המכללה האקדמית להנדסה בכרמיאל
אורט בראודה

Capstone Project Phase A

Denoising using Noise-to-Noise

(22-1-R-7)

Supervisor:

Miri Weiss Cohen

Submitters:

Or Man – ormn1996@gmail.com

Liad Yadin – liadyadin8@gmail.com

Table Of Content

1. Abstract

In computed tomography (CT) imaging, image quality depends on the patient's exposure during the scan. Reducing the exposure reduces patients' health risks, but also reduces image quality due to higher noise in the image. A denoising technique preserving image features enables the acquisition of low-dose images without losing too much information. In this work, the performance of convolutional neural networks in denoising CT images was evaluated.

The peculiarity of the network is that it does not require the usage of clean data for training. instead, two or more independent noise realizations of each image are input during training(Noise2Noise). The network output was compared with that of a network trained using clean images, We will go through the results of works that indicate that the absence of clean images during training does not prevent the network from learning a good denoising model.

In this work, an analysis of the Noise2Noise learning strategy is done using real noise and synthetic datasets. This paper demonstrates, using diverse network architectures and loss functions, that the duplicity of information in the noisy pairs can be exploited to reach increased denoising performance of Noise2Noise.

2. Introduction

In the medical field, doctors need accurate CT images to give better diagnostics and to treat patients in the best way.

If a CT image has noise it can cause some problems:

- The edges of organs can be blurry(not accurate), which can cause problems in determining the size of things in the image.
- Noise can change the "value" of some parts in the image that may cause misdiagnosis of the part material(fat, muscle, water, gray and white matter, lung …).
- Small features in the CT can be mistreated as noise if the noise levels are too high.

In current days not all countries have expensive medical equipment capable of creating high resolution & noise-free medical images. furthermore, to create such images the patient needs to be exposed to high-power radiation which can cause medical risks.

This project's goal is to create a network capable of denoising lower quality images without losing medical needed data, so even poor countries can benefit from good medical care and without exposing the patients to unnecessary risks.

Today there are image denoising methods that are being used in the medical field, for example, filter-based methods that try to smooth and sharpen the image to get rid of the noisy values. Such image filtering can harm the original image, these methods can be improved using deep learning.

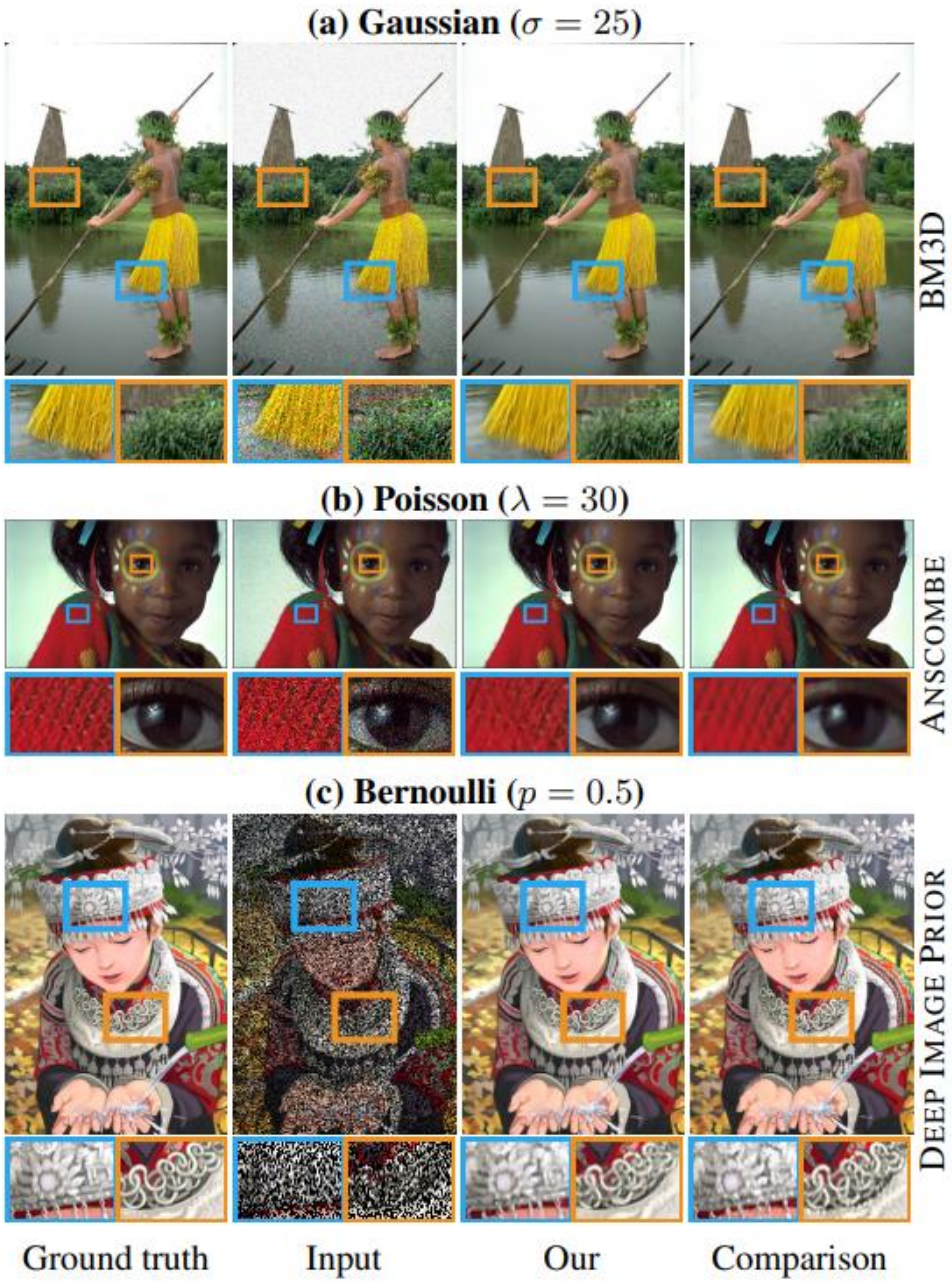


Figure 1 - Example of different noises and the importance of denoising.

Using traditional deep-learning methods can be tricky because they need the existence of a large "clean" dataset that is hard to get, most deep-learning methods need to see the desired output to be able to "learn" how to recreate it. And a "clean" image is not always in reach.

We propose the use of an encoder-decoder network with only "noisy" images ("Noise2Noise"), and the use of data augmentation to increase the size of the existing datasets.

In further sections, we will go through the background to the problem and solution.

We will describe the method in which we tackled the problem and our proposed method.

3. Background and Related Work

3.1. Background

CT Noise

CT has a high contrast sensitivity characteristic which is used to differentiate among the soft tissues within the human body. This characteristic is affected by noise which harms the visualization of low contrast structure. Before image denoising, the types of source noise and general properties of noise in CT images must be known [1].

Type

Before considering methods for noise reduction in CT images, it is important to get an overview of source noise.

Random noise:

Some of the X-ray rays may act differently in a random matter which can create differences in the density of the data in specific locations.

Statistical noise:

The energy levels of the X-ray rays are not constant and variance in statistical form. The only way to reduce the effects of statistical noise is to increase the number of detected X-rays. Normally, this is achieved by increasing the number of transmitted X-rays through an increase in X-ray dose.

Electronic noise:

Noise from the electric circuits used to receive the analog signals of the CT. The latest CT scanners are well designed to reduce electronic noise.

Roundoff errors:

Noise that created by the limited ability to transform the analog signal to digital. This transform can be achieved only using rounding of the data which can cause rounding errors [1].

Examples of noisy CT images in Figure 2.

Figure 2 - A noisy image is the sum of the clean image and the noise component.

Distribution

Generally, noise in CT images is introduced mainly by two reasons. First, a continuously varying error due to electrical noise or roundoff errors can be modeled as simple additive noise, and the second reason is the possible error due to random variations in detected X-ray intensity.

The distribution of noise in CT-image can be accurately characterized using the Poisson distribution. But for a multi-detector CT (MDCT) scanner, the noise distribution is more accurately characterized by the Gaussian distribution.

The literature also confirms that the noise in CT images is generally an additive white Gaussian noise. [1]

Denoising

CT image denoising can be better performed with prior knowledge about CT images and the noise. Without prior knowledge, the accuracy of CT image denoising is not effectively improved.

Conventional smoothening and sharpening filters are the most popular filters for noise reduction in digital images. Smoothening filters are not sufficient for effective noise reduction, especially for higher noise, and they may also harm detailed parts such as edges due to the smoothness factor. For example Figure 3.

This is the major problem of image denoising in medical image analysis because missing structures give an inaccurate analysis of CT images which may harm the life of a human.

The main challenges for noise reduction in CT images are:

- Flat regions should be flat.
- Image boundaries should be preserved (no blurring).
- Texture details should not be lost.
- Global contrast should be preserved.
- New artifacts should not be generated

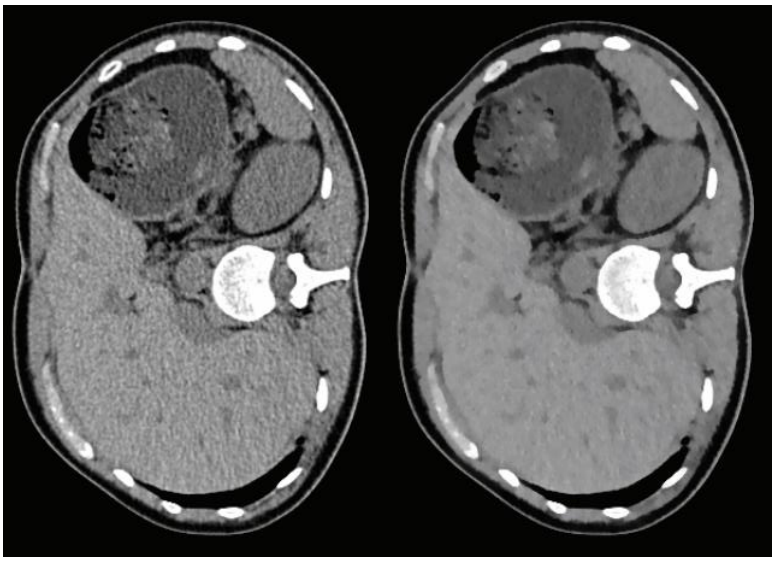


Figure 3- Noisy CT and blurry denoised output

Peak Signal-to-noise Ratio (PSNR)

Peak Signal-to-noise Ratio (PSNR) is an important factor to evaluate denoising performance. The high PSNR value represents more similarity between the denoising and original image than lower PSNR values. For clean image (X) and denoised image (R), the PSNR is expressed as:

$$PSNR = 10 \cdot \log_{10}\left(\frac{255 \cdot 255}{MSE}\right) \tag{1}$$

$$MSE = \frac{1}{m \cdot n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[X(i,j) - R(i,j)]^2 \tag{2}$$

Encoder-Decoder CNN

One typical application of the CNN is for classification tasks, where the output of each image is a one-hot vector indicating the likelihood that the image belongs to each class. However, in some tasks, such as cell segmentation in biomedicine, we need to classify not only an image but also each pixel in the image. This kind of task is called semantic segmentation (or image segmentation) [2] [3].

The main Idea of the Encoder-Decoder CNN was to replace the fully-connected part of the CNN(used for classification) with convolutional layers, which are referred to as the Decoder part of the network, while the existing convolutional layers are called Encoder.

The purpose of the Encoder part is to reduce the image size while increasing the information kept for each pixel. While the purpose of the Decoder part is to increase the image size back to its original size while keeping only the necessary information.

This network architecture can be used both for semantic segmentation (where the necessary data is the class) or for image denoising (where the necessary data is the clean image).

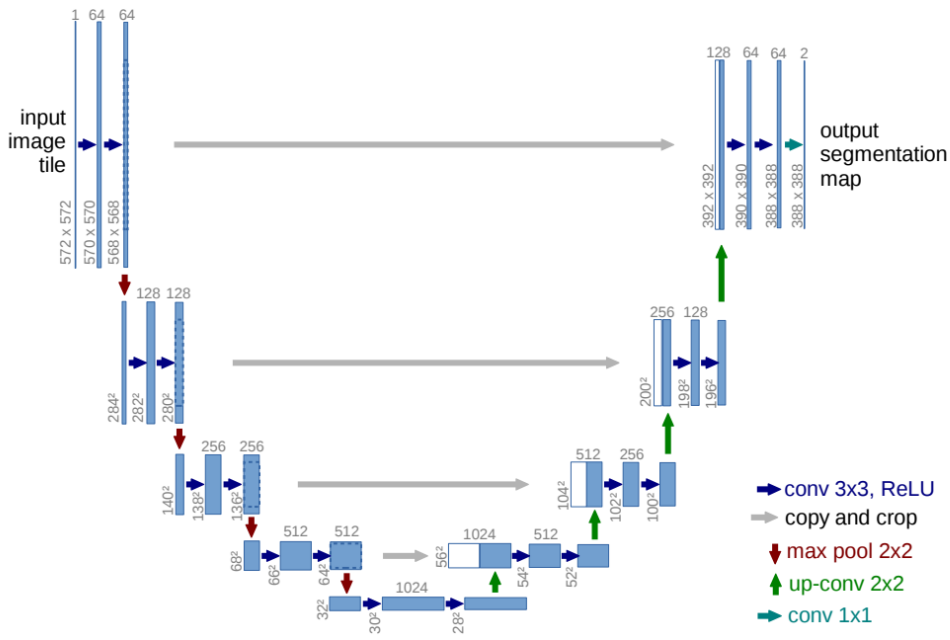


Figure 4- Encoder-Decoder CNN example, where the left side is called Encoder, and the right is called Decoder

The Idea in the Decoder part is to create an opposite network, where pooling operators are replaced by upsampling operators. Hence, these layers increase the resolution of the output. In order to localize, high-resolution features from the Encoder path are combined with the upsampled output. A successive convolution layer can then learn to assemble a more precise output based on this information [2].

Examples of such network can be seen in Figure 4 and Figure5 .

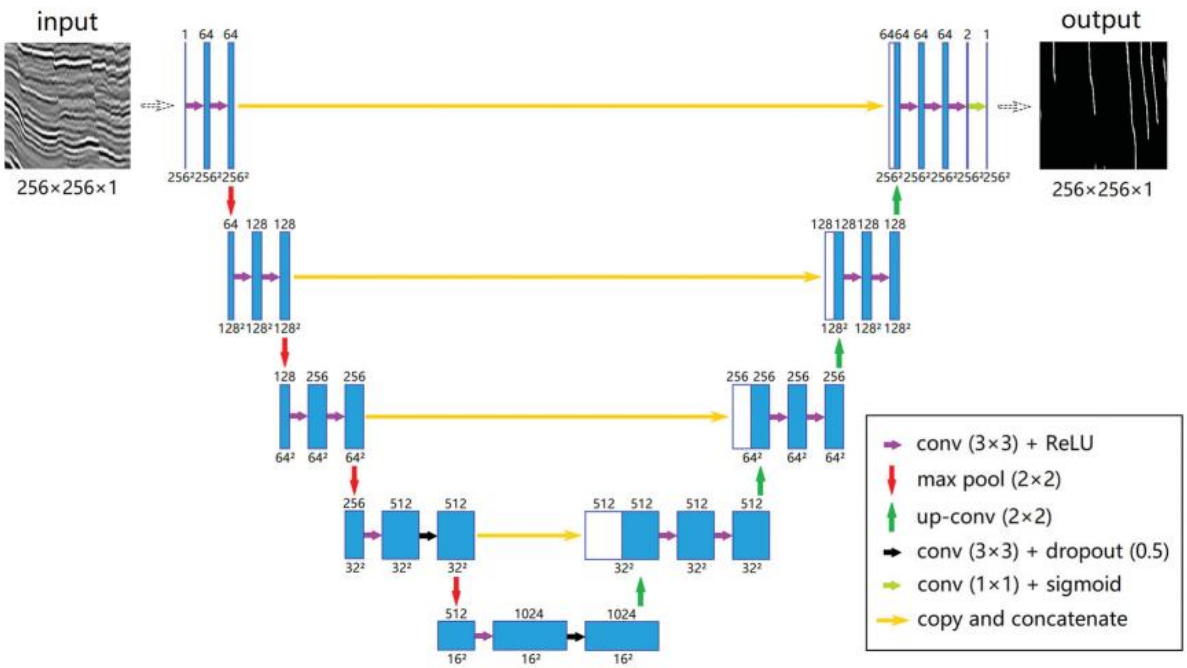


Figure5 - Example of Encoder-Decoder based network used for seismic fault detection [3]

Network components:

Convolution layers - The convolution layer is the first action to extract features from the input image. Convolution maintains the connection between pixels by studying image features using tiny squares of input data. The convolution layer is the core structure block of a CNN that does most of the complex calculations. The Convolution layer's parameters consist of a set of learnable filters.

During the forward pass, we convolve each filter over the width and height of its input volume and try finding edges of some orientation, spots of some color, or patterns on higher layers of the network. Then we compute dot products between the values of the filter and the input at any position. The result will be a 2-dimensional activation map that gives the responses of that filter at every spatial position.

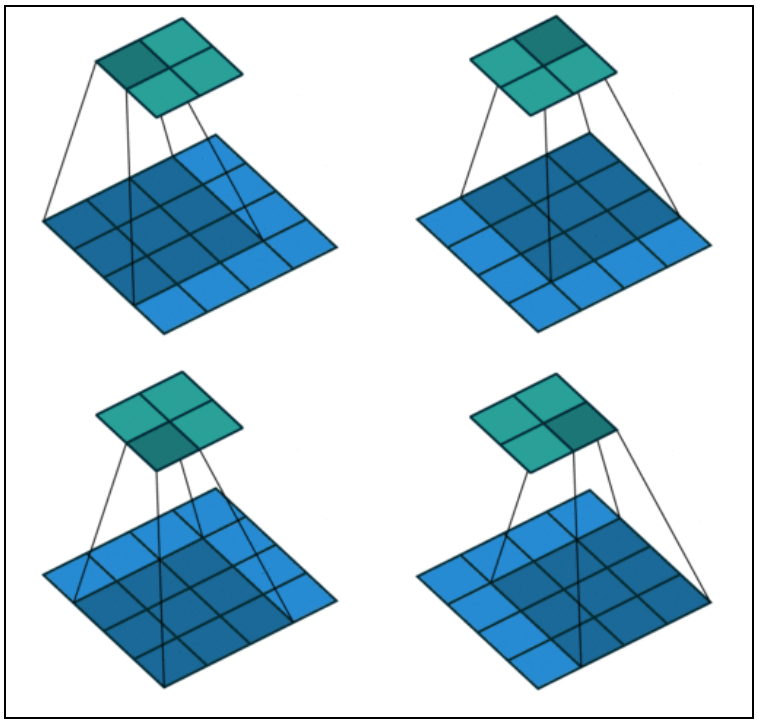


Figure 6 - 3x3 Convolution

Max pooling - The max pool layers are in charge of downsampling the spatial dimensions of the input while retaining essential information. The most common setting is to use max-pooling that calculates the maximum value for each patch of the feature map with 2x2 receptive fields and with a stride of 2.

It is very uncommon to see receptive field sizes for max-pooling that are larger than three because the pooling is then too lossy and aggressive and usually leads to worse performance [4].

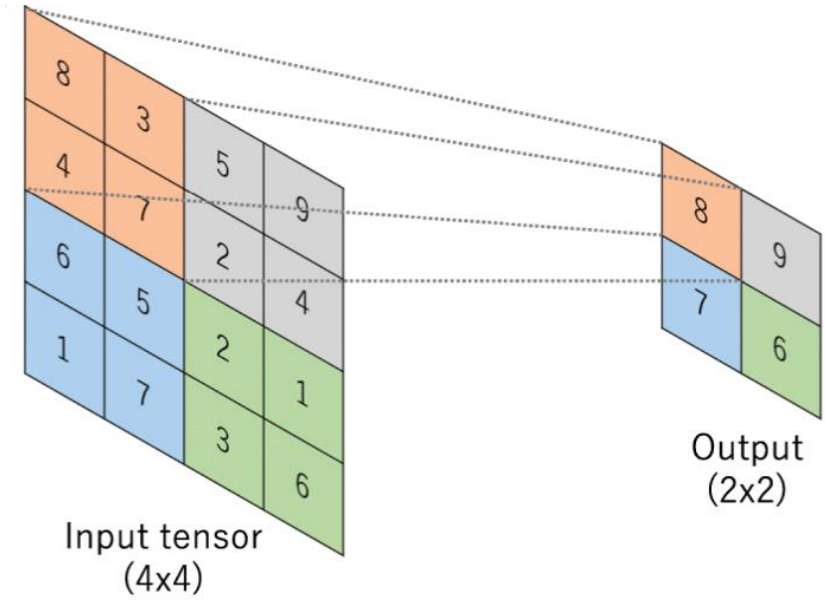


Figure 7 - Example of MAX-pooling with a stride of 2.

<u>Up-convolution layers</u> - The up-convolution layers are pool layers that are in charge of upsampling the spatial dimensions of the input. Usually used in the same dimension of the corresponding max-polling used before.

The up-convolution works similarly to the convolutional layer but in a backward way. Each value in the input will be multiplied by the filter and will be put in the output in the correct position(going with a stride of 2 for example for 2x2 up-convolution)

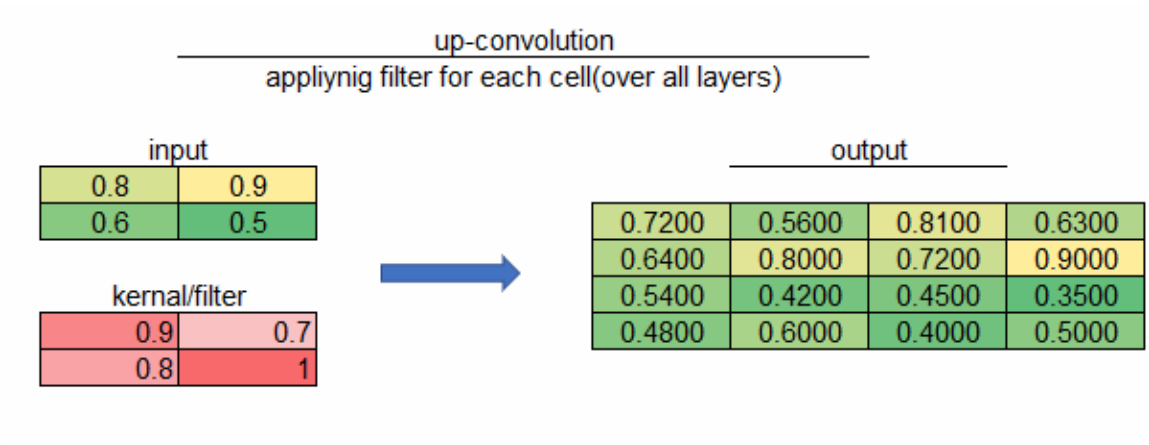


Figure 8 - 2x2 up-convolution example.

Copy and Concatenate - These types of connections between layers are called skip connections. Broadly speaking, there are two advantages to using skip connections in this case:

- They allow gradients to more freely flow through the model, helping the issue of vanishing gradients
- They allow features from the encoder side of the network to be transferred to the decoder side of the network, adding extra information that might be lost because of the downsampling on the encoder side of the network.

3.2. Related Work

P. Gnudi et al. said that a denoising model can be obtained by training a CNN model with a large number of training pairs $(\hat{x}, x)$ of noisy inputs $\hat{x}$ and clean images x. In such cases, the training inputs are the noisy images and the training targets are the clean images. The network is trained to output an image matching the clean input image as well as possible, as measured by a loss function, learning to denoise a noisy image by looking at both noisy and clean images, this method is called Noise2Clean (N2C). One of the problems in this approach is to find a sufficient number of clean targets to train the model.

To create a clean target one can try using longer acquisition times, which usually increases the signal-to-noise ratio, but is not always practical in medical imaging due to issues such as patient comfort and the higher operating costs. In some medical imaging techniques, such as X-ray and CT imaging, a longer acquisition time also exposes the patient to a higher quantity of radiation, which should be avoided.

A denoising model can also be obtained by training a CNN model with a large number of training pairs $(\hat{x}, \hat{y})$ of noisy inputs $\hat{x}$ and another noisy realization of the training instances as targets $\hat{y}$. In such a situation, the network is forced to learn to reproduce the second noisy realization, which is an impossible task because of the random behavior of the noise. Still, through a large number of training samples, the network learns to reproduce an average representation instead, which yields the desired denoising effect, this method is called Noise2Noise (N2N). In their work, the authors showed that, for noise removal in magnetic resonance imaging, the results produced with Noise2Noise are comparable with the ones produced with Noise2Clean.

The Noise2Noise technique is particularly suitable to train a denoising model for medical applications because sometimes it can be challenging to obtain enough clean training targets. The Noise2Noise approach solves this problem not least because with three noisy realizations of the same image, six training pairs can be produced. In general, if N is the number of noisy realizations per image and I is the total number of available images, the number of possible training pairs is $I \cdot (N \cdot (N-1))$.

Table 1- Comparison between N2C to N2N on CT-image dataset

EXPERIMENTAL RESULTS ON THE CT DATASET (BASELINE PSNR 24.44 dB)

Tool	Mean PSNR [dB]
Noise2Noise MAE loss	41.80
Noise2Noise MSE loss	42.09
Noise2Clean MSE loss	40.63

Comparison between N2C to N2N on CT-image dataset[Table 1], the N2N got a better score than the N2C using both loss functions [5].

Example of the N2N denoising in Figure 9.

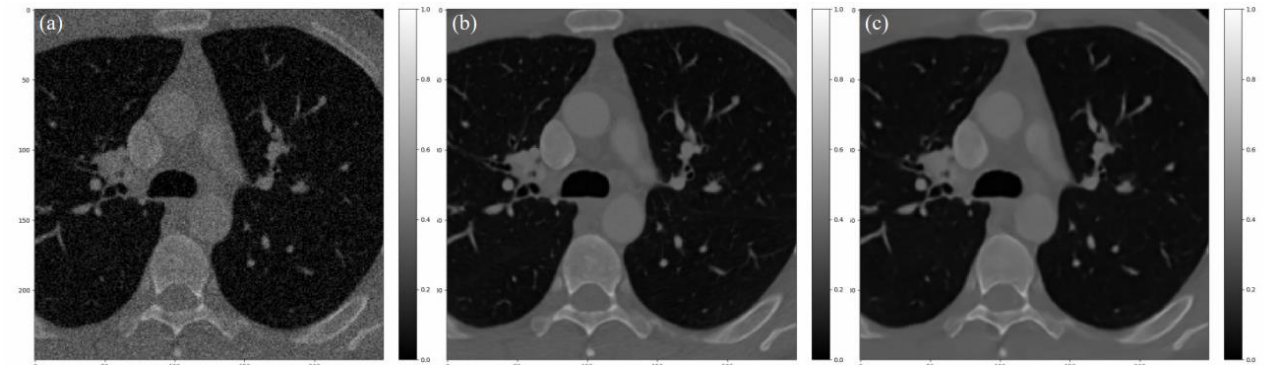


Figure 9-Comparison between (a) noisy input (24.44 dB), (b) clean reference, and (c) prediction of the Noise2Noise model trained with the MSE loss (40.22 dB).

N2N Vs N2C output

Figure 10b and Figure 10c show outputs of N2N and N2C, in which the PSNR of the N2N is almost identical to the PSNR of N2C [6].

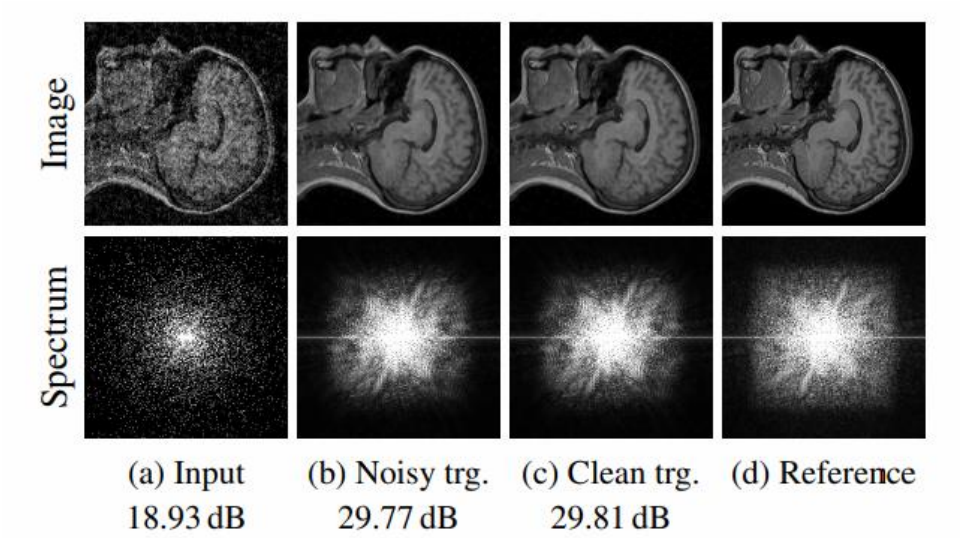


Figure 10 - MRI reconstruction example. (a) Input image with only 10% of spectrum samples retained and scaled by 1/p. (b) Reconstruction by a network trained with noisy target images similar to the input image(N2N). (c) Same as previous, but training is done with clean target images similar to the reference image(N2C). (d) Original, uncorrupted image.

C. A. Font state the performance of Noise2Noise drops when the amount of training data is reduced, limiting its capability in practical scenarios. Commonly used datasets for training denoising networks are designed for synthetic noise, which can be added on-the-fly, and are usually made up of several hundred to several thousand images. However, the smaller training datasets get, the less one can theoretically expect Noise2Noise to remain competitive.

The assumption of unlimited noisy samples per scene is not realistic in many practical scenarios (e.g. if a noise model is not known). Let us assume only two noisy samples ($\hat{y}$,$\hat{x}$) per scene are available.

Assuming noisy samples are drawn from the same distribution, one obvious idea is that they can both be used as input and target respectively(e.g. the pair $[\hat{x}, \hat{y}]$ and $[\hat{y}, \hat{x}]$ can be used in the network).

This method of using both pairs to increase training data size is called Alternating Noise to Noise (AltN2N).

Just like AltN2N to increase the size of the training set we can see that changes in only a few pixels in the input or the target can yield virtually unseen samples of a scene. In a Noise2Noise setting, the pixels (or pixel regions) in $\hat{y}$ and $\hat{x}$ are interchangeable as long as the images are well-aligned, and the noise is not correlated (or correlation is not destroyed in the process). Under this assumption, one can swap one or more single pixels (or pixel regions) between $\hat{y}$ and $\hat{x}$, such that two new unseen yet plausible images $\hat{y}_s$ and $\hat{x}_s$ are generated[Figure 11]. This method of using both pairs to increase training data size is called surrogate Noise to Noise(SN2N) [7].

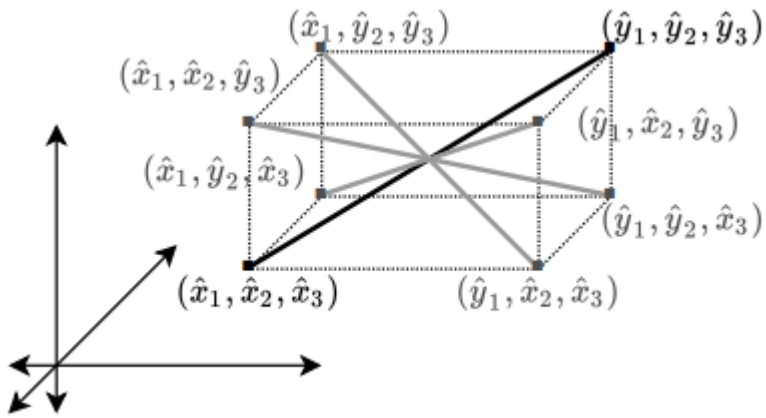


Figure 11- Illustration of the noise surrogates strategy, for a Noise2Noise image pair $(\hat{x}, \hat{y})$ of 3 pixels(or pixel regions) each.

<u>Comparison of AltN2N, SN2N, N2N, and N2C:</u>

As can be seen in Figure 12, while the Noise2Clean strategy quickly reaches higher overall metric values, using the noise surrogate technique eventually causes the learning trend to dissociate from the trends of the other Noise2Noise techniques and eventually reach levels comparable to those of the Noise2Clean approach [7].

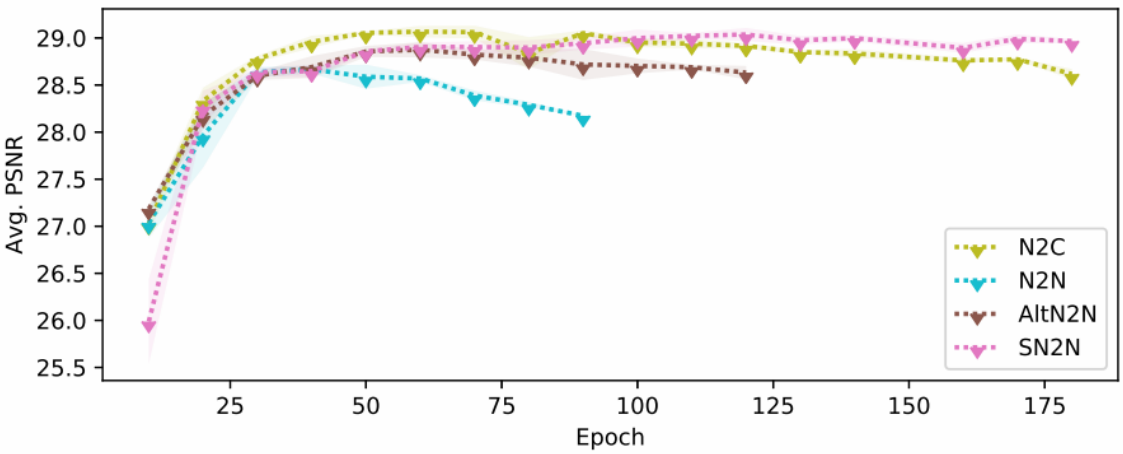


Figure 12- Average trends of PSNR(dB) throughout the training of the networks.

A. M. Hasan et al. state that competition and collaboration are widely used in the multiagent environment. Competition is analogous to the adversary principle between the generator and discriminator in various generative adversarial network (GAN) models and it has found a few applications in CT image denoising tasks. GAN models use the min-max optimization framework, where the generator tries to reconstruct fake images from random noise; the discriminator works as a classifier to distinguish between the real and fake images. As the generator gets sufficiently trained, it starts to produce realistic images as fake images, and then the discriminator finds it difficult to distinguish between real and fake images.

Architectures[Figure 13 show the concept of the different networks]:

- CN with two generators (CN2G) that work on two low-dose image sets.
- CN with three generators (CN3G) that work on three low-dose image sets.
- hybrid CN three generators (HCN3G) using one of our previous works with blind source separation (BSS) with a block matching 3-D (BM3D) filter.

Table 2 shows the comparison between these architectures [8].

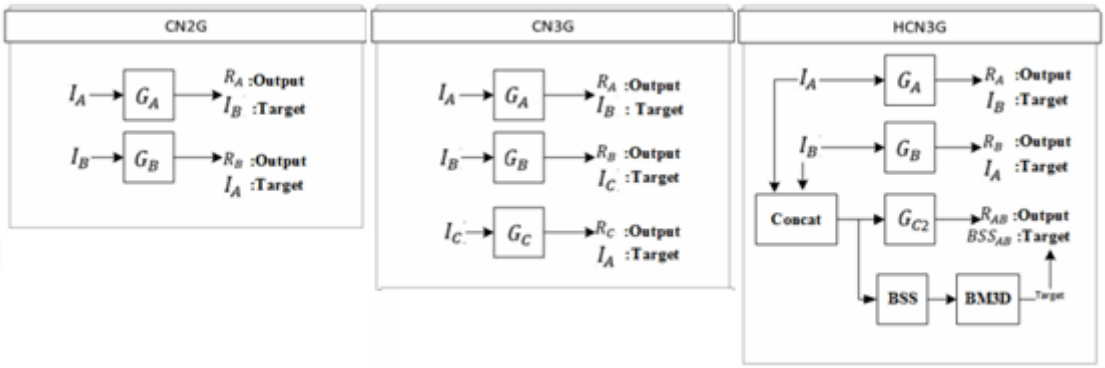


Figure 13 - GAN architectures examples

PSNR IN dB FOR VARIOUS METHODS AND MODULES

Module	Input LDCT	Benchmark		Our methods		
		N2N	BSS+ BM3D	CN2G	CN3G	HCN3G
CTP404	24.53 ± 0.07	31.12 ±0.08	31.29 ± 0.29	33.38 ± 0.11	34.06 ± 0.26	**34.61** ± 0.12
CTP528	22.38 ±0.12	30.12 ±0.16	29.66 ±0.15	31.46 ± 0.45	33.30 ± 0.45	**33.89** ± 0.02
CTP591	24.43 ±0.02	31.55 ±0.05	31.48 ±0.30	33.13 ± 0.33	33.94 ± 0.34	**34.62** ± 0.07
CTP515	23.54 ±0.40	31.78 ±0.23	30.19 ±0.45	33.54 ± 0.20	**34.51** ± 0.39	33.76 ± 0.35
CTP486	25.12 ±0.17	32.08 ±0.04	32.37 ±0.27	34.25 ± 0.29	35.11 ± 0.33	**35.78** ± 0.13

Table 2 -GAN compare for CN2G, CN3G, and HCN3G

J. Lehtinen suggested Model RED30. This model is a 30-layer hierarchical residual network with 128 feature maps, which has been demonstrated to be very effective in a wide range of image restoration tasks, including Gaussian noise. In this model, we train the network using 256*256-pixel crops drawn from the 50K images in the IMAGENET validation set. We furthermore randomize the noise standard deviation $\sigma \in [0,50]$ separately for each training example, I.e. the network has to estimate the magnitude of noise while removing it [6].

4. Expected Achievements

We expect to build a system that will be able to denoise a CT image with better accuracy than other available methods.

This project's goal is to create a network capable of denoising lower-quality images without losing needed medical data.

We will train the network using real medical data(Noisy images). after the training, the network will be able to take any CT image and improve it(by denoising).

Our network model is going to be based on the Encoder-Decoder CNN architecture. We are going to compare the model result according to the hyperparameters that will be mentioned later.

The system will help us to improve CT images so even poor countries(without expensive medical equipment) can benefit from good medical care and without exposing the patients to unnecessary risks.

For us, success will be expressed in:

- Build a network based on the Encoder-Decoder CNN with our modifications.
- Create a network that can be trained using the available medical data.
- A system that can reduce noise by a factor of 10-15%.
- Find the best values for the hyperparameters.

One of the challenges in denoising is that the use of "clean" images to teach our network is difficult because "clean" images are hard to get by. On the opposite, "noisy" images are more commonly found, so we decided to use the Noise-to-Noise algorithm which uses pair of "noisy" images to teach the network, with the same results and sometimes even better than Noise-to-Clean.

5. Research Process

5.1. Process

<u>Challenges and achievements</u>

- We have started working on the project by independent learning, Before starting the project our knowledge in machine learning accumulated in one course of "data mining and machine learning" which only gave us little background in the subject of machine learning.

  One of us took an additional course of "seminar in machine learning" to further deepen our knowledge in the subject. In the end, most of the knowledge was received through reading and researching.

- We had no medical background knowledge on that subject. We studied medical materials by ourselves, We read articles and visited a variety of websites dealing with that issue.

<u>Hyperparameters</u>

Learning rate – {0.000001,…,0.0001}.

Epochs – {50,100,180}.

Batch Size –{32,64}.

Loss function – {MAE, MSE, Huber Loss} inc. Huber Loss parameter δ [9].

Network depth(pool layers) – {4,5}.

<u>Dataset</u>

We use the DeepLesion dataset [10] which contains 32,735 clean CT images(Figure 14), so we can add synthetic noise, and use the clean version to assert the network quality. The data is diverse from 4,400 patients.

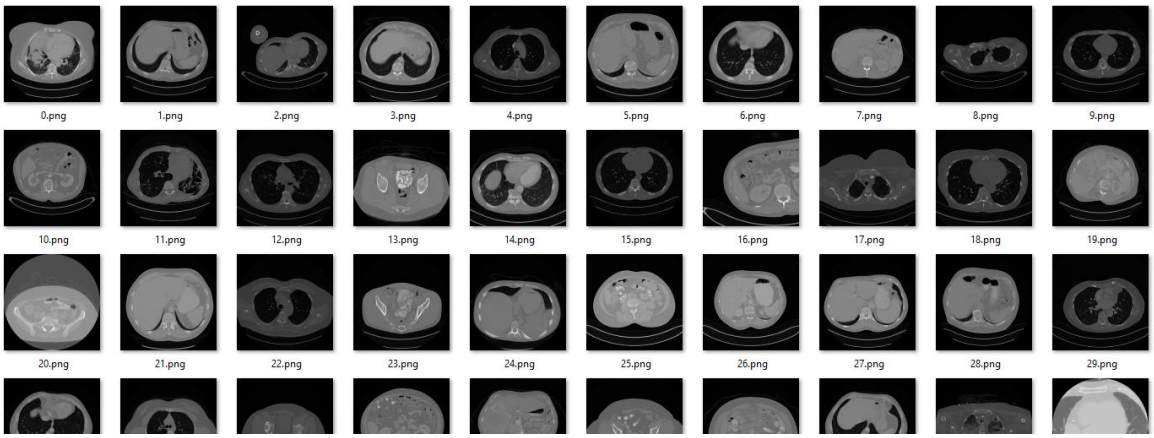
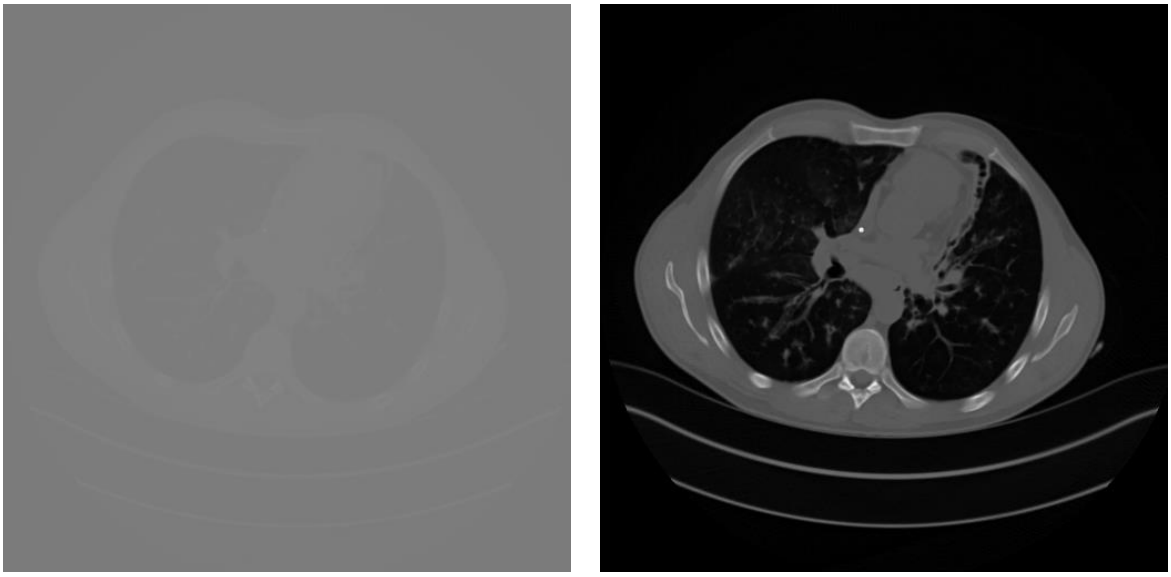


Figure 14 - Example of DeepLesion dataset after normalization

We will normalize the images(to a range of [0,1]) and then add an Additive White Gaussian Noise (AWGN) to each image.

Clean slices were rescaled by dividing all pixels in an image by the maximum pixel value of the image itself. The noisy images were rescaled by dividing each pixel by the maximum of the respective clean image. This produced noisy images where some pixel values were greater than 1. an example of normalization is in Figure 15.

The baseline PSNR of this dataset was 29.91 db.

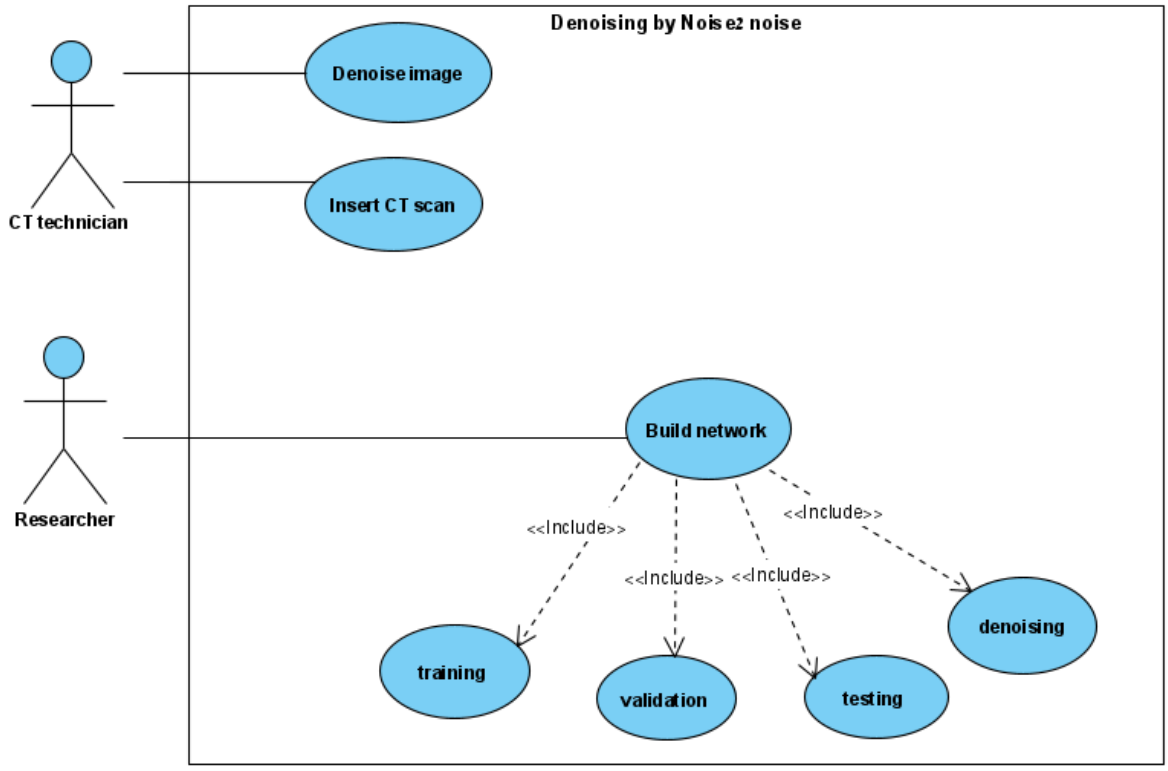


(a) **(b)**

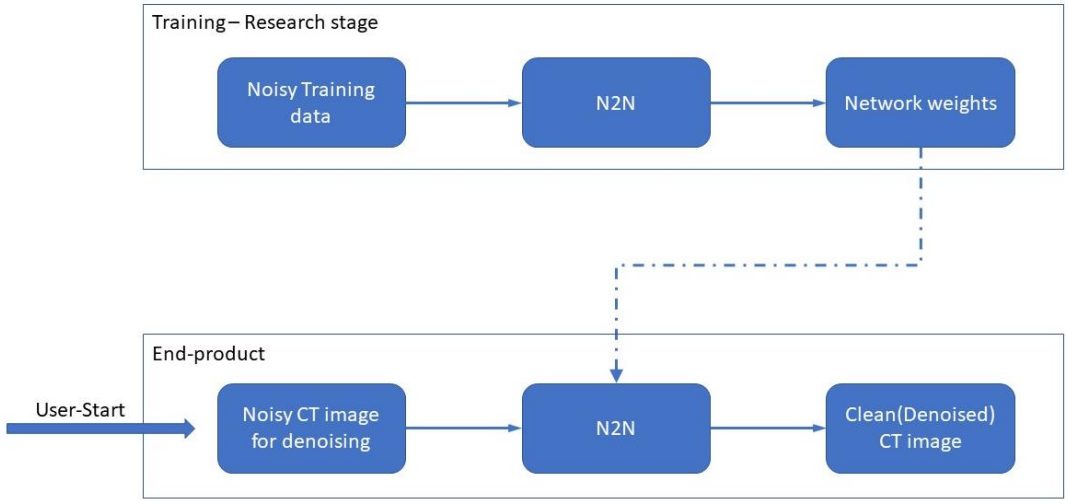
Figure 15 - Example of normalization, (a) image before normalization, (b) image after normalization.

5.2. Product

Use-Case



Flow



GUI

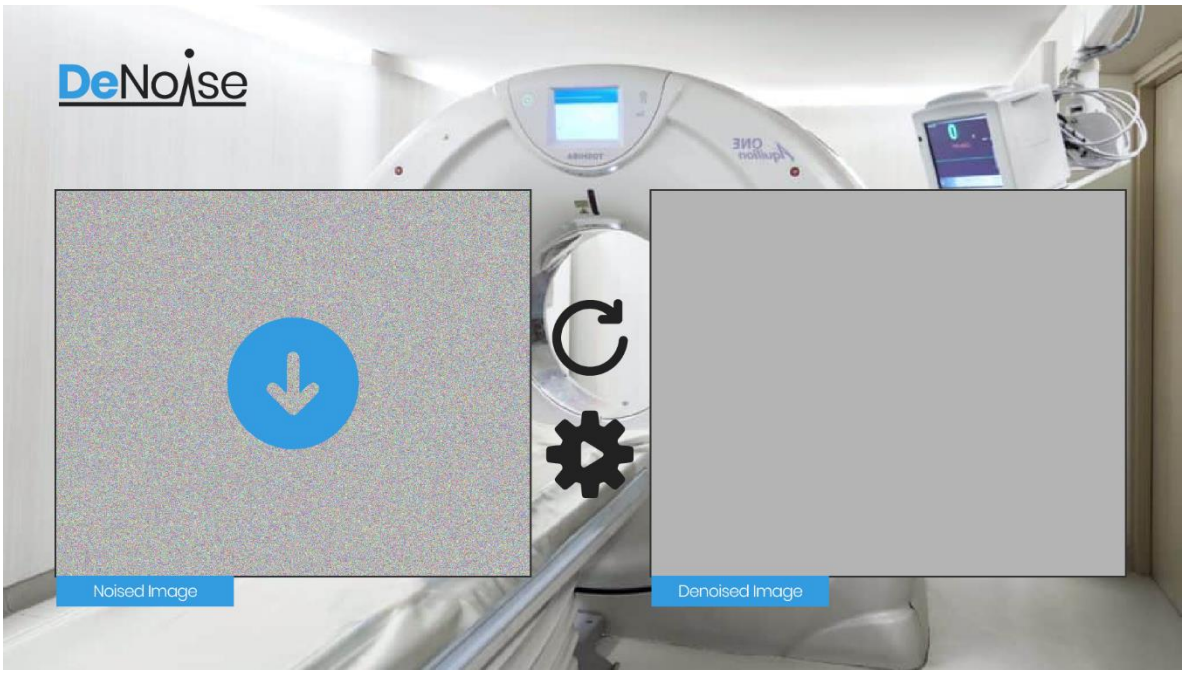


Figure 16 - Main page interface.

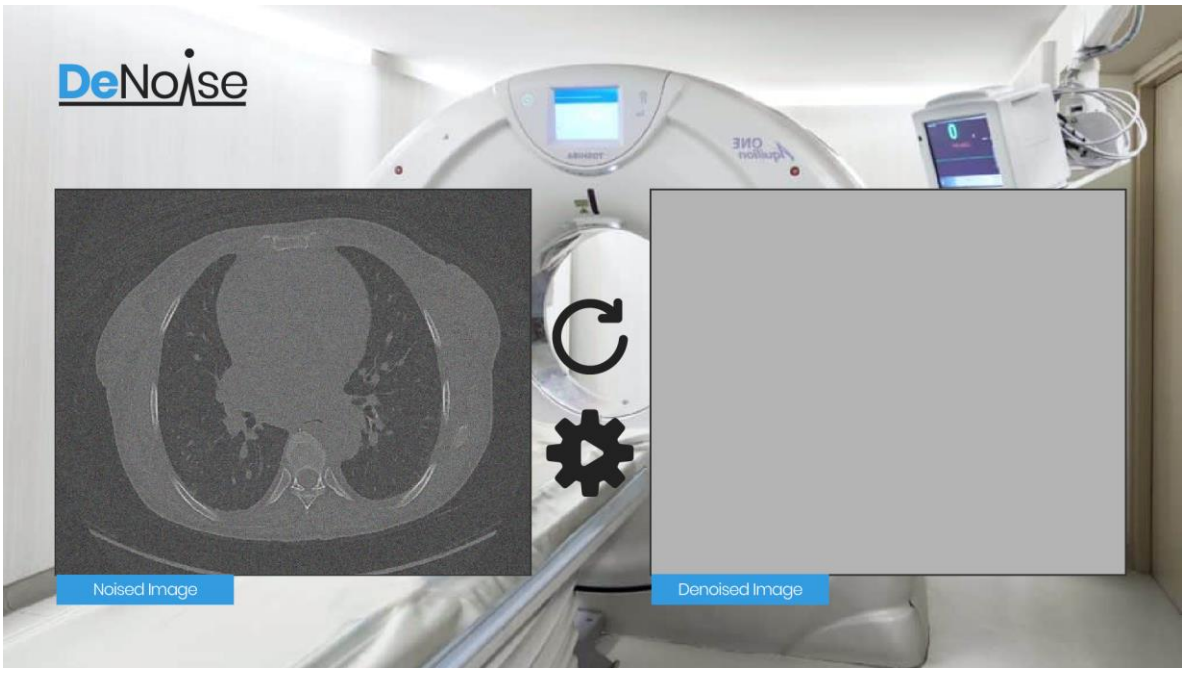


Figure 17 - Image after noisy image insertion.

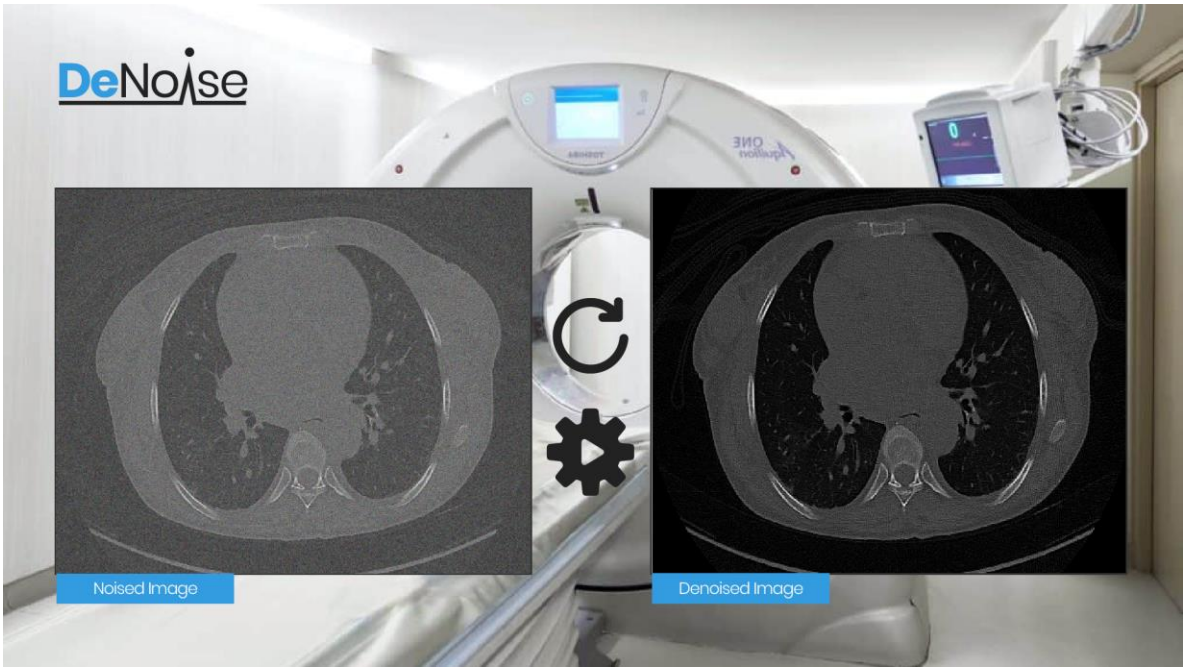


Figure 18 - Image after running the denoising.

6. Evaluation/Verification Plan

Case number	Test case	Expected result
1	Insert wrong picture format	Error message: "Wrong format"
2	Insert expected picture format	The system will denoise the picture
3	Press "Denoise CT scan" without inserting a picture	Error message: "Insert picture"
4	Press "Denoise CT scan" after inserting a picture	The system will denoise the picture
5	Press "back"	Back to the main window

7. References

[1] M. Diwakar and M. Kumar, "A review on CT image noise and its denoising," *Biomedical Signal Processing and Control,* vol. 42, pp. 73-88, 2018.

[2] O. Ronneberger, P. Fischer and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention,* Springer, 2015, pp. 234-241.

[3] S. Li, C. Yang, H. Sun and H. Zhang, "Seismic fault detection using an encoder-decoder convolutional neural network with a small training set," *Journal of Geophysics and Engineering,* vol. 16, no. 1, pp. 175-189, 2019.

[4] "CS231n Convolutional Neural Networks for Visual Recognition," [Online]. Available: https://cs231n.github.io/convolutional-networks/. [Accessed 21 12 2021].

[5] P. Gnudi, B. Schweizer, M. Kachelrieß and Y. Berker, "Denoising of X-ray projections and computed tomography images using convolutional neural networks without clean data," 2020.

[6] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala and T. Aila, "Noise2Noise: Learning Image Restoration without Clean Data," *arXiv preprint arXiv:1803.04189,* 2018.

[7] C. A. Font, "Improved Noise2Noise Denoising With Limited Data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2021, pp. 796-805.

[8] A. M. Hasan, M. R. Mohebbian, K. A. Wahid and P. Babyn, "Hybrid-Collaborative Noise2Noise Denoiser for Low-Dose CT Images," *IEEE Transactions on Radiation and Plasma Medical Sciences,* vol. 5, no. 5, pp. 235-244, 2020.

[9] G. Seif, "Understanding the 3 most common loss functions for Machine Learning Regression," towards data science, 21 May 2019. [Online]. Available: https://towardsdatascience.com/understanding-the-3-most-common-loss-functions-for-machine-learning-regression-23e0ef3e14d3.

[10] NIH Clinical Center (CC), 20 July 2018. [Online]. Available: https://www.nih.gov/news-events/news-releases/nih-clinical-center-releases-dataset-32000-ct-images.

[11] T. Zhao, M. McNitt-Gray and D. Ruan, "A convolutional neural network for ultra-low-dose CT denoising and emphysema screening," *Medical physics,* vol. 46, no. 9, pp. 3941-3950, 2019.

Appendix A – Table of Figures