Image Denoising using Deep Learning

1. **Some types of Noises in images**
   i. Gaussian Noise
   ii. Salt and Pepper Noise
   iii. Speckle noise

   For this task, Gaussian noise with mean 0 and standard deviation 0.1 is used.
   CIFAR10 dataset with 50,000 training images and 10,000 test images has been used.

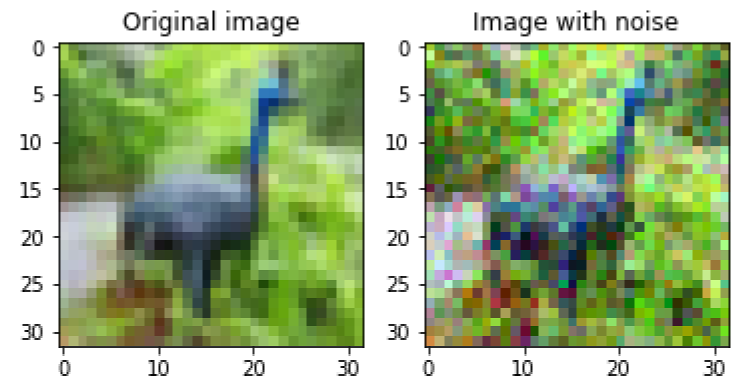

Fig. 1: On left: Original image from CIFAR10, On right: Image with Gaussian Noise

2. **Denoising algorithms**
   Autoencoders have been used for image denoising. Four different architectures of autoencoders were experimented with.

i. Denoising Autoencoder

Fig 2. shows the architecture of the autoencoder used.
In order to evaluate the performance of the model in denoising images, Mean Square Error of the denoised and original data was compared with the Mean Squared Error of noisy and original data.

MSE of noisy and original data: `0.009301587876863535`
MSE of denoised and original data: `0.002526932`
The model was trained for 40 epochs with ADAM optimizer and MSE loss.
It reduced the noise by around 72.8%.
Implementation:
https://github.com/Prachi-Agr/Image-Denoising/blob/master/Denoising_Autoencoder.ipynb

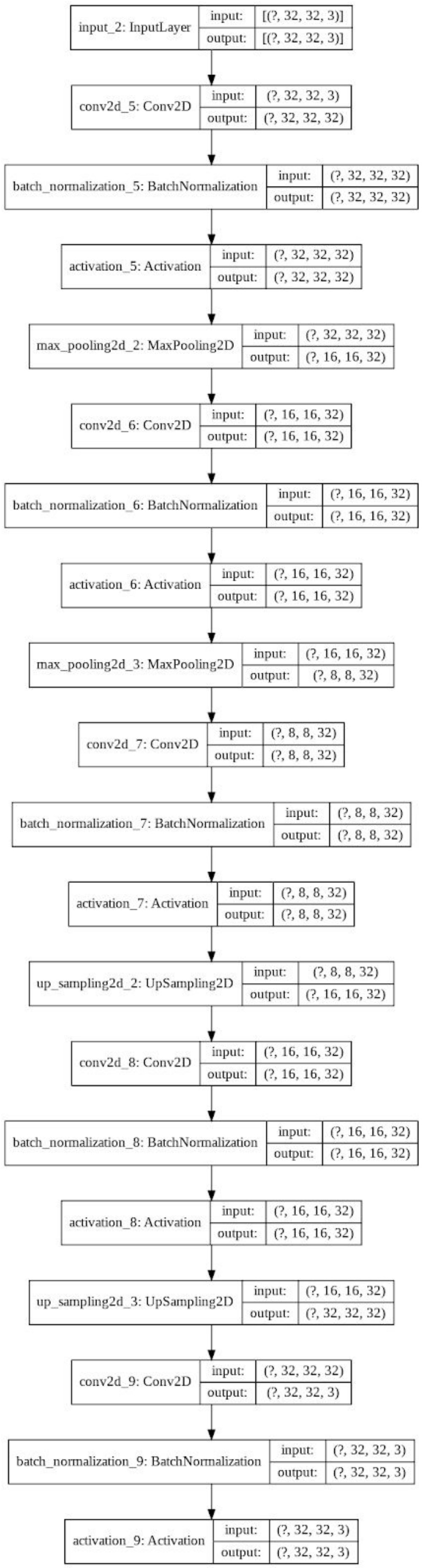

Fig 3: Architecture of Denoising Autoencoder

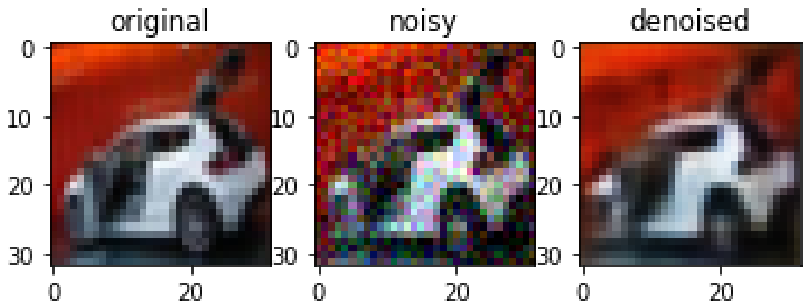

Fig 3: Visual Analysis of original, noisy and denoised images

ii. UNet Autoencoder

Fig 5 shows the architecture which consists of 4 convolutional blocks with downsampling followed by a convolutional block without downsampling. This is followed by 4 deconvolutional blocks with upsamping and 1 deconvolutional block without upsampling.

MSE of noisy and original data: `0.00930624888302536`
MSE of denoised and original data: `0.0014075692`
It reduced the noise by around 84.8%.

Implementation:
https://github.com/Prachi-Agr/Image-Denoising/blob/master/UNet_Autoencoder.ipynb

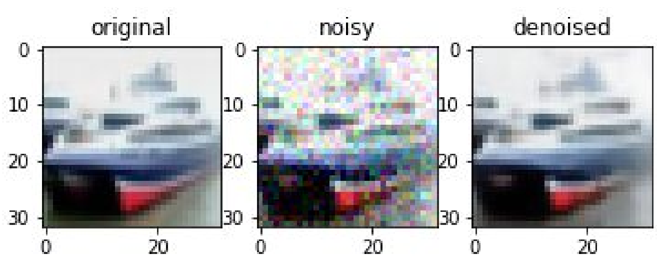

Fig 4 :Visual Analysis of original, noisy and denoised images

Fig 5: UNet Autoencoder Architecture

iii. **Modified UNet Autoencoder**
The architecture is similar to the UNet autoencoder but does not use Max Pooling layers for downsampling. Additionally, it has batch normalization layers. The architecture is shown in Fig 7 .

MSE of noisy and original data: `0.009304892998598215`
MSE of denoised and original data: `0.0015104393`
It reduced the noise by around 83.7%.

Implementation:
https://github.com/Prachi-Agr/Image-Denoising/blob/master/Modified_UNet.ipynb

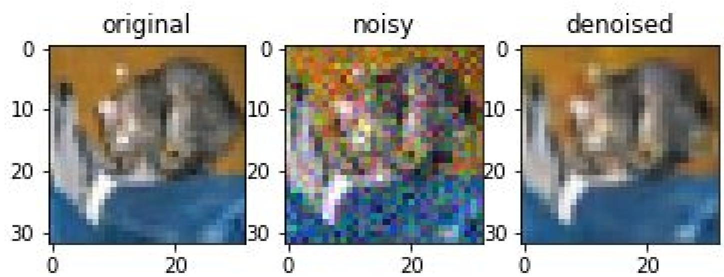

Fig 6: Visual Analysis of original, noisy and denoised images

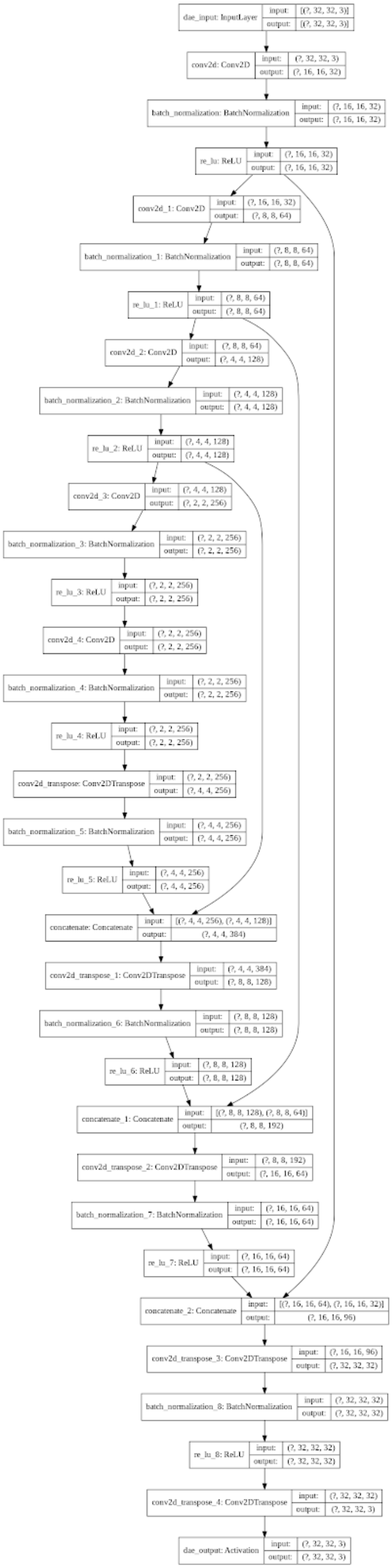

Fig 7: Modified UNet Autoencoder

iv. Autoencoder with Skip connection

The architecture has one skip connection as shown in Fig 9.

MSE of noisy and original data: `0.009305910325287093`
MSE of denoised and original data: `0.0022548838`

It reduced the noise by around 75.81%.

Implementation:
https://github.com/Prachi-Agr/Image-Denoising/blob/master/Skipconnections.ipynb

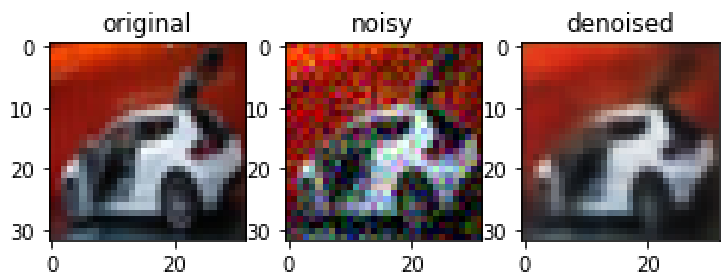

Fig 8: Visual Analysis of original, noisy and denoised images

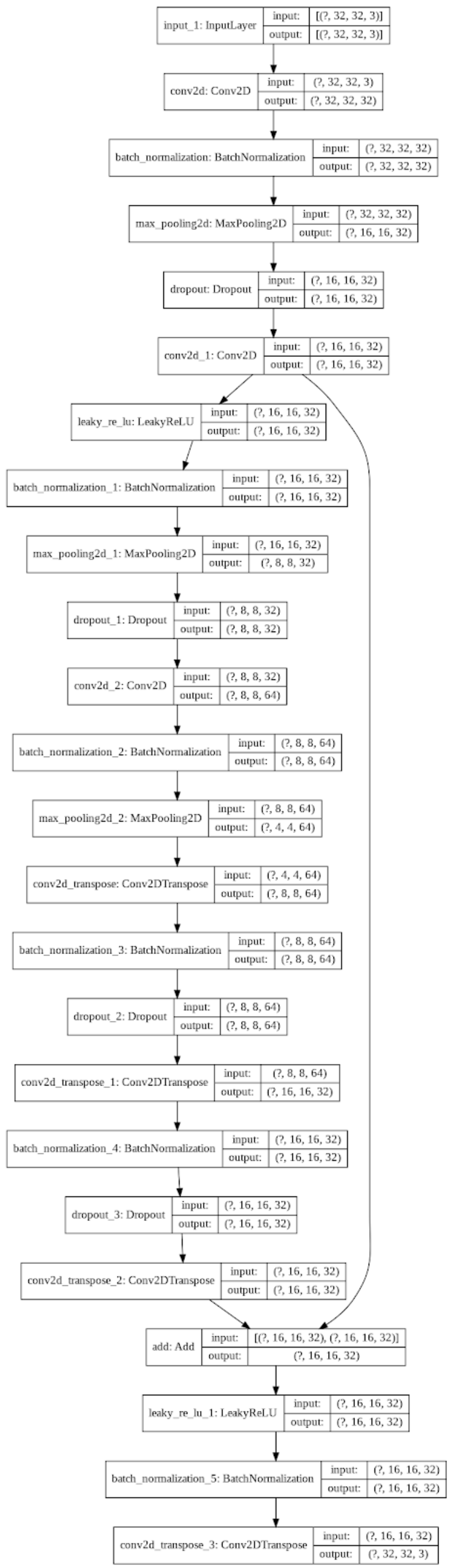

Fig 9: Architecture of autoencoder with skip connection

Conclusion

Out of the 4 architectures implemented, the best denoising performance was of the UNet autoencoder.