Exploring Noise2Noise for Image Denoising

Mihai David, Ioan-Florin-Cătălin Nițu, Ana-Arina Răileanu
{mihai.david, ioan.nitu, ana-arina.raileanu}@epfl.ch
School of Computer and Communication Sciences, EPFL, Switzerland

***Abstract*—Image denoising has applications in many of today's fields, from photography, where they can be used to remove noise from pictures, to medicine, where researches aim to clean X-rays and CT scan. This report discusses our approach to develop a Noise2Noise model trained on noisy pairs of images, with the purpose of image restoration. Our best model achieves a Peak Signal-to-Noise Ratio of** 25.66 **in** 19 **epochs.**

I. Introduction

A Noise2Noise model is an image denoising network trained without a clean reference image [1]. Our purpose is to reduce the effects of downsampling on unseen images using a robust neural network trained on the given data II.

II. Data

The data consists of 50,000 noisy image pairs for training and 1000 noisy image pairs for validation. Each pair in the training and validation datasets correspond to downsampled, pixelated RGB images with a size of 32x32 pixels.

A. Preprocessing

We apply a simple normalisation on the training image pairs for changing their range from [0, 255] to [0, 1]. Small pixel values result in faster convergence rates and easier computations.

B. Data Augmentation

An unknown noise is added on each input-target pair in the dataset. Therefore, we did not consider any pixel transformation in our data augmentation pipeline, precisely for preserving the original noise added. Thus, we randomly split the dataset into 4 batches and augmented each batch by one of the following augmentation techniques: vertical flip, horizontal flip, both vertical and horizontal flip and inverse pairing. The latter technique is referring to swapping the input with the target in a pair of images. This method assumes that noisy samples are drawn from the same distribution and it is known as Alternating Noise2Noise technique in [2], yielding improved overall results. Finally, we end up with a training dataset of 100,000 samples.

III. Models and Methods

The core neural network architecture used in our experiments is based on an encoder-decoder structure called UNet [3]. It uses the skip connection between encoders and decoders to enhance the reconstruction process of the image. Several papers proved that this architecture also yield very good results in image denoising tasks ([4], [5]). We investigate several variants of UNet and perform data augmentations together with hyperparameter tuning for improving the baseline.

A. Architecture

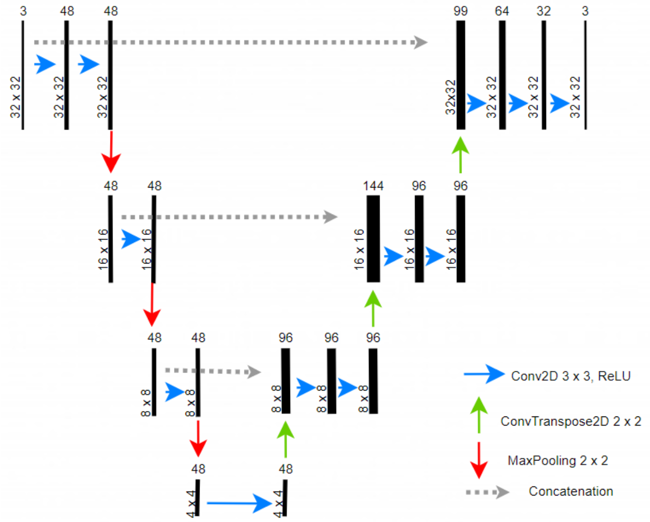

Figure 2: Baseline architecture.

The baseline neural network is using a downsampling factor of 8, resulting in a high-level feature ap of 4x4. This is achieved by using 3 pooling layers. Skip connections are used to concatenate the first feature map from each downsampling block to the first feature map from each upsampling block. Compared to the lighter blocks in the encoder, the ones in the decoder are using 2 convolutions each, because the model must firstly extract useful information from the concatenated feature maps, than pass the combined representation through the next convolution. The final layer is represented by another 3x3 convolution with 3 channels, in order to output an image with the same dimensions as the input one.

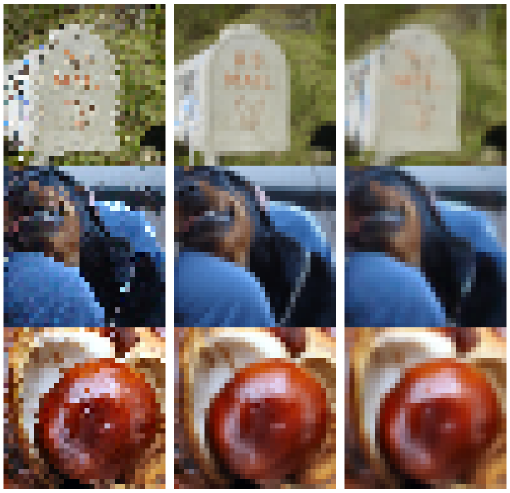
(a) Input (b) Target (c) Prediction

Figure 1: Visualizing model performance on validation samples: (a) - noisy input image, (b) - noisy target image, (c) - predicted image.

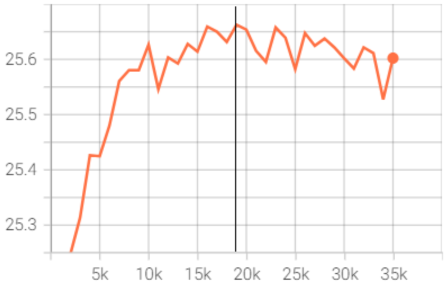

Figure 3: Peak Signal-to-Noise Ratio on validation data. Black line represents the best model, where validation loss is the smallest.

B. Training and evaluation

All models use the Mean Squared Error (MSE) as an objective function and the Adam optimizer with a fixed learning rate of 0.001. In order to avoid overfitting we use early stopping. The training stops if there is no improvement in the validation loss after $nr\_epochs/2$ epochs. In Figure 3 we present the effect of early stopping for choosing the best model. All models were trained on the entire training dataset, in a Google Colab environment, using a remotely provided GPU.

The evaluation metric used is Peak Signal-to-Noise Ratio (PSNR), measured in dB. PSNR measures the quality of an image after the reconstruction. The higher the PSNR, the better the reconstruction or the noise removal. After each epoch we compute the PSNR on the vaildation data, and only save the currently best model. We used TensorBoard in order to track the loss, evaluation metrics and predictions.

C. Results

Table I summarizes the different architectures and augmentation techniques that we used along the experiments.

Model	**Epochs**	**Nr. params.**	**PSNR (dB)**
*UNet**	20	721,395	25.48
UNet	17	721,395	25.54
UNet_NoSkip	18	719,667	22.5
UNet_Upsample	20	534,531	25.51
UNet_32C	19	311,523	25.51
UNet + Aug	22	721,395	25.64
UNet + AugSwap	19	721,395	25.65
UNetSmall + AugSwap	19	410,019	**25.66**

Table I: Results of different denoising architectures, in terms of PSNR, on the validation dataset.

In the table above, $UNet$ refers to the baseline architecture described in Figure 2. $UNet*$ is only using a batch size of 4, while all the other models use a batch size of 100. From the first 2 rows we can see the benefit of using a larger batch size. $UNet\_NoSkip$ does not use any skip connection, yielding a score with 3.04 lower than the baseline. Therefore, the skip connections are highly beneficial for a better image reconstruction. $UNet\_Upsample$ uses only bilinear upsampling layers instead of transposed convolution layers in the decoder branch. Even though the network is lighter, the PSNR score is lower. $UNet + Aug$ uses the 3 types of flips presented in II. We conclude that data augmentation have a high contribution to generalization, as there is an increase of 0.1 PSNR compared to $UNet$. Swapping the input and the target in the data augmentation pipeline ($UNet{+}AugSwap$) also brings a small improvement of 0.01. By reducing the number of filters of the convolutional layers, from 48 to

32, as in $UNet\_32C$, we see a drop of 0.03 compared to $UNet$. The best model, $UNetSmall{+}AugSwap$ is a lighter version of $UNet$. Here we removed on block from the downsampling branch and one block from the upsampling branch. Therefore, the feature map, before upsampling, has a size of 48x8x8 instead of 48x4x4. As input images are already of low resolution, it can bee seen that a too strong downsampling hinders the network from learning.

When training the best model ($UNetSmall + AugSwap$) for only 10 minutes, on a Tesla T4 with 15 GB RAM, we got a score of 25.65 after 12 epochs. One epoch, on the same GPU, takes around 45 seconds. Figure 1 shows some of our results.

IV. Conclusions

Even though the baseline already yielded good results, we showed how data augmentation and different types of architectures could contribute to a better performance, increasing the PSNR score from 25.48 dB to 25.66 dB. As feature experiments we are looking forward to test other types of architectures for image denoising such as transformers, and explore more powerful augmentation techniques such as surrogate Noise2Noise, as proposed in [2].

References

[1] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, “Noise2noise: Learning image restoration without clean data,” 2018. [Online]. Available: https://arxiv.org/abs/1803.04189

[2] A. F. Calvarons, “Improved noise2noise denoising with limited data,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2021, pp. 796–805.

[3] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” 2015. [Online]. Available: https://arxiv.org/abs/1505.04597

[4] J. Gurrola-Ramos, O. Dalmau, and T. E. Alarcón, “A residual dense u-net neural network for image denoising,” *IEEE Access*, vol. 9, pp. 31 742–31 754, 2021.

[5] C.-M. Fan, T.-J. Liu, and K.-H. Liu, “Sunet: Swin transformer unet for image denoising,” 2022. [Online]. Available: https://arxiv.org/abs/2202.14009