Image denoising using Convolutional Auto-encoder

Sirusala Niranth Sai

Electronics Engineering, IIT (BHU)
Varanasi-221005
sirusalansai.ece18@itbhu.ac.in

I. Introduction

Image denoising is a well-studied problem in Computer Vision and Image Processing, as well as a testbed for low-level Image modeling. A **convolutional auto-encoder** was used for this task. The network consists of multiple **convolutional** and **transposed convolutional(deconvolution)** layers, learning a mapping from corrupted images to original images.

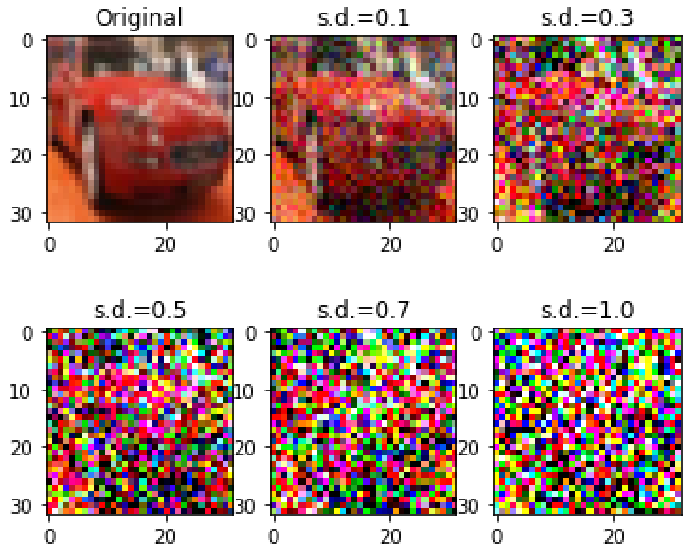
II. My approach

First of all, I searched for denoising and different methods in denoising. I followed an **overview of deep learning used in Image Denoising**[1] and got to know the methods and terminology used. I found the approach given in [2] interesting as they used **skip connections** to pass data from convolutional to deconvolutional layers. The feature maps passed through these skip connections carried image detail and helped deconvolution layers to **recover** an **improved cleaner version** of the **image**. These skip connections were also useful in **avoiding** the problem of **gradient vanishing** in deep networks. The author has put their code[3] written using the Caffe framework. Referring to that network architecture I made my code in Keras.

III. Experimentation

I have done various experiments using Gaussian noise with **standard deviation** values of **0.1, 0.3, 0.5, 0.7, 1.0** and have presented the results in this pdf.

A sample image with different noise levels is shown in Fig-1



(Fig-1)

- **Peak signal-to-noise ratio(PSNR)** was used to compare the performance of different models.
- PSNR was calculated using:

  psnr = 20 * log10(max_pixel / sqrt(mse))

A. Comparison of performance with and without skip connections

A model with **10 convolutional** layers and a gaussian noise of standard deviation of **0.7** was used to test

(Fig-2a) shows the loss of both the models

(Fig-2b) shows the performance of both the models

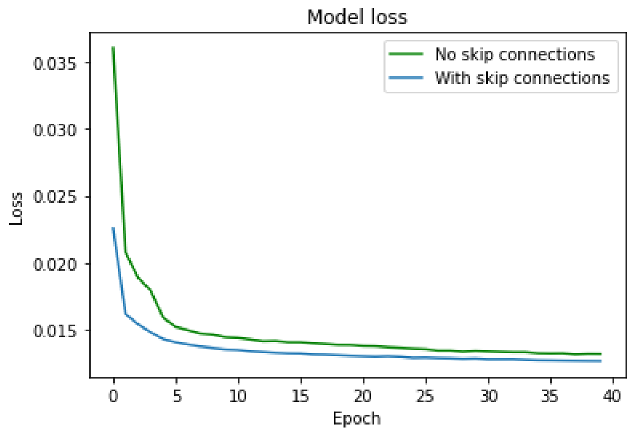


Fig-2a

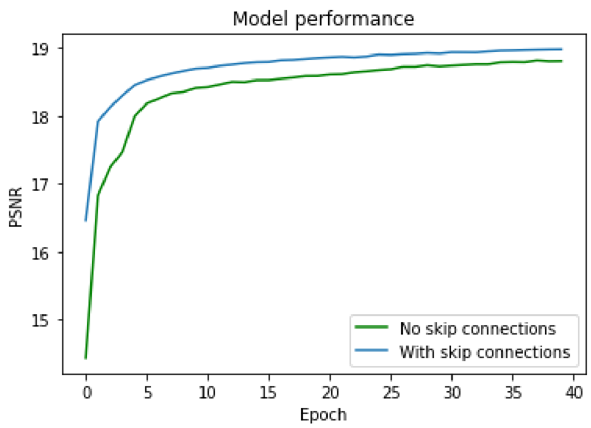


Fig-2b

The model with skip connections shows a better performance than the model without skip connections.

B. Comparison of performance of the model with 5 Convolutional layers and 10 Convolution layers:

The performance of the model with **10** convolutional layered network was **better** than that of **5** convolutional layered network but its training time was almost double than that of the later network.
It is a **tradeoff** between training time and performance.
Model performance for 5 and 10 convolutional layers is shown in (Fig-3a)
Training time for both models is shown in (Fig-3b)

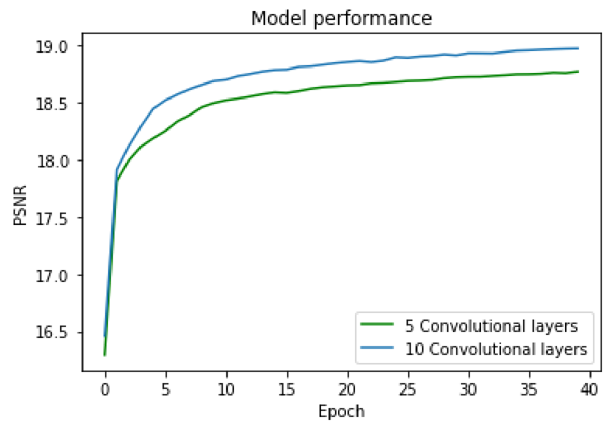


Fig-3a

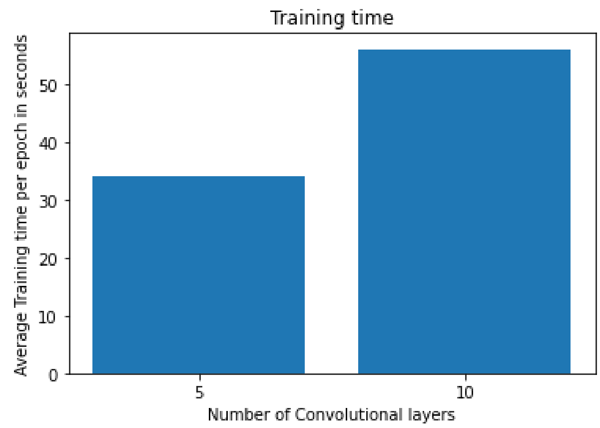


Fig-3b

C. PSNR for different values of standard deviation

A 10 convolutional layered network is used here

Standard Deviation	PSNR
0.1	28.33
0.3	22.81
0.5	20.42
0.7	18.98
1.0	15.35

Model performance for **different standard deviations** is shown in (Fig-4)

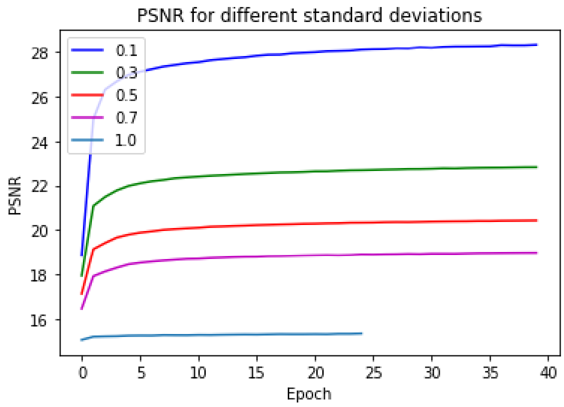


Fig-4

D. Denoised Images

Original, noisy and denoised images when the standard deviation is 0.1 are shown in (Fig-5)

Original, noisy and denoised images when the standard deviation is 0.3 are shown in (Fig-6)
Original, noisy and denoised images when the standard deviation is 0.5 are shown in (Fig-7)
Original, noisy and denoised images when the standard deviation is 0.7 are shown in (Fig-8)
Original, noisy and denoised images when the standard deviation is 1.0 are shown in (Fig-9)

References

[1] Chunwei Tian et al., Deep Learning on Image Denoising: An Overview

[2] iao-Jiao Mao et al., Image Restoration Using Convolutional Auto-encoders with Symmetric Skip Connections.

[3] https://bitbucket.org/chhshen/image-denoising/src/master/model/RED Net_ch1.prototxt

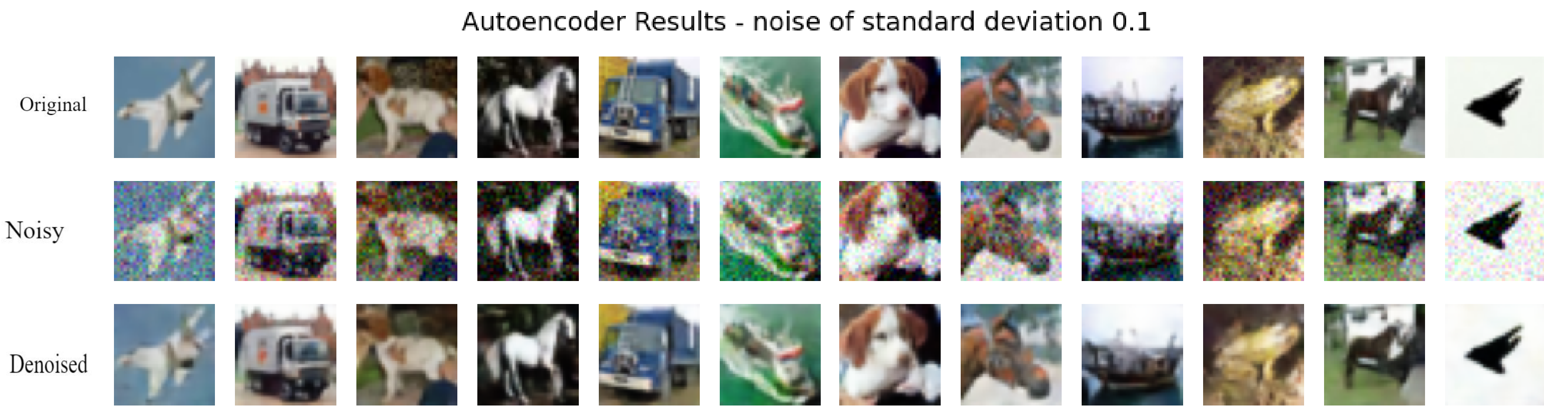


Fig-5

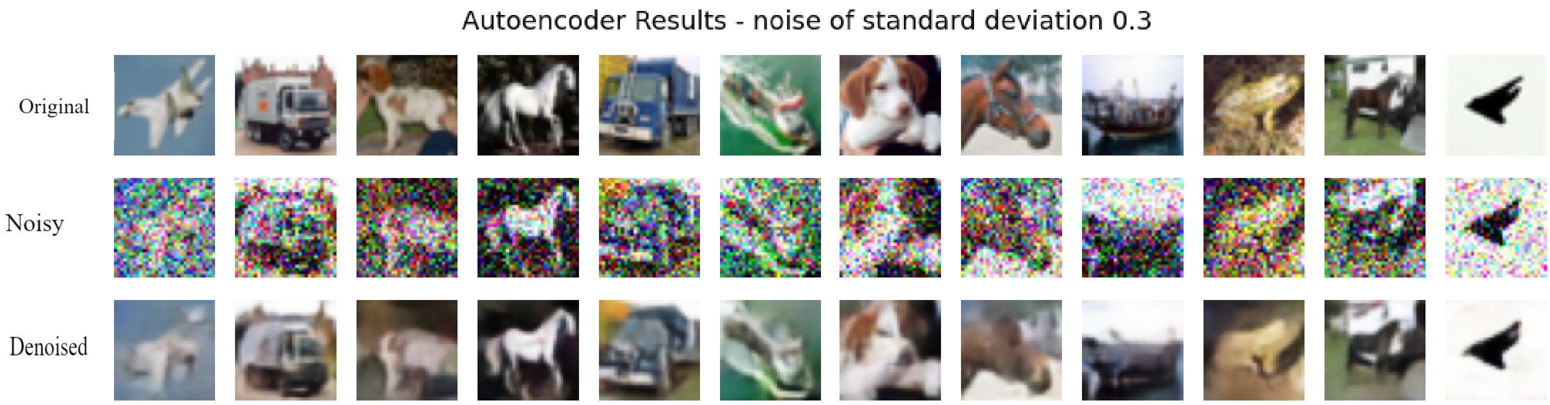


Fig-6

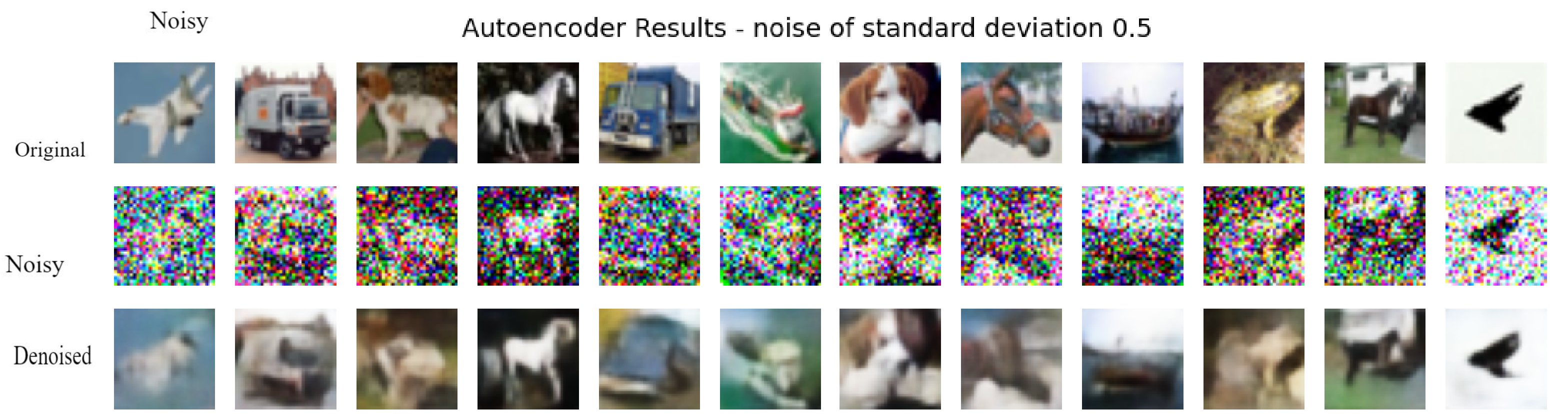


Fig-7

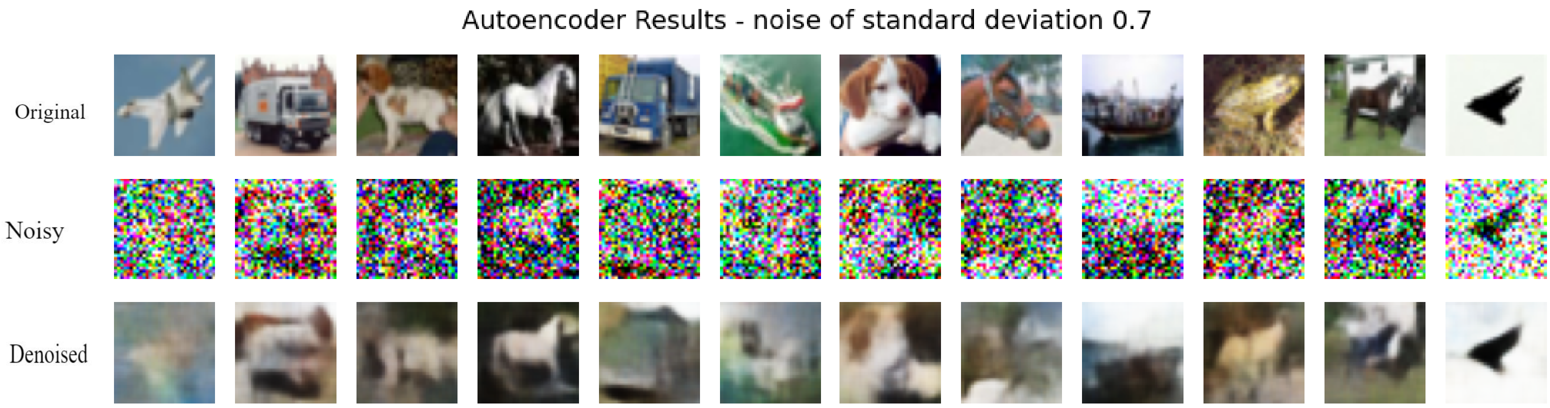


Fig-8

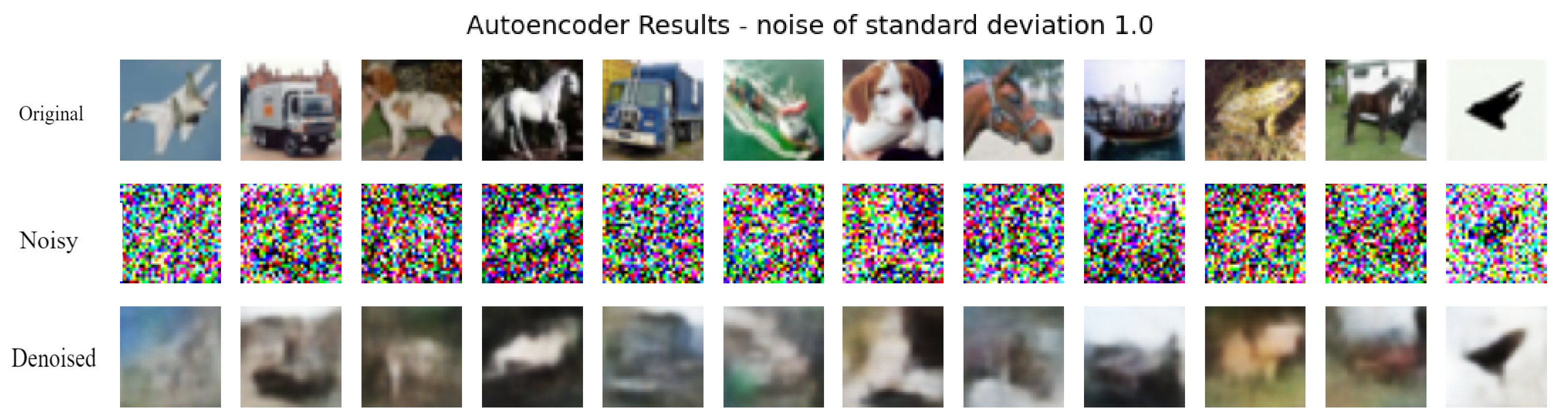


Fig-9