



e-ISSN: 2319-8753 | p-ISSN: 2320-6710

# IJRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

**Volume 9, Issue 9, September 2020**

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

**Impact Factor: 7.512**



9940 572 462



6381 907 438



ijirset@gmail.com



www.ijirset.com

# Fast and Flexible Denoising Network using Noise Based Predefined Layers Based on Image Denoising

Smita H. Natikar<sup>1</sup>, Dr. Smitha Sasi<sup>2</sup>

M.Tech Student, Department of Telecommunication Engineering, Dayananda Sagar College of Engineering, Bangalore, India<sup>1</sup>

Associate Professor, Department of Telecommunication Engineering, Dayananda Sagar College of Engineering, Bangalore, India<sup>2</sup>

**ABSTRACT:** Discriminative learning methods have been widely studied in image denoising due to the fast interference and good performance. However these models are required to learn a specific model for each noise level, and it requires multiple models for denoising images with different noise levels. In order to deal with spatially variant noise they lack flexibility, and limits their applications in practical denoising. To overcome these issues, we present a fast and flexible denoising convolutional neural network, namely FFDNet. FFDNet is a recent image denoising method based on convolutional neural network architecture. FFDNet has several desirable properties such as it has the ability to spatially variant noise, smaller memory footprint and the ability to handle a wide range of noise levels with a single network model, and faster speed than benchmark BM3D even on CPU without sacrificing denoising performance. The combined operation of its denoising performance and lower computational load makes this algorithm attractive for practical denoising applications.

**KEYWORDS:** Image denoising, convolutional neural networks, Gaussian noise, salt and pepper noise, spatially variant noise.

## I. INTRODUCTION

Denoising an image is a fundamental task for correcting imperfection during the acquisition process of real world scene and its reproduction on its display, due to physical and technological limitations [1]. The importance of image denoising in low level vision can be known from many aspects. First, noise corruption is unavoidable during the image sensing process and it may decrease the visual quality of acquired image. Removing noise from the observed image is an essential step in various image processing and computer vision tasks [2][3]. Second, we can also evaluate image prior models and optimization method from Bayesian perspective approach [4][5][6].

Image denoising is used to remove additive noise while retaining the signal features. Generally datasets collected by image sensors are contaminated by noise. Data of interest can be corrupted because of imperfect instruments, problems with data acquisition process, and interfering natural phenomena. Thus noise reduction is an important technology in image analysis. For image denoising different algorithms are used depending on the noise model. Most of the natural images are assumed to have additive random noise which is modeled as Gaussian. Speckle noise is observed in ultrasound images whereas Rician noise affects MRI images.

In this paper we adapted a fast and flexible denoising convolutional neural network, namely FFDNet which is able to deal with noise on different levels by taking a tunable noise level map as input. FFDNet works on downsampled sub-images which largely accelerates training and testing speed. FFDNet has several desirable properties such as the ability to handle a wide range of noise levels with a single network, the ability to remove spatially variant noise by specifying a non-uniform noise level map, and faster speed than benchmark BM3D.

Pictures are most common and convenient means of conveying or transmitting information. Pictures give a lot of information about positions, sizes and interrelationships between objects. They describe spatial information in a particular way that we can recognize as objects.

The images are represented in digital form. When the images are represented as numbers, brightness can be added, subtracted, multiplied, divided and in general subjected to statistical manipulations that are not possible if an image is presented only as photograph.

## II. RELATED WORK

### A. System Overview

In general, image denoising methods are divided into two groups, those are model based methods and discriminative learning based ones. Model based methods are further classified as BM3D [9] and WNNM [10]. Discriminative learning based methods can be further classified as MLP, and convolutional neural network based methods.

In our approach we used CNN based method over other methods, why because as compared to Multi Layer Perception (MLP) [11], both MLP and CNN can be used for image classification, where MLP takes vector as input and CNN takes tensor as input. So CNN can understand spatial relation between pixels of images. Thus for complicated images CNN will perform better than MLP.

As compared to RNN, CNN is suitable for spatial data such as images whereas RNN is suitable for sequential data. CNN is a type of artificial neural network with variations of multilayer perceptrons designed to use minimal amount of pre-processing. RNN unlike feed forward neural networks can use their internal memory to process arbitrary sequences of inputs. CNN are ideal for images and video processing whereas RNNs are ideal for text and speech analysis.

### B. Architecture of FFDNet

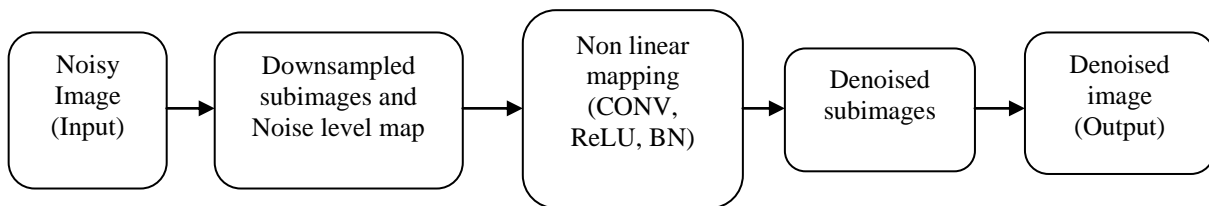


Fig 1. Basic Architecture of FFDNet

The basic architecture of the FFDNet for image denoising as shown in fig 1. Where it takes noisy image (x) as input and fed it to the first layer that is to down sampling and noise level map. In general, down sampling is used to reduce the spatial resolution of while keeping the same 2D representation and is used to reduce the storage. Here the down sampling operation reshapes a noisy image(x) into downsampled sub images; further these downsampled images are added to a tunable noise level map to form a tensor size image as the input to CNN.

With the tensor size as input, the CNN consists of a series of convolution layers. Each layer is composed of three types of operations. Those are convolution (CONV), Rectified Linear Units (ReLU) [7] and Batch Normalization (BN) [8]. The term convolution refers to the mathematical combination of 2 functions to produce a third function. In the case of CNN, the convolution is performed on the input data with the use of a filter to produce a feature map. The ReLU is nothing but a supplementary step to the convolution operation. To increase the stability of a neural network, Batch Normalization (BN) normalizes the output of a previous activation layer by batch mean and bath standard deviation. After performing these operations the resulting image is fed to the up scaling operation which performs the reverse operation of the down sampling operator applied in the input stage to produce clean image and finally we get denoised image as output.

### C. Functional Blocks of Convolution Neural Network (CNN)

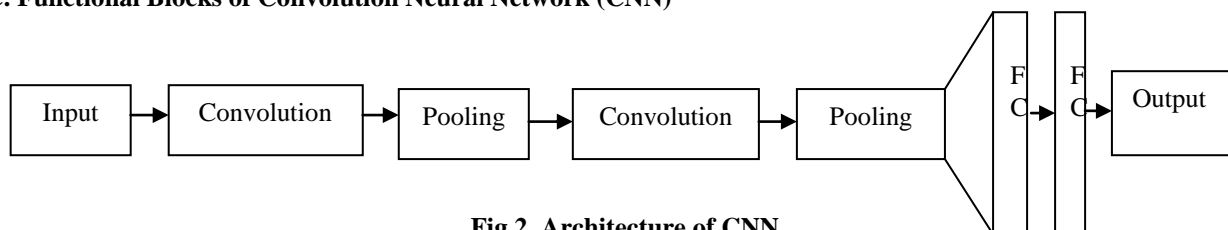


Fig 2. Architecture of CNN

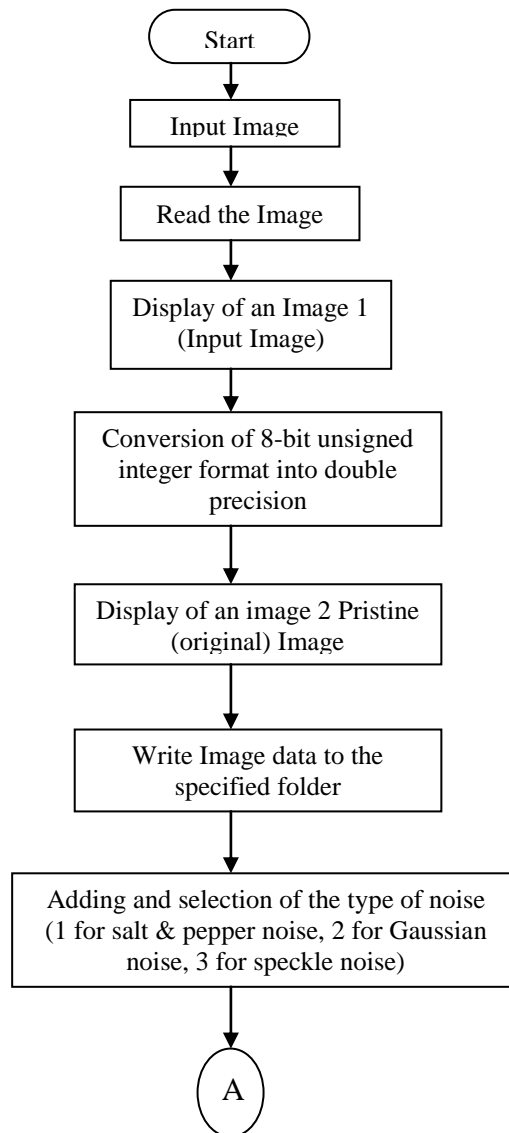


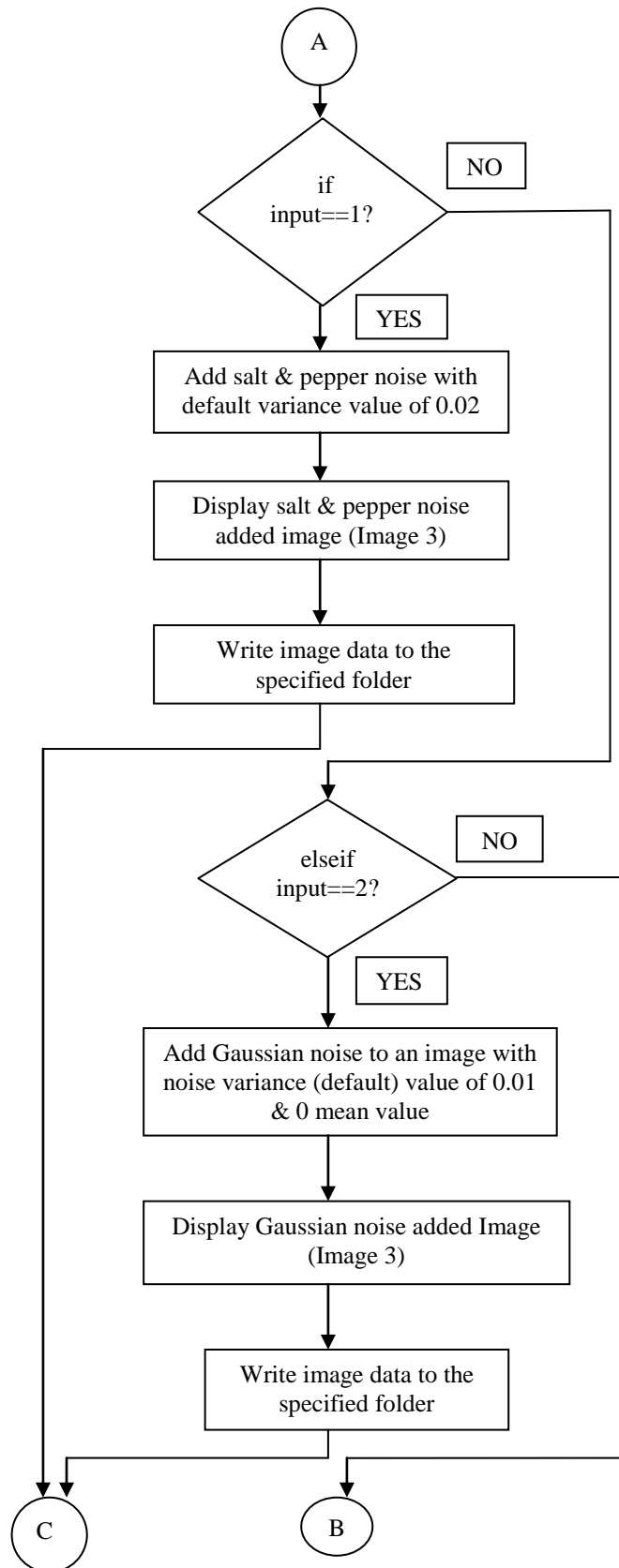
**Convolution:** The main building block of the CNN is the convolution layer. Convolution is a mathematical operation which is used to merge 2 sets of information. Convolution operation is performed on the input data using a convolution filter to produce a feature map.

**Pooling:** After convolution pooling operation will be performed to reduce the dimensionality. Pooling operation enables us to reduce the number of parameters, which both shortens the training time and prevents over fitting. Pooling layer down sample each feature map independently, reduces the height and width. Max pooling is a most common type of pooling which just takes the maximum value in the pooling window.

**Fully Connected (FC):** The output of both convolution and pooling (conv+pooling) layers are 3D volumes, but a FC layer expects a 1D vector of numbers. So the output of a final pooling layer is flattened to a vector and that becomes input to the FC layer.

### III. METHODOLOGY





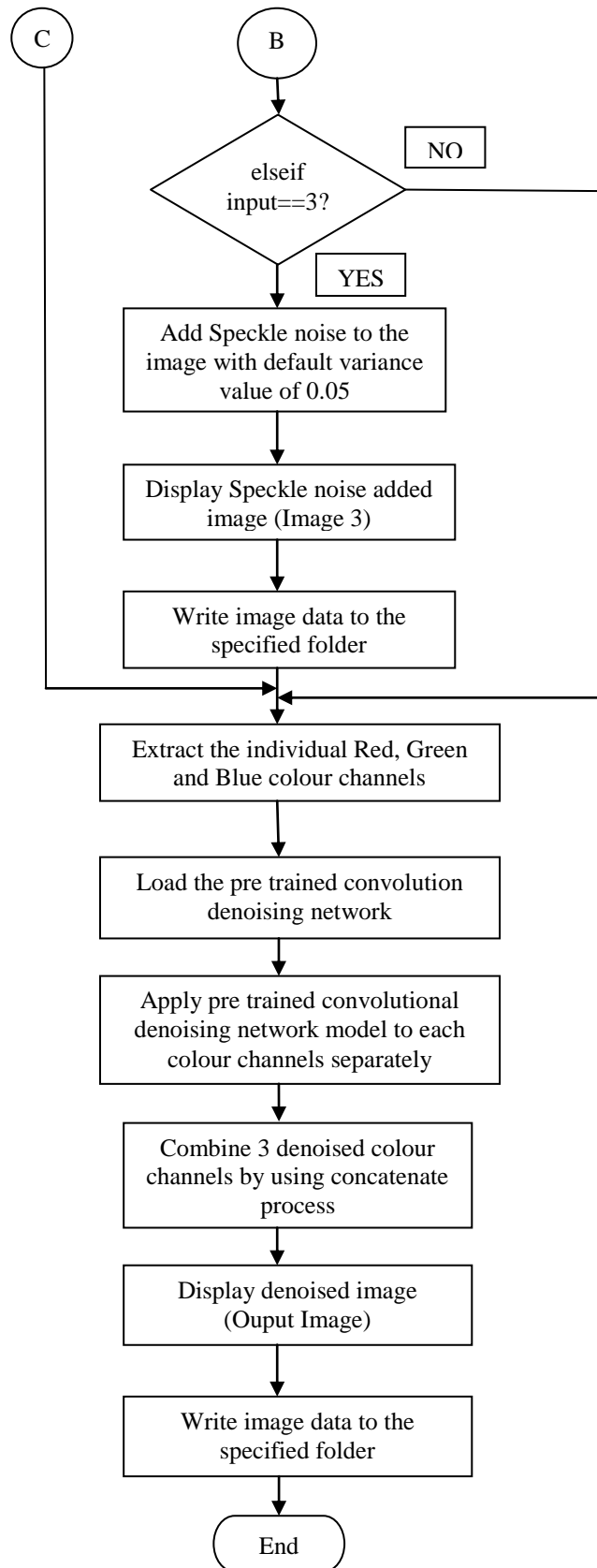


Fig 3. Flowchart of the proposed methodology

The image will be exported from the disk (file) using the command ‘uigetfile ( )’. The exported image will be read using the command ‘imread’. The file name used here is a type of string which contains the complete name of an image. After reading an image it displays the image (input image). The image will be converted to double precision. Image can be grayscale intensity image, a truecolor image, or a binary image, the command ‘imdouble’ rescales the output from integer data types of the range [0, 1]. After the conversion of unsigned integer 8-bit format into double precision it displays the image with the title pristine (original) image. This image data will be written to the file specified by filename in current folder. Naturally images contain some amount of noise. But it’s difficult to denoise. So simulated noise is added to the image to understand the different process of denoising. In our approach we added 3 noise to the image, where input 1 is for salt and pepper noise, input 2 is for Gaussian noise and input 3 is for speckle noise. If the input is 1 it adds salt and pepper noise to an image with 0.02 default variance value, else if input is 2 then it adds Gaussian noise to an image with default 0.01 variance and 0 mean value. Else if input is 3 then it adds speckle noise to an image with the equation  $J=I+n*I$ . Where, n is uniformly distributed random noise with mean 0 and variance 0.05. Depending on the type of the input it displays the image and writes data of that image to the specified filename. After checking the conditions, colour channels (RGB) from noise added image will be extracted. Once the colour channels are extracted from noise added image, it loads the pretrained denoising network, in which process undergoes using DnCNN layers such as input layer, convolution layer, Rectified linear unit and batch normalization. After performing these operations on input, pre trained convolutional denoising network model applied to each channels (RGB) separately using the command ‘denoiseImage’ to estimate denoised images from noisy image using a denoising deep neural network which is specified by net. These denoised channels are combined by concatenate process. After combining all 3 channels finally it produce a denoised image as an output and displays on the screen using the command ‘imshow ( )’ and finally writes image data to the specified file name.

Peak signal to noise ratio (PSNR) gives the quality of an image when it is reconstructing. As the PSNR ratio increases, the image which is reconstructed is of better quality. The PSNR ratio is calculated by using the formula,

$$PSNR= 10\log_{10} (R^2/MSE)$$

Where, R indicates the maximum fluctuation in the input image data type. For a double-precision floating point data type of an input image, R is 1. If it has 8-bit unsigned integer data type, R is 255, etc.

#### IV. EXPERIMENTAL RESULTS

Fig 4 shows the input image which is exported from the file. This image is converted into double precision (unsigned 8-bit format is converted into double precision) to produce a pristine (original) image which is shown in fig 5. Fig 6 shows the Gaussian noise added image (input 2). From noise added image colour channels are extracted and undergoes to the process of convolution operation, ReLu and Batch normalization process to load the pre trained denoising neural network. After performing these operations, pre trained convolutional denoising network model is applied to each colour (RGB) channel in order to estimate denoised image from noisy image using a denoising deep neural network specified by net. After removing noise from an image it gives denoised image as shown in fig 7.



Fig 4. Input Image



Fig 5. Pristine (Original) Image

**Fig 6. Gaussian Noise Added Image****Fig 7. Denoised (Output) Image**

## V. CONCLUSION

A deep convolutional neural network is proposed for image denoising, where residual learning is adapted to separating noise from noisy observation. The batch normalization and residual learning are integrated to speed up the training process as well as boost the denoising performance. In this paper, we proposed a new CNN model, namely FFDNet. FFDNet is a faster, more effective, more versatile and flexible discriminative denoising. FFDNet takes noise level map as input and denoising in downsampled sub-images space. The result on 2D images with AWGN demonstrated that FFDNet can not only produce state-of-the-art results when input noise level matches ground-truth noise level, but also controls the trade-off between noise reduction and detail preservation. By considering its flexibility, efficiency and effectiveness, FFDNet provides a practical solution to CNN denoising applications. The peak signal to noise ratio (PSNR) of both noise and denoised images is computed, wherein the PSNR value of noisy image is 20.1691dB and PSNR value of denoised image is 30.2263dB. The structural similarity index (SSIM) value of noisy image and denoised image are 0.3398 and 0.829 respectively and correlation between original image and denoised image is 99.1413.

## REFERENCES

- [1] M. Bertalmío, *Image Processing for Cinema*. Boca Raton, FL, USA: CRC Press, 2014.
- [2] H. C. Andrews and B. R. Hunt, "Digital image restoration," *Prentice-Hall Signal Processing Series*, Englewood Cliffs: Prentice-Hall, 1977, vol. 1, 1977.
- [3] P. Chatterjee and P. Milanfar, "Is denoising dead?" *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 895–911, 2010.
- [4] S. Roth and M. J. Black, "Fields of experts: A framework for learning image priors," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2005, pp. 860–867.
- [5] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *IEEE International Conference on Computer Vision*, 2011, pp. 479–486.
- [6] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2862–2869.
- [7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", in *Advances in neural information processing systems*, 2012, pp. 1097-1105.
- [8] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448-456.
- [9] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.
- [10] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2862-2869.
- [11] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" in *IEEE conference on Computer Vision and Pattern Recognition*, 2012, pp. 2392-2399.





**INNO**  **SPACE**  
SJIF Scientific Journal Impact Factor

**Impact Factor:  
7.512**

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
**INDIA**



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 **9940 572 462**  **6381 907 438**  **ijirset@gmail.com**



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details