INTERNATIONAL JOURNAL
FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

www.ijraset.com

Call: ⓒ08813907089 | E-mail ID: ijraset@gmail.com

Imagify: Reconstruction of High-Resolution Images from Degraded Images

Chris Lopes[1], Pavan Patel[2], Rutwek Hirwe[3], Piyush Sonawane[4], Apashabi Pathan[5]

Department of Information Technology, G.H Raisoni College of Engineering and Management, Pune, India

Abstract: Image Denoising simply uses photos with noise and the U-Net architecture. In this study, we demonstrate that the u-net design, which is based on convolutional and deconvolutional neural networks or transpose convolutional neural networks, is quite effective at reducing image noise. The assignment falls under abroad category of problems on distribution of posterior probabilities P(theta | X) represents the probability of the parameter theta given the evidence X. These techniques yield great results, but their practical application might be problematic due to issues like the expensive forward and adjoint operators' computations and the challenging hyper parameter selection.

Keywords: Convolutional Neural Networks(CNN), Denoising.

I. INTROCUTION

Deep learning, or more specifically the formal generality of Convolutional Neural Networks, is quite good at doing the fantastic task of denoising images, or basically removing noise or disturbance from the image and producing a decent quality image. For completing many of these jobs, deep learning has not only shown to be effective but also strong. Denoising imaging can have a significant influence by providing an image with a high resolution that contains some noise and can add additional clarity without sacrificing the image's overall generality. Neural network techniques like Convolution Neural Networks, Max Pooling, dropout, concatenation,and Lambda Layers are used in the suggested system. The U-Net architecture is used in the model, which contains roughly 15 million trainable parameters. Convolutional Neural Networks, also known as CNNs, have become the most advanced method for solving image processing and pattern identification issues including face and object recognition, among other computer vision applications. Multilayer perceptron's are modified into CNNs. In multilayer perceptron, or MLPs, each artificial neuron or perceptron is connected to every other neuron in the layer below it, forming completely connected networks. Multilayer perceptron's are made up of numerous perceptron's that are separated into several layers. An input layer, hidden layers, and an output layer make up this structure. The middle layers in any neural network architecture are known as hidden layers because the activation function hides their inputs and outputs. The hidden layers in a convolutional neural network contain convolutional layers. In this situation, CNNs are used to address inverse issues like denoising and super-resolution. Inverse problemsare essentially the opposite of forwarding problems in that they involve determining the causes of a set of observations.
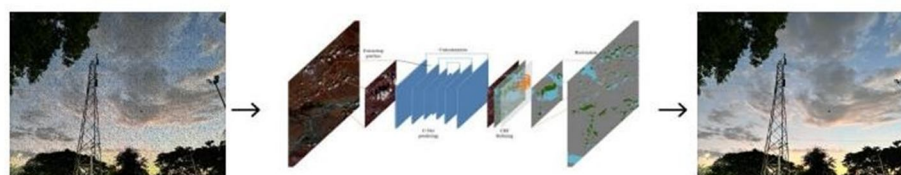

Figure 1. Denoising of the image using our model consisting of deep CNN, MaxPooling2D, Dropout, concatenation and Lambda layers. Here, the initial layer performs extension of patches, hidden layers form unitarchitecture and skip connections and end layers performs Restoration

The images in this work are subjected to convolutional and deconvolutional operations. Transposition or Deconvolution Convolutional neural networks essentially carry out the convolutional method's opposite operation. At CNN, we minimise Deconvolution increases the height and width of the image rather than decreasing them. By merging the outputsof the groups of neurons at one layer into a single layer in the next immediate layer, MaxPooling is mostly utilised for downsampling of the image representation, resulting in the decrease of the dimensions of input data.

A regularisation technique called dropouts or dilution thins the weights or randomly removes units (both hiddenand visible) from the network during training in order to reduce overfitting.

A neural network technique called concatenation joins together multiple input tensors into a single tensor. The concatenation layer is mostly utilised for skip connections, which enhance feature processing and produce more effectively. Any expression can be wrapped as a layer object using lambda layers. It is incorporated into the model to improve its generality with regard to image scaling.

II. METHODOLOGY

An image is provided as input for denoising. Our goal is to remove as much noise as we can from the image, which may or may not have a specific level of noise embedded in it.

In this paper, we provide a neural framework built on a concatenation of layers-based U-Net architecture with skip connections. Skip Connections transmits the image's key components, resulting in better image processing. Noised pictures are essentially a concatenation of latent images and noises, to put it simply. The necessary features and patterns that describe the latent image are present in the input data at the beginning of each layer. The requirement to generate images with the generality of lossless composition is thus provided by these features.

As suggested by U-Net design, we have kept this feature throughout our model and processed them in an orderly manner.

A. Working on Datasets and Creating Labels

We must collect the dataset to work with for the first task. No matter what picture dataset is used, we can define the problem's fundamentals because the process of denoising does not require a specific dataset because noise reduction is a general problem.

As a result, selecting a dataset simply including photographs is all that is left, and it may also be influenced by personal interest. Again, we chose the flower dataset (for no particular reason), as they are widely accessible.

The second stage is to decode the flower dataset's image data, which can be done quickly with the aid of frameworks like OpenCV and TensorFlow. The dataset was divided in the third phase so that we could add labels. We require labels in order to map the inputs to the proper outputs because our model relies on the Supervise technique to generate high-resolution images.

As a result, we compiled a collection of clear and fuzzy photos. The blurred images list contains the same images that are stored in the true images list with the exception that some noise has been added to the images in this list. The true images list contains a list of the true images of the flower dataset.

The OpenCV library includes a function called Gaussian Blur that can be used to add noise. This function typically requires four arguments: the image, the size of the Gaussian kernel, the standard deviation in both the X and Y directions, and the image itself. Once the lists have been made, we can separate them into inputs and labels. We first convert lists into corresponding arrays before labelling the true images list for the blurred images list.

Now that the dataset processing is complete, we proceed to building the model.

B. Working on the Model

Here, U-Net architecture is being used. Convolutional blocks, middle layers, and transposed convolutional or deconvolutional blocks are the three components that make up the U-Net architecture. Everywhere, the kernel size is (3, 3), and each Conv2D and Conv2DTranspose used padding as "same" and activation function as "relu".

Convolutional blocks are made up of many Convolutional layers, also known as Conv2D, followed by concatenation, skip connections, and MaxPooling2D. Conv2D layer is input into Conv2D layer, which is input into Conv2D layer, which is input into Conv2D layer, and so on. This process is repeated two to three times. A succession of Conv2dTranspose, skip nets, and Dropout are used in deconvolutional blocks, which are repeated two to three times.

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

We used input to be as follows: (128, 128, 3). The output of this is then passed as input to the middle blocks, which are essentially layers of CNNs, while using skip connections to maintain generality. Next, the input layer is passed through the convolutional block, which performs the extension of patches. The Deconvolutional block uses the middle layer's output as input and performs Transpose Convolutional or Deconvolution on it to produce an output that is then checked against 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).

When it comes to matching the patterns to lessen the noise, the model appears to be highly effective. Momentum regularisation is made generic by using the Mean Squared Error (MSE) loss function and the Adamoptimizer.

Our model has the absolute result as a major benefit. No matter how much noise is added to the image, noise reduction always accomplishes its task.
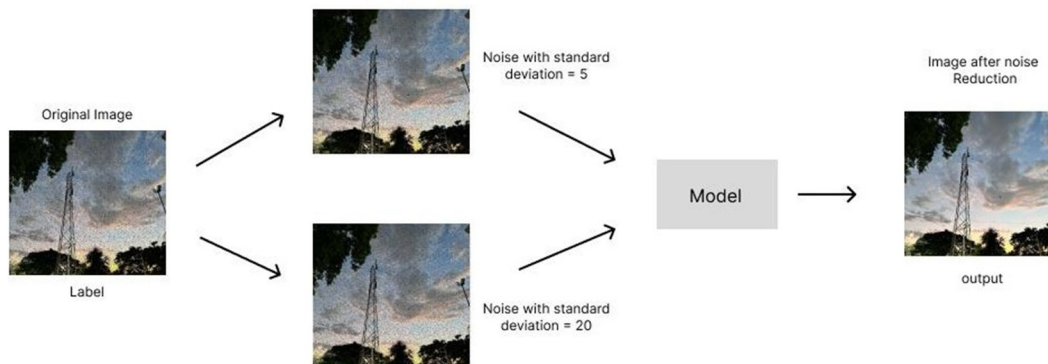

Figure 2. Here, we have divided the representation into 3 parts, 1st part shows the label image that is the true image, 2nd part shows the true image induced with different intensity of noises, 3rd part shows that when givennoised image to our model it produces the absolute same results.

III. RESULTS

The study produced an accuracy of 92.01% for the validation data and an accuracy of 85.09% for the train data,where the train data was made up of 90% of the 100 photos and the validation data was made up of the remaining 10%.

The model produced high-resolution photos while keeping the generality of the image quite well after 500iterations of training.

IV. CONCLUSION

We demonstrate that the supervised deep learning method is sufficient for image denoising and producing high- resolution images without sacrificing the generality of the image. Here, we aimed to demonstrate the effectiveness of deep learning in the context of inverse problems in computer vision and to demonstrate that thisgeneral illustration of cutting-edge technology can also be applied in the correct restoration 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 ImageRestoration.
[4] Shady Abu-Hussein, Tom Tirer, Se Young Chun, Yonina C. Eldar, Raja Giryes: Image Restoration by DeepProjected GSURE
[5] Shiming Chen, Shaoping Xu, Xiaoguo Chen and Fen Li: Image Denoising Using a Novel Deep GenerativeNetwork 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. IEEEComputer 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. IEEEComputer 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 forInverse 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 rethinkinggeneralization. In: ICLR (2017)

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)