Noise2noise Implementation report 1

Qing Jun, François Charroin, Khalfallah Selim *EPFL, Switzerland*

***Abstract*—Noise2noise model contains encoder and decoder processes and is belongs to the machine learning-learning strategy by compressing and reconstructing images to removing noise signals. In this project, different noises are implemented to see if we could generalize the model. Various training pipeline such as data augmentation, loss function, optimizer are also explored to find the best model. Although the psnr value would be different concerning to the noise type for training and validation, we arrive the average 25.43 psnr for the validation dataset provided by the teacher as well as the validation dataset we create combining various noise type. The total training time is less than three minutes with ten epochs and the batch size of 64 with GPU.**

I. Introduction

In this project, we use the noised input images and clean target to train the model and minimizing the empirical lost.

$$argmin_\theta \sum_i L\left(f_\theta\left(\hat{x}_i\right), y_i\right) \tag{1}$$

The input images have 3 channels and is size of 32×32. We add the Gaussian noise and Poisson noise to the clean targets to create new dataset. Besides, the random crop and the flip are also considered in the data augmentation part, which is directly implemented in the code. For the optimizer, We use the mini-batch training and choose two optimization method, the Adam and Momentum SGD with the weight decay strategy. The Adam shows a better results and quicker descending to the optimum. For the loss function, the L2 loss performs a higher psnr value than the L1 loss with in the training process (only 10 epochs).

II. Network Structure

The network is the same with the UNet that Jaakko Lehtinen[1] explored in the Noise2Noise thesis. The network contains convolution layer with kernel 3×3 and same padding strategy, maxpool layer with kenel size 2, up-sampling layer, and activation layer with ReLu and LeakyRelu, only the last layer applies ReLu.The schematic diagram is shown in the following figure. The L1 and L2 loss are evaluated with the recover image and clean target.

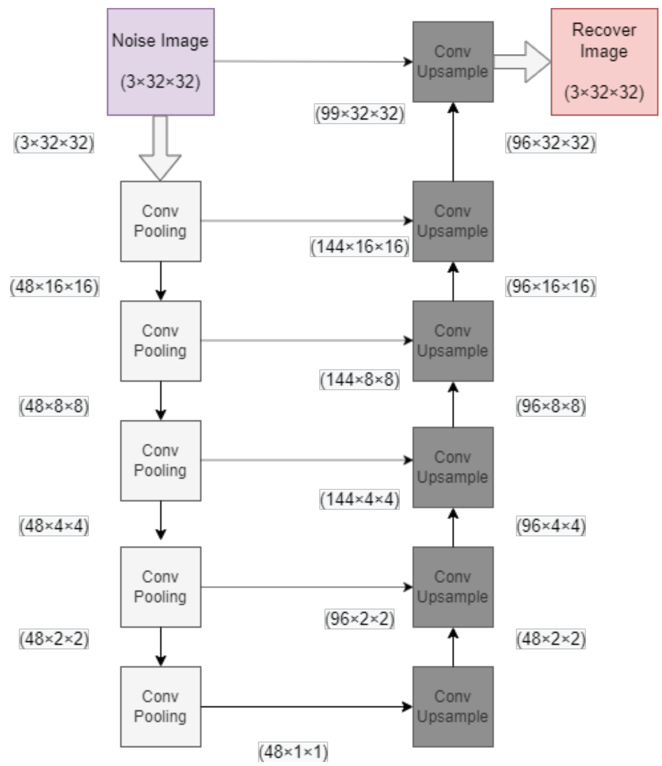

Figure 1. schematic diagram

III. Data Preprocessing

A. Dataset

The dataset is divided into train and validation files with corrupted images and corresponding clean images in each file. The file for training contains 50,000 pictures and the validation file contains 1,000 images. The images are the size of 3×32×32.

B. Data Augmentation

We perform the random crop and vertical and horizontal flip to the training dataset. Keeping the same operation to the pairs of noised and clean picture is an important part in the data augmentation.
Since we need to limit the training time, we only perform the data augmentation randomly on half of the overall training data. The data augmentation strategy is ignored when loading the validation as well as test dataset.

C. Gaussian Noise

Since the noise in the training set remains unknown, we use the clean targets in the training set to create noises with various type to generalize the data, and perform the analysis within these new dataset. Gaussian noise refers to a class of noise whose probability density function follows a Gaussian distribution (normal distribution).We choose five different Gaussian noise with sigma ranging from 0.1 to 25 in 50,000 clean images, 80 percent of which is selected as training data and the remained is left for validation.

D. Poisson Noise

Poisson noise is a model that conforms to the Poisson distribution, which is suitable for describing the probability distribution of the number of random events occurring per unit time. Poisson noise is also applied to the dataset.

IV. Hyperparameters Search

Before we search the results for general noise, The well-performed hyperparameters are chosen based on the original dataset performed by the teacher.

A. Regularization and Optimizer

We choose the mean square loss (L2) and L1 loss for the loss function. Adam and Momentum SGD are choosed for the optimizer. The parameters for Adam are beta1 = 0.9, beta2 = 0.99, e = 10^{-8}. The SGD Momentum parameter is 0.98 and the weight decay rate is 0.01. Finally, we find that the L2 loss with Adam is the best combination, and we also notice that SGD performs better with L1 loss.The validation psnr with the training epochs is shown in the following figure.

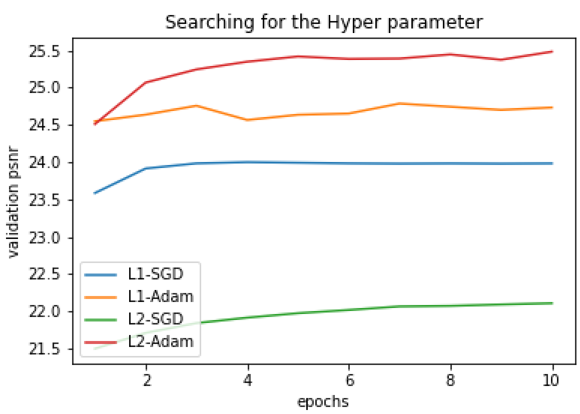

Figure 2. loss-optimize search diagram

B. Batch Size

After choosing the loss function and optimizer, we are going to search the best batch size for the training. We explore the batch size of 16, 32, and 64 with the L2 loss and Adam on the original training and validation set, the result is performed in figure 3. With 10 epochs of training, all of them reach a nice result. Concerning the converge speed as well as the accuracy, the line with batch size of 32 performs best.

C. Results Present

We get the validation psnr of 25.43 with the UNet, L2 loss, Adam, batch size of 32 and 10 epochs. The figure 4 presents the results of our model. We could see that the noise is removed from the noise input, and the predict image become smooth.

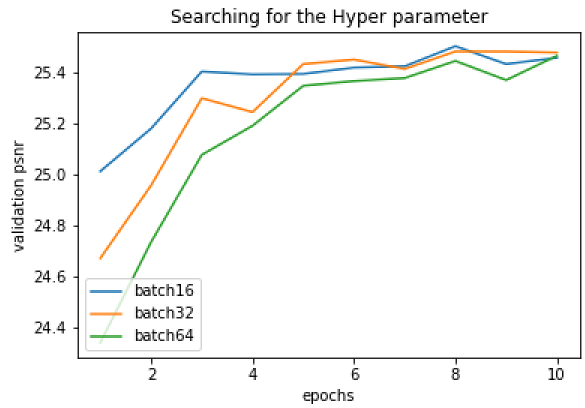

Figure 3. batch search diagram

Noised Image　Predict Image　Target Image

PSNR = 25.43

Figure 4. prediction diagram

V. Noise Generalization Results

For this part, we want to find whether using more noise data for training can generalize our model.

A. Generalize Dataset

For each noise type, we create a train and validate data and we choose 50 percent of noise data we create and 50 percent noise data provided by the teacher to establish a combined dataset, and explor whether the latter one are able to perform better on other noise dataset.

B. Validation

1) Train Specific Noise with according images: Before use the original or combined dataset to test the results on various noise data, the validation psnr is evaluated with corresponding noise data. This procedure is to find the maximum validation psnr value for different noise images. The result is shown in figure5. From the picture, we can find with the increasing of the sigma value for Gaussian noise, the picture is less able to reconstruct to clean target, which is a obvious view. With larger sigma value, the picture is damaged greatly and it becomes hard for the model to recover them.

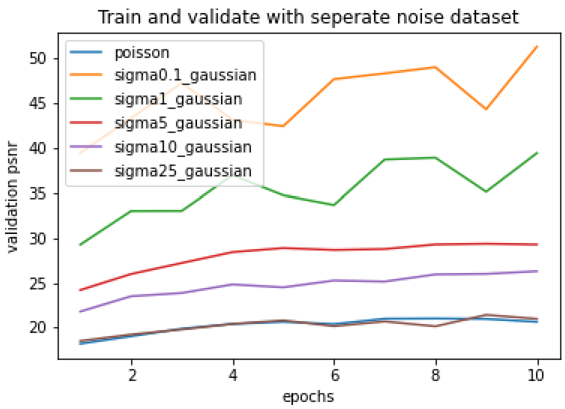

Figure 5. Noise search diagram

2) Test Specific Noise with original and combined image separately: Here we use the dataset provided by the teacher to train the model and look into the validation results on various noise data. We could find that this model do not have a good generalization, the psnr of these noise is less than those trained on their own dataset.

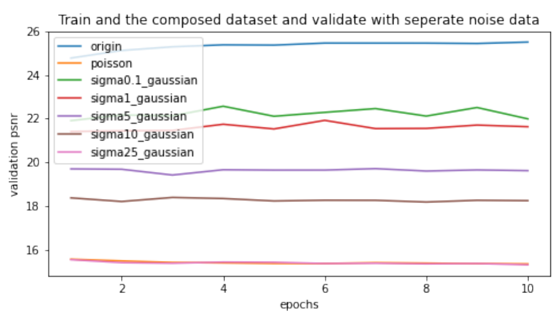

Figure 6. None Generalize results search diagram

For comparison, we use the dataset combined with all these noise for training and validate the psnr on various noise data. The result can be found on figure 7. According to the figure 6 and figure 7, we can make two conclusion. Using combined dataset do not affect the performance on origination validation set, while it also do not imporve the performance on other noise data.

There are two explanation on it. One reason is that even we add training sample from different noise data, the total amount is not enough, because all of the 6 other noise only contribute to the 50 percent of overall training data. The other reason is that this model can only reconstruct the noise image that are specific trained, training too much other noise data would affect the effect on original noise data.

[Generalize search diagram]

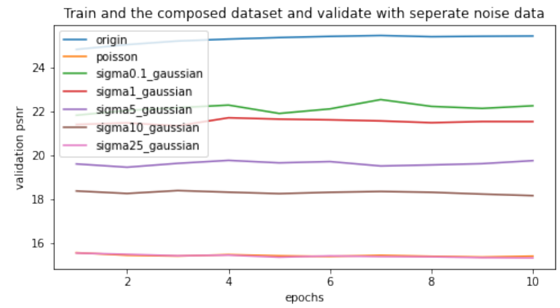

Figure 7. Generalize search diagram

VI. Summary

The original data is processed with data augmentation and many different noisy in our project. After the experiments on the hyper-parameters and a comparison of these methods, we choose the mean square lose with Adam optimizer with batch size 32 and reached 25.43 psnr in the validation set. Then, we explore the results on other noise data with adding them in the training set. The generalization results are not satisfied although the psnr in validation data provided by the teacher still remains high.

For further improvement, the model could be trained with more noise dataset to see whether it can improve the generazation results and find if it's conflict to train different noise on the UNet model. Also, other method for compression and reconstruction could be considered in this project like GAN for the generalization.

Acknowledgement

We would like to acknowledge joeylitalien for his noise2noise-pytorch in Github. The frame work has inspired us a lot and his utils.py in src folder help and motivate us to realize the function of monitoring the loss and validation psnr at every eopch.

References

[1] 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.