGAN}}(G, D) + \lambda_1 L_{\text{MSE}}(G)$
Generator-VGG	$\min_G L_{\text{VGG}}(G)$
WGAN-VGG	$\min_G \max_G L_{\text{WGAN}}(G, D) + \lambda_1 L_{\text{VGG}}(G)$

Table 1. All tested models

6 Experiments and analysis of the results

6.1 Dataset

We used a Benchmark Dataset for Low-Dose CT Reconstruction Methods [9]. In total, the dataset contains 35 820 training images, 3522 validation images, 3553 test images. Each part contains scans from a distinct set of patients as we want to study the case of learned reconstructors being applied to patients that are not known from training.

However since we are limited by computing capacity we only used a subset of these images for our project.

We train our different models with 6272 (corresponding to 50 distinct patients) couple of noisy images and real images. The noisy images were computed by adding a centered gaussian noise with a variance of 0.015 to the real images (the range of our images pixel value was $[0, 1]$).

Furthermore, since we do not have hyper parameters to optimize we do not use a validation set. We test our models on 640 images (corresponding to 4 distinct patients). Figure 3 shows a sample couple noisy image and real image.

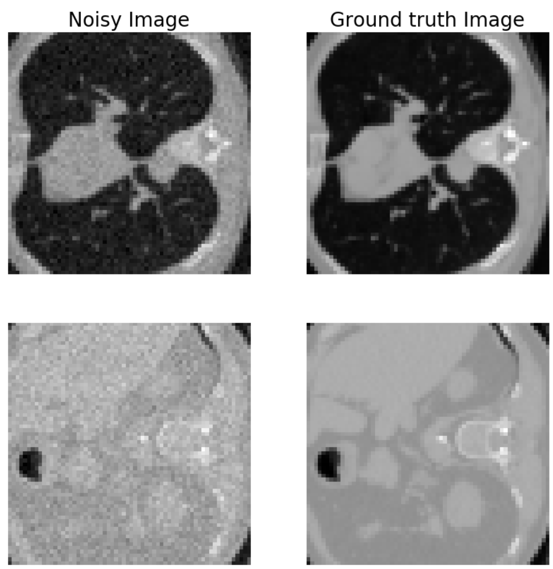

Fig. 3. Sample : at left the noisy image and at right corresponding real image

Further details on the dataset can be found in [3].

6.2 Training

The inputs image were all downsized to the shape (64 x 64) and the batch size when training was set to 32. All the networks we tested were optimized using Adam algorithm with a learning rate of 10^{-5} for the discriminator and 10^{-4} for the generator. we also use $\lambda = 10$ and $\lambda_1 = 0.1$ like in the original paper. We implemented the whole code ourselves and here [10] can be found the details of our implementations.

The training process for 100 epochs took about 6 hours for WGAN-VGG and WGAN-MSE and about 1h for the two remaining models.

Note : We also noticed an important remark : the initialisation of the weights of the Generator when training the WGAN-WGG plays an important role. Indeed we found out that if the weights are not properly initialize in the sense that for the first batch data, the pixels of $G(z)$ have values close to zero, then the Generator will take a lot of epochs to start output better reconstructed images and thus impacts negatively the convergence process. We could have decide to not randomly initialize the weights but instead we just reset the generator till obtaining acceptable $G(z)$ for the first batch data.

6.3 Convergence

To estimate the convergence of the models we trained, we stored the evolution over the epochs of the different parts of our loss function computed on the test set. Figure 4 shows the results.

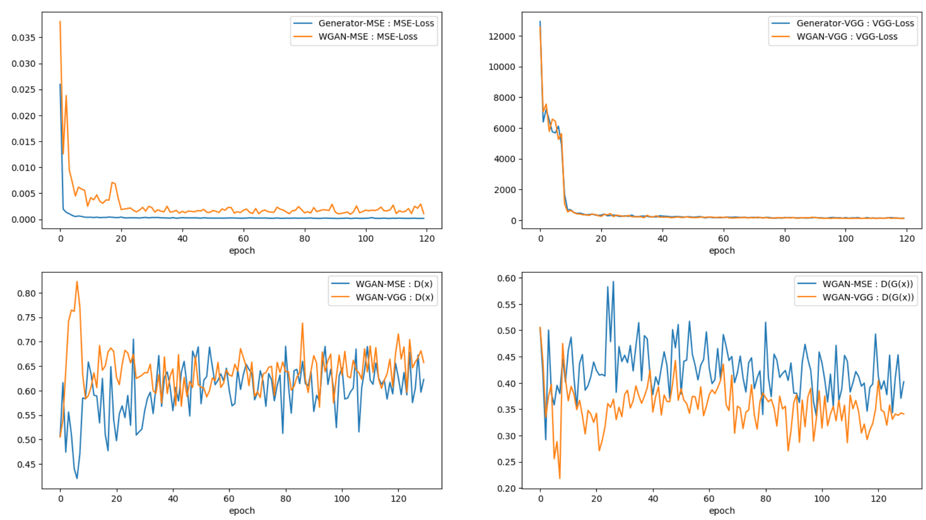

Fig. 4. All loss functions. (top left) : MSE loss for the Generator-MSE and WGAN-MSE. (top right) : VGG loss for the Generator-VGG and WGAN-VGG. (bottom left) : $D(x)$ for the models with WGAN. (bottom left) :D(G(Z)) for the models with WGAN.

Top left and *Top right* showed the decreasing of the MSE and the VGG over the epochs meaning that the training process is converging in terms of reconstruction loss. The Generator-MSE's mse loss is lower than the WGAN-MSE and on the other side the VGG loss of the the Generator-VGG and WGAN-VGG are quite close. We can also noticed that using WGAN creates oscillations in the MSE and VGG losses. Such observations imply that the two losses are not treated similarly by the networks. The bottom images showed that the Wasserstein distance related to the real images and the generated images stabilize around certain values.

Remark : Those losses values were similar to the ones computes on the training set meaning that there was strictly no overfitting in our models.

6.4 Visualisation of the denoised images

Figure 5 shows a representation of the original, noisy and denoised images for all the models we trained. We select this image on our testing set for this visualization and if our attention has been drawn to this image it is because it presents a lot of details.

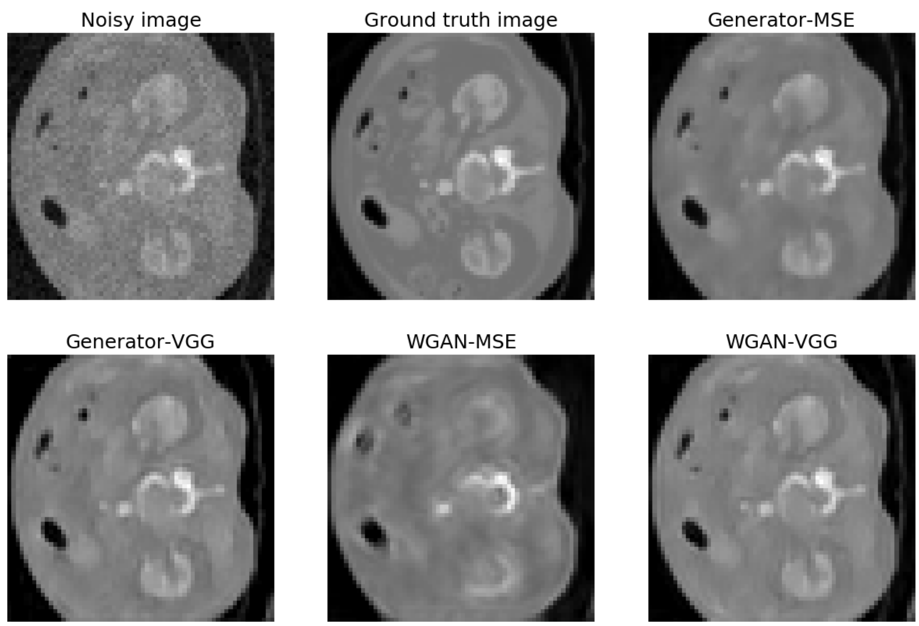

Fig. 5. Noisy, Ground truth and denoised images

The first thing we noticed is the poor visual performance of the WGAN-MSE. The Generator-MSE tends to smooth the reconstructed pixel and this is where the impact of the VGG appears : the MSE loss is not able to reconstruct enough well all the details while, on the contrary, Generator-VGG and WGAN-VGG images are visually closer to the ground truth image. This is explained since the VGG loss is a MSE loss computed in a feature space that is trained previously on a very large natural image dataset. By using VGG loss, we transferred the knowledge of human perception that is embedded in VGG network to CT image quality evaluation. As it is said in [1].

Table 2 gives the values of the PSNR and the SSIM on the previous images. The best values are the ones obtained with the WGAN-VGG. It is important to noticed that the values of the metrics for the Generator-MSE and the WGAN-VGG are quite the same despite the fact that we can visually observe differences especially in the artifacts.

	SSIM	PSNR
Generator-MSE	0.996293	37.213245
Generator-VGG	0.994949	35.088562
WGAN-MSE	0.990097	32.812336
WGAN-VGG	**0.996800**	**37.692360**

Table 2. Metrics values on the sample image

6.5 Metrics

Along with the qualitative analysis we made in the previous section, we have also estimated quantitatively the results of our models. We compute the *Peak to noise ratio* and the *Structural Similarity* for each batch of our test set and we record the mean values of both metrics. Table 2 summarize the results.

	SSIM	PSNR
Generator-MSE	**0.9967**	**37.871**
Generator-VGG	0.9901	33.325
WGAN-MSE	0.9856	31.521
WGAN-VGG	**0.9941**	**35.654**

Table 3. Statistics of the metrics on the test set

Generator-MSE appears to give the best noise suppression score which is quite logic since the PSNR is computed using the MSE which is the loss optimized when training the Generator-MSE. WGAN-VGG has the second highest scores and as we have seen in the previous part it outputs better visual performance than the Generator-MSE. Indeed it reconstructs better the artifact parts of the images.

The last figure [6] shows the evolution of the PSNR and the SSIM computed on the training set over the epochs. These values converge fairly quickly.

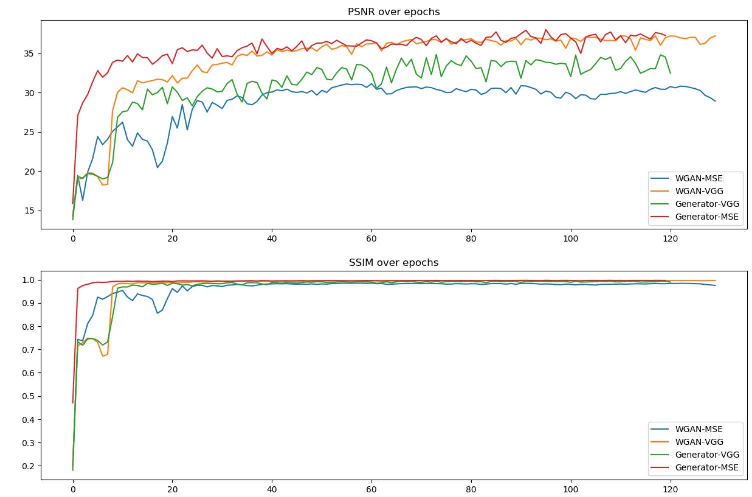

Fig. 6. Evolution of the metrics over the epochs

7 Conclusion and Perspectives

In this short project, we addressed the problem of image reconstruction or image processing methods for LDCT images. We based our study on the previous experiments on Wasserstein Generative Adversarial Network and we take in account a perceptual loss computed using the well know VGG model as a feature extractor. We trained several models and compared qualitatively and quantitatively their performances. It appears at the end of this study that considering only a CNN-network and minimising its MSE reconstruction error gives interesting results in terms of the traditional denoising metrics. But MSE tends to smooth the images and thus leads to the loose of details. WGAN-VGG gives traditional denoising metrics values close to the ones of the Generator-MSE and perform better than it in terms of qualitative aspects and keeping details. We also showed that Generator-VGG and WGAN-VGG visually share a similar result but the quantitative analysis shows that WGAN-VGG gives better PSNR and SSIM scores.

We wanted at first to work with real world paired dataset of LDCT and NDCT but we couldn't find one in the required time for the project. Adding a noise on the NDCT allows us to simulate LDCT and to test the algorithm mentioned in [1]. We faced a lot of issue since we were limited by our computational power and Google Colab didn't help us since our kernel was disconnecting after a long period of training... Anyway we did found a way and we were able to process the training of the models.

In summary, it was a really interesting project that made us address an important problem in the medical imaging field by the use of Deep Learning. We really really learned a lot of things doing this.

References

1. Yang et al: Low-Dose CT Image Denoising Using a Generative Adversarial Network With Wasserstein Distance and Perceptual Loss. IEEE Transactions on medical imaging, vol. 37, 1348–1357 (2018).
2. Yang's Github Repository of the Low-Dose CT project, https://github.com/hyeongyuy/CT-WGAN_VGG_tensorflow.
3. Leuschner et al: The LoDoPaB-CT Dataset. A Benchmark Dataset for Low-Dose CT Reconstruction Methods. arXiv:1910.01113 (2019).
4. Karen Simonyan & Andrew Zisserman : VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE SCALE IMAGE RECOGNITION https://arxiv.org/pdf/1409.1556.pdf
5. Park et al: Unpaired image denoising using a generative adversarial network in X-ray CT,IEEE Access, 2019.
6. Wang et al: DCGANs for image super-resolution, denoising and debluring, 2017.
7. Chen et al: Image Blind Denoising With Generative Adversarial Network Based Noise Modeling, CVPR, 2018.
8. Simonyan et al: Very deep convolutional networks for large-scale image recognition. https://arxiv.org/abs/1409.1556 (2014)
9. Datasets of LDCT Images, https://zenodo.org/record/3384092.
10. https://github.com/Ryosaeba8/DLMI.
11. M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017.[Online]. Available: arXiv:1701.07875.