Inverse Imaging: Reconstructing High-Resolution Images from Degraded Images

Samir Mishra, Mrunal Chide, Aishwarya Manmode,
Rohit Pandita and Mansi Bhonsle

April 27, 2022

Inverse Imaging: Reconstructing high-resolution images from degraded images

Samir Mishra, Mrunal Chide, Aishwarya Manmode, Rohit Pandita

Mrs Mansi Bhosle
G H Raisoni College of Engineering and Management, Pune

ABSTRACT

Image Denoising simply follows the U-Net architecture which uses images with noise. In this work, we show that u-net architecture, based on convolutional and deconvolutional or transpose convolutional neural networks does a pretty good job in removing noise from the image. The task belongs to a general class of problems on Posterior probability distribution that is the probability of the parameter theta given the evidence X: P(theta | X).

INTRODUCTION

Denoising Image or basically removing noise or disturbance from the image and producing a good quality image is quite a wonderful task that can be achieved by deep learning or formally generality of Convolutional Neural Networks. Deep learning has not only proved to be efficient but also powerful for solving many such tasks. Denoising imaging can produce a great impact by giving a high resolution of the image containing some noise which can give more clarity without losing the generality of the image.

In the proposed system, neural network techniques such as Convolution Neural Networks, MaxPooling, dropout, Concatenation and Lambda Layers are used. The model adopts the U-Net architecture and has approximately 15 million trainable parameters.

Convolutional Neural Network or CNNs has established a state-of-the-art approach in image processing and pattern recognition problems such as face recognition, object detection and other computer vision application.

CNNs are regularised versions of multilayer perceptrons. Multilayer perceptrons or MLP contains many perceptrons divided into many layers where each artificial neuron or perceptron is connected to all other neurons in the next layer i.e., fully connected networks. It consists of an input layer, hidden layers and an output layer. In any neural network architecture, middle layers are called hidden because their inputs and outputs are masked by the activation function. In a convolutional neural network, the hidden layers include layers that perform convolutions.

Here CNNs are used to solve inverse problems such as denoising, super-resolution. Inverse problems are basically the reverse of forwarding problems i.e., calculating from a set of observations the factors that produced them.

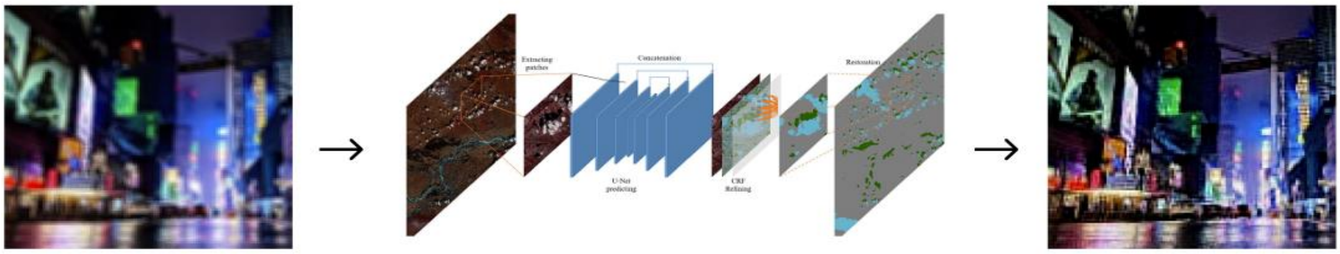


Figure 1. Denoising of the image using our model consisting of deep CNN, MaxPooling2D, Dropout, concatenation and Lambda layers. Here, the initial layer performs extension of patches, hidden layers form unit architecture and skip connections and end layers performs Restoration

In this work, convolutional and deconvolutional are performed on the images. Deconvolutional or Transpose Convolutional neural networks basically performs the inverse of the convolutional method. In CNN, we reduce the height and width of the image whereas in deconvolution we are increasing the height and width.

MaxPooling is basically used for downsampling of the image representation, causing the reduction of the dimensions of input data by combining the outputs of the groups of neurons at one layer into a single layer in the next immediate layer.

Dropouts or dilution is a regularisation technique for reducing overfitting by thinning the weights or randomly dropping out units (hidden and visible) during the training process of the network.

Concatenation is a neural network technique in which it takes input numbers of tensors and concatenates them into a single tensor. The concatenation layer basically is used for skip connection which basically improves the feature processing and gives a better efficient result.

Lambda layers are used to wrap arbitrary expressions as a layer object. It is used in the model to give the model a better generality in terms of image resizing.

Related Work

Over the past few decades, considerable methods have been studied for denoising the images. Among all supervised deep convolutional networks and deep images prior have gained popularity.

A supervised deep convolutional network requires a large set of training data. Also, if the image to be denoised is significantly different from the training images then it might lead to inferior results and may also create hallucinations.

Deep image Prior(DIP) overcomes this drawback to some extent. DIP is capable of capturing the low-level statistics of the natural image using an unsupervised learning model that does not require training images other than the image itself. Also, it is more flexible. Nevertheless, the accuracy of DIP is usually inferior to the supervised learning-based methods using deep convolutional neural networks and is also susceptible to over-fitting problems.

To solve this problem of DIP, a paper proposed a novel deep generative network with multiple target images and an adaptive termination condition. Specifically, they utilized mainstream denoising methods to generate two clear target images to be used with the original noisy image, enabling better guidance during the convergence process and improving the convergence speed. Moreover, they adopted the noise level estimation (NLE) technique to set a more reasonable adaptive termination condition, which can effectively solve the problem of over-fitting.

Although this approach requires a large number of gradient updates, resulting in long inference times. Thus, its execution efficiency is relatively low.

METHODOLOGY

In denoising, we are given an image as input. The image may contain a certain amount of noise (or may not) embedded in it, our task is to reduce the noise from the image as much as possible.

In this paper, we propose a neural framework based on U-Net architecture and skip connections resulting from the concatenation of layers. Skip Connections forwards the essential features of the image which results in better processing of the images. To simply consider, noised images are basically concatenation of latent images and noises. At the start of layers, the input data carries with it the necessary features and patterns that describe the latent image. These features thus provide the necessity to generate images with the generality of lossless composition.

In our model, we have maintained this feature throughout and processed them in an orderly fashion of what U-Net architecture suggests.

1. Working on Datasets and creating labels.

For the first task, we need to capture the dataset to work on. Since the task of denoising does not need some specific dataset as it is the general problem of noise reduction, thus no matter what image dataset is used we can define the base of the problem.

Therefore the choice of the dataset is just reduced to choosing a dataset that only contains images and can also depend on personal interest. We chose the flower dataset, again (for no specific reasons) as they are readily available everywhere.

The second step is to decode the images from the flower dataset which can be easily done with the help of libraries such as OpenCV and TensorFlow.

The third step was to divide the dataset such that we have labels. Since our model is based on Supervise technique to produce high-resolution images, we need labels to map the inputs with their correct outputs.

Thus we created a list of true images and blurred images. The true images list contains a list of the true image of the flower dataset and the blurred images list contains the same images that are stored in true images but only the difference is the images in this list are introduced with some noise. We have a function in the OpenCV library itself to introduce noise called Gaussian Blur which generally need four arguments first being the image and second the Gaussian kernel size, third the standard deviation in the X direction and fourth the standard deviation in the Y direction. Now then we have created the lists we can divide them into inputs and labels. The true images list is labelled for the blurred images list, but first, we convert lists into respective arrays.

Now after we are done with dataset processing, we move on to building the model.

2. Working on the model

Here we are using U-Net architecture. U-Net architecture has 3 building blocks: convolutional blocks, middle layers and transpose convolutional or deconvolutional blocks. Each Conv2D and Conv2DTranspose used padding as “same” and activation function as “relu”, kernel size is (3, 3) everywhere.

Convolutional blocks consist of a series of Convolutional layers or Conv2D and then concatenation or skip connections and MaxPooling2D. Conv2D layer is input to the Conv2D layer which in turn is input to the next Conv2D and then concatenated with the first Con2d layer of this sequence and this repeats two to three times.

Deconvolutional blocks consist of a series of Conv2dTranspose, skip nets and Dropout and this is also repeated two to three times.

The middle layer basically is the Conv2D layer repeated two to three times.

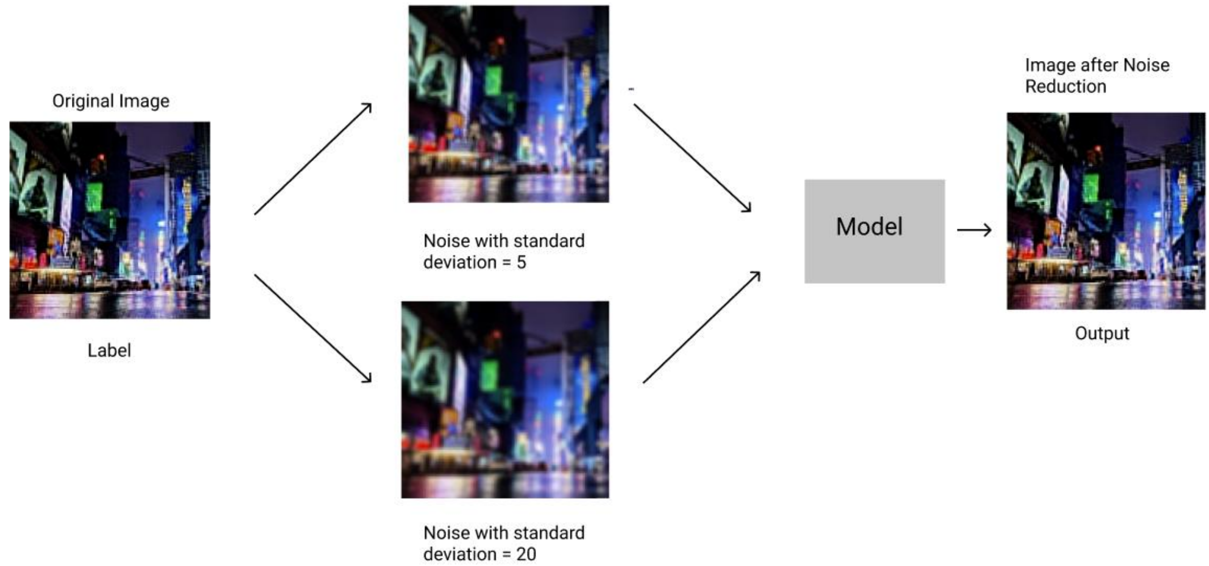
We have taken input shape to be as (128, 128, 3). Then the input layer is passed through the convolutional block which performs the extension of patches, then the output of this is passed as input to the middle blocks which are basically layers of CNNs and meanwhile using skip connections to preserve the generality. The output of the middle layer is then passed as input to the Deconvolutional block which performs Transpose Convolutional or Deconvolution of the input and generates the output which is then checked with the label.

We also made use of the skip connection in the deconvolutional block. As we talk about the shallow layers or layers which are not deep in the network consist of rich features about the image such as corners, edges, diffusion of colours, etc. But as we go deeper in the network, layers tend to learn different interesting features, by using skip connection we allow our network not just to learn those new features but also carry on the rich features about the image giving greater power to our network.

The output of our deconvolutional block is an image with the same shape as our image i.e. (128, 128, 3)

The model seems to do a very good job of matching the patterns to reduce the noise. The generality is achieved in conjunction with Mean Squared Error(MSE) loss function and Adam optimizer to achieve momentum regularization.

One great advantage of our model is the absolute result. No matter how deep the noise is introduced in the image the noise reduction does its job the same every time.



Here, we have divided the representation into 3 parts, 1st part shows the label image that is the true image, 2nd part shows the true image induced with different intensity of noises, 3rd part shows that when given noised image to our model it produces the absolute same results.

RESULTS

The work resulted in an accuracy of 92.01% on validation data and 85.09% accuracy on train data where train data consisted of 90% of 100 images and validation data consisted remaining 10% of 100 images.

The model was trained for 500 epochs and performed very good in producing high-resolution images with maintaining the generality of the image.

CONCLUSION

Thus, we show that the supervised deep learning technique suffixes in denoising images and producing high-resolution images without losing the generality of the image. Here we tried to show the power of deep learning in reference to computer vision application on inverse problems and to show that this general representation of state of art technology can also be extended in the proper reconstruction of degraded images

REFERENCES

1. Keiron O'Shea 1 and Ryan Nash 2: An Introduction to Convolutional Neural Networks
2. Wikipedia: Convolutional neural network.
3. Indra Deep Mastan and Shanmuganathan Raman: Multi-level Encoder-Decoder Architectures for Image Restoration.
4. Shady Abu-Hussein, Tom Tirer, Se Young Chun, Yonina C. Eldar, Raja Giryes: Image Restoration by Deep Projected GSURE
5. Shiming Chen, Shaoping Xu, Xiaoguo Chen and Fen Li: Image Denoising Using a Novel Deep Generative Network with Multiple Target Images and Adaptive Termination Condition
6. Dmitry Ulyanov, Andrea Vedaldi, Victor Lempitsky: Deep Image Prior
7. Bristow, H., Eriksson, A.P., Lucey, S.: Fast convolutional sparse coding. In: CVPR, pp. 391–398. IEEE Computer Society (2013)
8. Buades, A., Coll, B., Morel, J.M.: A non-local algorithm for image denoising. In: Proc. CVPR, vol. 2, pp. 60–65. IEEE Computer Society (2005)
9. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. In: NIPS, pp. 658–666 (2016)
10. Dosovitskiy, A., Brox, T.: Inverting convolutional networks with convolutional networks. In: CVPR. IEEE Computer Society (2016)
11. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: Proc. ICCV, pp. 349–356 (2009)
12. Michael T. McCann Convolutional Neural Networks for Inverse Problems in Imaging: A Review
13. Alice Lucas; Michael Iliadis; Rafael Molina; Aggelos K. Katsaggelos Using Deep Neural Networks for Inverse Problems in Imaging: Beyond Analytical Methods
14. Gregory Ongie; Ajil Jalal; Christopher A. Metzler; Richard G. Baraniuk Deep Learning Techniques for Inverse Problems in Imaging
15. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Proc. NIPS, pp. 2672– 2680 (2014)
16. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires rethinking generalization. In: ICLR (2017)