International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

DCNMAF: Dilated Convolution Neural Network Model with Mixed Activation Functions for Image De-Noising

C. Radhiya Devi[1]*, S. K. Jayanthi[2]

Submitted: 16/11/2022 **Accepted**: 19/02/2023

Abstract: In image processing applications pre-processing the image is the most crucial step. It is essential to eliminate the noise of the image and enhance its quality for further processing. This paper proposes a novel idea to de-noise the image using the Dilated Convolution Neural Network model with Mixed Activation Functions (DCNMAF). Performance Evaluation is done based on the metrics PSNR and SSIM and the proposed model out performs other methods with higher PSNR and SSIM values.

Keywords: *de-noising, deep learning, CNN, Dilation, Activation Function, PSNR, SSIM*

1. Introduction

In image processing, Noise is an undesired information that degrades image quality and should be eliminated. Noise removal is the pre-processing stage for many applications. Noise is a major obstacle for automatic image understanding. Noise reduction is essential because it improves image quality and makes it more suitable for subsequent processing. Image de-noising is a vital task in image processing for image analysis [1]. The de-noising model is a technique used to remove unnecessary data from images; different images are denoised using various methods. In this paper, a new model named Dilated Convolution Neural Network model with Mixed Activation Functions is proposed and used as image pre-processing method to minimize noise in images.

This paper is organized as follows, Section 2 briefly reviews the existing De-Noising techniques. Section 3 describes the methodology along with the comparison of existing methods. Experimental results are discussed in Section 4. Performance Evaluation is done in Section 5 and finally conclusion is presented in Section 6.

2. Literature Review

A number of filtering techniques have been proposed to remove the noise in an image.

Afrah Ramadhan et al. [2] proposed a new method of image de-noising using median filter (MF) in the wavelet domain. The method employs median filter and adaptive wavelet threshold, using different types of wavelet transform filters. It achieves better PSNR values compared to that of applying DWT or median filter only.

B. K. Shreyamsha Kumar [3] proposes a mixture of NLM (Non-Local Means) filter and noise thresholding using wavelets for image denoising. The application of NLM filter on the noisy image removes the noise and cleans the edges without losing too many fine structures and details.

Zhang M et.al., [4] proposed an extension of the bilateral filter named as multi-resolution bilateral filter, where bilateral filtering is applied to low-frequency subbands of a signal decomposed using an orthogonal wavelet transform. The authors proposed this new image denoising framework by combining with specific wavelet thresholding technique.

Zhe Liu et al. [5] have used a CNN model in deep learning for image denoising. Their proposed linear CNN model uses nine convolution kernels and each output color component is composed with the convolutional outputs of all input color components and multiple convolution kernels. The proposed CNN model can effectively remove Gaussian noise and improve the performance of traditional image filtering methods significantly.

Ghose et al. [6] used the pre trained denoising model because CNN model can be optimized continuously and the weights of convolutional kernel can be improved during training of the network. Authors also have analysed the results and compared them with Wiener filtering, Bilateral filtering, PCA, Wavelet based transform method and observed that CNN based denoising method shows best performance than all other compared methods.

[1]*Research Scholar, Department of Computer Science, Vellalar College for Women, Erode, Tamil Nādu, India,*
E-mail:radhiyadevic@gmail.com
[2]*Principal and Head, Department of Computer Science, Vellalar College for Women, Erode, Tamil Nādu, India,*
E-mail:jayanthiskp@gmail.com

From the above research works, it can be inferred that deep Convolutional Neural Network architecture can be applied for the denoising process to enhance the quality of the result.

3. Methodology

Gaussian Filter

A type of linear smoothing filters called Gaussian filter select their weights based on the characteristics of a Gaussian function. The Gaussian filter outputs a weighted average of each pixel's neighborhood, with the weighted average more towards the value of the central pixels.

Average Filter

Average Filter is applied for smoothing images by lowering the intensity values between adjacent pixels. In average filter the value of each pixel is replaced by the average of the intensity levels in the neighborhood defined by the filter mask which produces a weighted sum of a pixel's and its neighbours' values. This process is repeated for every pixel value in the image. The Average Filter has the limitation of blurring the image's object edges. [7]

Existing filtering methods Gaussian Filter and Average Filter denoise the image by concentrating only on neighborhood pixels and it will not preserve the image edge details.

Deep Learning is a subset of Machine Learning research, which has gained popularity in recent past. A unique class of Artificial Neural Networks called Deep Learning models employs a hierarchy of multiple layers to learn multiple levels of representation. The biggest advantage of Deep Learning techniques is that they do not rely on hand-crafted features. Rather, these networks learn features while training without any human intervention. Image de-noising has been accomplished through the use of a deep learning architecture known as Convolutional Neural Networks (CNN).

DnCNN

One existing method using CNN is the denoising CNN (**DnCNN)** proposed by Zhang et al [8]. The authors used the DnCNN for image denoising, super-resolution, and JPEG image blocking. The DnCNN is an efficient deep learning model to estimate a residual image from the input image with the Gaussian noise. The difference between the noisy image and the residue image can be used to predict the noise-free image.

The Proposed DCNMAF Model

As Convolutional Neural Networks (CNN) is one of the most widely used deep neural network models and also has gained popularity for image denoising problems, in this paper CNN model has been used for denoising the image and provide better test results than that of the existing methods.

To complete a specific task using a CNN model, first the corresponding network architecture must be designed, and then the model must be learned using trained data.

A CNN architecture is designed with three layers, namely, the input layer, multiple hidden layers and an output layer. The first layer defines the size and type of the input data. The middle or hidden layers of the network define the core architecture of the network, where most of the computation and learning take place. The final layers define the size and type of output data.

The design of the proposed DCNMAF Model is combined together in a Layer array as shown in Fig 1.

```
layers =

  15×1 Layer array with layers:

     1   ''   Image Input           40×40×1 images with 'zerocenter' normalization
     2   ''   Convolution           32 3×3 convolutions with stride [1  1], dilation factor [2  2], and padding 'same'
     3   ''   Batch Normalization   Batch normalization
     4   ''   ReLU                  ReLU
     5   ''   Convolution           32 3×3 convolutions with stride [1  1], dilation factor [2  2], and padding 'same'
     6   ''   Batch Normalization   Batch normalization
     7   ''   ELU                   ELU with Alpha 1
     8   ''   Convolution           32 3×3 convolutions with stride [1  1], dilation factor [2  2], and padding 'same'
     9   ''   Batch Normalization   Batch normalization
    10   ''   ReLU                  ReLU
    11   ''   Convolution           32 3×3 convolutions with stride [1  1], dilation factor [2  2], and padding 'same'
    12   ''   Batch Normalization   Batch normalization
    13   ''   ELU                   ELU with Alpha 1
    14   ''   Convolution           1 3×3 convolutions with stride [1  1] and padding 'same'
    15   ''   Regression Output     mean-squared-error
```

Fig 1: Layers of the Proposed DCNMAF Model

The input images are 40-by-40-by-1. An image input layer is created of the same size as that of the training images. Hidden layers perform the feature extraction and produce feature maps by the process of the convolution operation. To enhance the non-linearity of the network's modelling capabilities, an activation function has been added to the hidden layers [9]. The Activation Function's role is to generate output from a set of input values fed to a layer. The architecture of the proposed DCNMAF model designed is shown in Fig. 2.

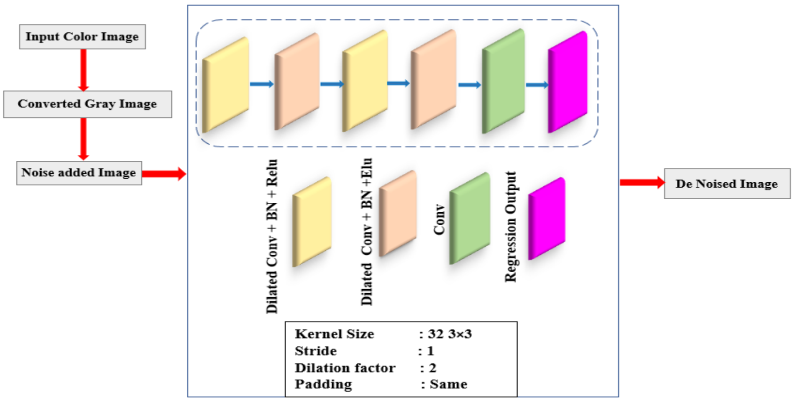


Fig 2: Architecture of the Proposed DCNMAF Model

The proposed DCNMAF model consists of 15 layers including 4 dilated convolution layers. Dilated Conv refers to the dilated convolution with factor of 2 and denoted by 'Dilated Conv'. Dilated convolution increases the receptive field of the layer without adding more parameters or computation by using a dilated filter. The layer enlarges the filters by adding zeros between each filter component. Expanding the CNN's receptive field is a very effective way to extract more features for image de-noising. The last convolutional layer uses regular convolution with dilation factor of 1, hence they are denoted by 'Conv'.

The preference of activation function has a major impact on the neural network's ability and performance, and different activation functions may be used in different parts of the model. The activation function RELU is abbreviated as Rectified Linear Unit. RELU has a derivative function, backpropagation support, and is computationally efficient. The RELU function does not simultaneously activate all neurons. The neurons will only be deactivated if the linear transformation output is less than 0. RELU function is defined by R (z) = max {0, z}. If the input value is positive, the RELU function outputs the input as it is. If the input value is negative, the RELU function outputs the value 0. The graph of RELU is shown in Fig 3 with negative inputs and positive inputs and outputs plotted as per the definition.

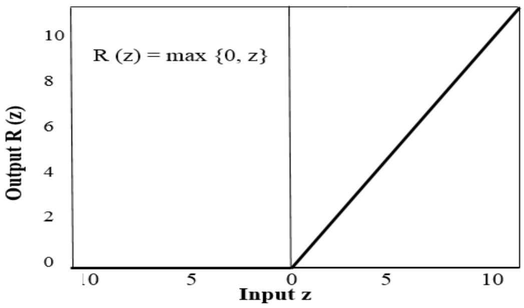


Fig 3: Graph of RELU

The activation function ELU is abbreviated as Exponential Linear Unit. It is a RELU variant that changes the slope of the negative part of the function that has an extra alpha constant defining function smoothness. It performs the identity operation on positive inputs and an exponential nonlinearity on

negative inputs. For positive input values of x, the function simply outputs x. For negative input values, the output is exp(x) – 1. The graph of ELU is shown in Fig 4 representing the outputs for inputs between -4 and 4.

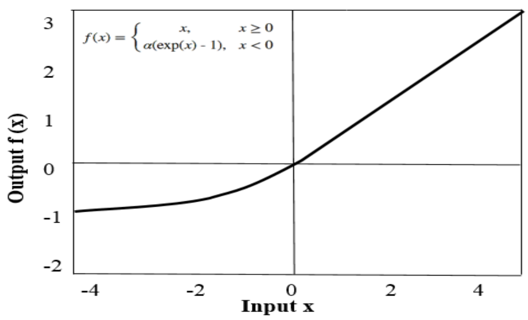


Fig 4: Graph of ELU

Another significant aspect in this proposed network for making the model non-linear is mixing different activation functions such as RELU and ELU for convolutional layers. To improve the problem of gradient disappearance, batch normalization is added between convolutional layer and activation functions. The final output layer is Regression Layer.

4. Experimental Results

The model has been trained after designing the network architecture. The model was trained using Adam optimizer with a mini batch size of 128, i.e., a total of 128 images will be processed at a time. Epochs at two, and with an initial learning rate of 0.0001. The properties of the model were shown in Fig 5.

```
      PatchesPerImage: 512
            PatchSize: [40 40 1]
   GaussianNoiseLevel: [0.0100 0.1000]
        ChannelFormat: 'grayscale'
        MiniBatchSize: 128
      NumObservations: 137216
DispatchInBackground: 0
```

Fig 5: Properties of the proposed DCNMAF model

The training process of the proposed network model is shown in Fig. 6.

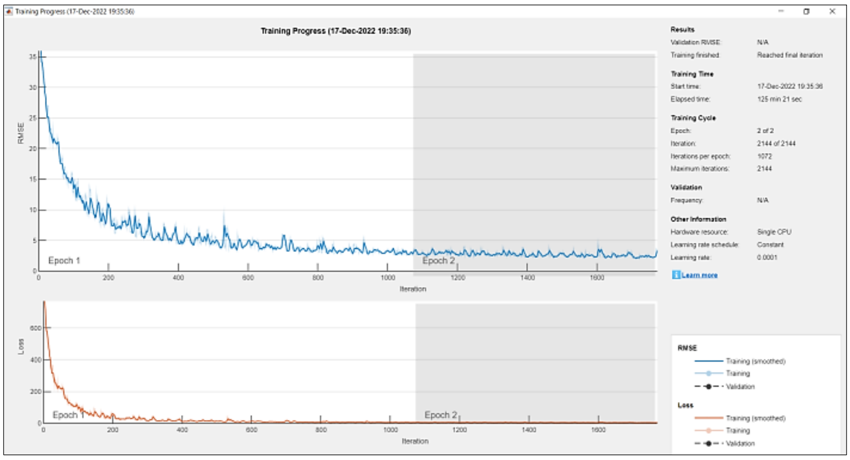


Fig. 6: Plot of the training process of the proposed DCNMAF model

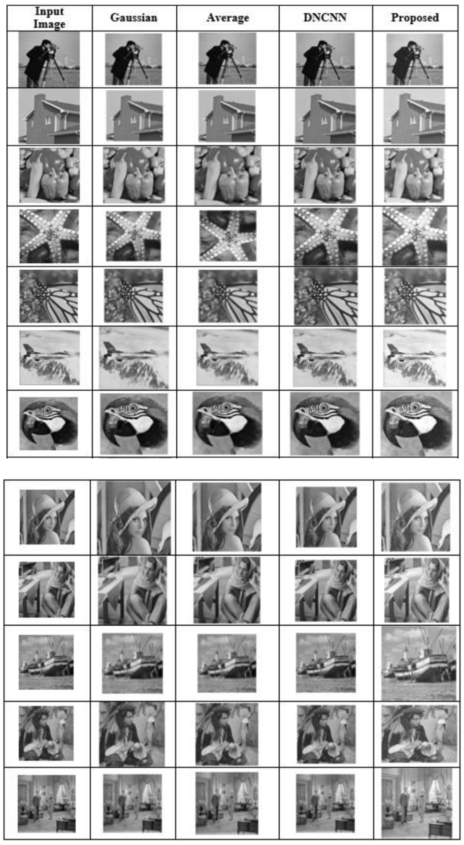
The proposed network and the network taken for comparison were executed on an Intel Core i3 64-bit processor at 2.30GHz with 4GB RAM under Windows

10 and running MATLAB 2021a. Data set used is Set12 dataset and the experimental results of image denoising is done on the 12 test images from the dataset. Fig 7 displays the twelve test images from Set12 dataset.

Fig 7: Twelve test images from Set12 dataset - Cameraman, House, Peppers, Starfish, Monarch, Airplane, Parrot, Lena, Barbara, Boat, Man and Couple

The proposed DCNMAF model reduces noise and enhances image quality. Table I shows the experimental result.

Table I - Experimental Result-Image De-noising using various filters



5. Performance Evaluation

The goal of performance evaluation is to assess the quality of an original image and the filtered image. It is evaluated based on PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index Method) values.

5.1.1. PSNR

The PSNR ratio between two images is the measurement of quality between the original image and the resultant image. The logarithm term of the decibel scale is often used to calculate the PSNR [10]. The higher the value of PSNR, the better will be the quality of the output image. PSNR is expressed as:

$$\boxed{\text{PSNR} = 10log_{10}(peakval^{2)}\ /MSE}$$

5.1.2 SSIM

Structural Similarity Index Method is a perception-based model. It measures the similarity between two images - the original and the recovered. The prediction of image quality depends on distortion free image which is used as reference image to assess quality [10]. The SSIM is represented as,

$$\text{SSIM}=\frac{(2*\bar{x}*\bar{y}+c1)(2*\sigma_{xy}+c2}{(\sigma_x{}^2+\sigma_y{}^2+c2)*((\bar{x}^2+\bar{y}^2+c1)}$$

Table II shows the result comparison of PSNR values and SSIM of Gaussian, Average, DNCNN and the proposed DCNMAF model.

Table II - Comparison of PSNR and SSIM values

Input Image	Gaussian		Average		DNCNN		**DCNMAF**	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Cameraman	22.07	0.61	22.14	0.60	28.98	0.67	**29.10**	**0.88**
House	22.61	0.59	22.58	0.58	29.78	0.68	**31.82**	**0.89**
Peppers	22.37	0.64	22.12	0.62	29.63	0.75	**29.99**	**0.89**
Starfish	22.15	0.69	22.42	0.70	31.44	0.85	**29.91**	**0.91**
Monarch	22.55	0.69	22.43	0.67	21.31	0.54	**28.76**	**0.89**
Airplane	21.75	0.65	22.00	0.64	21.20	0.48	**29.29**	**0.91**
Parrot	22.26	0.65	22.18	0.65	21.43	0.46	**28.98**	**0.89**
Lena	22.33	0.58	22.37	0.56	29.84	0.69	**31.90**	**0.88**
Barbara	22.33	0.65	22.29	0.60	21.09	0.47	**29.26**	**0.89**
Boat	22.49	0.62	22.39	0.59	21.11	0.43	**30.64**	**0.88**
Man	22.47	0.62	22.37	0.59	21.08	0.42	**31.06**	**0.89**
Couple	22.51	0.63	22.53	0.61	21.20.	0.46	**30.37**	**0.88**

From Table II it can be inferred that the proposed DCNMAF method can perform better and give higher PSNR and SSIM values in the range of [28, 31] and [0.88, 0.91], respectively. Averagely achieved 30.09 dB of PSNR and 0.89 of SSIM value. The PSNR and SSIM value of the proposed model gave better results. Chart Comparison of PSNR values and SSIM values are shown in Fig 8 and 9 respectively.

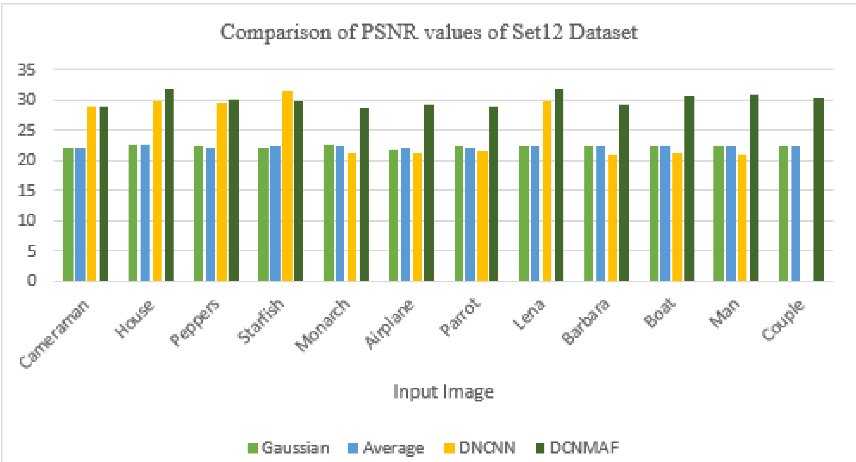


Fig 8: Chart Comparison of PSNR values

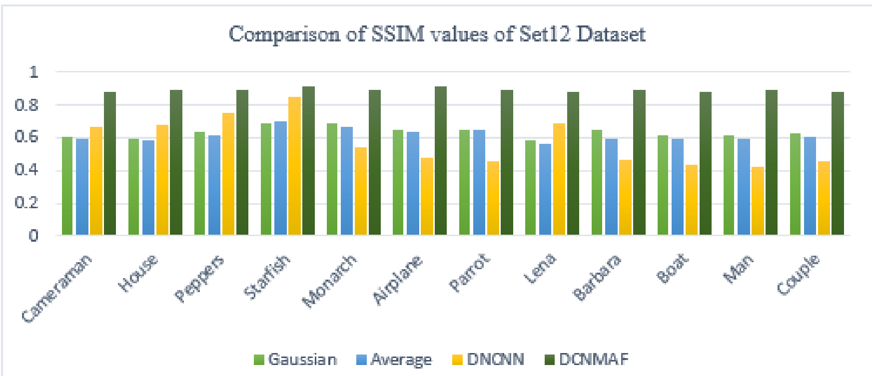


Fig 9: Chart Comparison of SSIM values

6. Conclusion

Pre-processing reduces noise in an image without compromising its information in order to enhance its quality to make it appropriate for future processing. The quantitative analysis is performed using visual quality of the image and using parameters like PSNR and SSIM. The proposed DCNMAF model preserves the edges information and doesn't blur the image. The experimental result shows high PSNR and improved the quality of images than the other filters such as Gaussian, Average filter and DNCNN. As a result, this general method will be utilized for pre-processing images, making it more suitable for subsequent processing.

References

[1]. Mr. Mandar D. Sontakke, Dr. Mrs. Meghana S. Kulkarni (2015), Different types of noises in images and noise removing technique, Inter-national Journal of Advanced Technology in Engineering and Science, Volume No.03, Issue No. 01, ISSN: 2348 – 7550.

[2]. Afrah Ramadhan, Firas Mahmood and Atilla Elci (2017), Image Denoising by Median Filter in Wavelet domain, The International Journal of Multimedia & Its Applications (IJMA) Volume No.9.

[3]. B. K. Shreyamsha Kumar (2013), Image Denoising based on Non Local-means Filter and its Method Noise Thresholding, Signal, Image and Video Processing, Volume No.7, Issue No. 6, pp. 1211-1227, doi: 10.1007/s11760-012-0389-y.

[4]. Zhang M, Gunturk BK (2008), A New Image Denoising Framework Based on Bilateral Filter. Proc SPIE Int Soc Opt Eng, doi: 10.1117/ 12.768101

[5]. Zhe Liu, Wei Qi Yan, and Mee Loong Yang (2018), Image Denoising Based on A CNN Model, 4th International Conference on Control, Auto-mation and Robotics.

[6]. Shreyasi Ghose, Nishi Singh, Prabhishek Singh (2020), Image Denoising using Deep Learning Convolutional Neural Network, IEEE 10th International Conference on Cloud Computing, Data Science & Engineering.

[7]. Rafael Gonzalez Richard Woods, Digital Image Processing, Third Edition, 2008, Pearson Publica-tions

[8]. K. Zhang, W. Zuo, Y. Chen, D. Meng and L. Zhang (2017), Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising, IEEE Trans. Image Process., Volume No. 26, no. 2, pp. 3142-3155.

[9]. Deep Learning: A Practitioner's Approach, Josh Patterson & Adam Gibson, 2017, O'Reilly Publications

[10]. Umme Sara (2019), Comparative Study of Different Quality Assessment Techniques on Color Images, IRE Journals, Volume No. 2, Issue No. 11, ISSN: 2456-8880