Deep CNN with Residual learning and Dilated Convolution for Image Denoising

Karthikeyan B
School of Computing,
SASTRA Deemed University, Thanjavur, India

Mokkala Mounika
School of Computing,
SASTRA Deemed University, Thanjavur, India

Sirigireddy Buchireddy Pravallika
School of Computing,
SASTRA Deemed University, Thanjavur, India
Email: pravallika0123456789@gmail.com

Karakala Mahitha
School of Computing,
SASTRA Deemed University, Thanjavur, India

***Abstract*— In the field of digital image processing, filtering noise from the image to extract a high-quality image is an important pre-processing step for object detection, segmentation, tracking, etc. Convolutional Neural Network (CNN) has gained great attention in the domain of Image Denoising, with its flexible architecture. There are some challenges in using the deep CNNs for image denoising tasks like training the model and performance saturation when the depth of the network is increased. In the proposed model, two networks are connected parallelly to increase the width of the network rather than depth, and hence obtain more context information and make it less prone to performance saturation. Batch Normalization improves the performance of the network and mitigates the internal covariate shift problem. Dilated convolutions are used in the proposed model to extract more features by enlarging the receptive field. Residual learning is adopted to overcome exploding gradient and vanishing gradient problems. Experimental results have shown that our proposed model is more efficient than many existing filters.**

Keywords—Deep learning, Residual Learning, Batch Normalization, Dilated Convolution, Convolutional Neural Networks, Tensorflow, Keras

I. INTRODUCTION

Computer vision is a field in Artificial Intelligence, where computers are trained to interpret the visual word. Digital images play a huge role in many aspects of our day-to-day life, as they are used in intelligent traffic monitoring, remote sensing, signature approval, satellite TV, recognition of handwriting on checks, in the field of geographical information systems, etc. Due to the impact of transmission channels, also the other factors, images are being corrupted by noise during the process of compression, transmission, etc., which ultimately results in loss of data in the image. Removing noise from an image to restore a high-quality image, has become an important task for further processing of an image in detection, tracking, object segmentation, etc. [1]. If the image isn't effectively filtered, then it would directly affect the subsequent process of feature extraction, detection, etc. Hence, Image-denoising is a classical-inverse problem in the field of computer vision. It aims to eliminate the noise from the original image as much as possible and tries to restore the image details. We have two forms of filtering for 2D visual signals, which are based on the frequency and spatial domains. Pixel intensity is the feature that matters in the image filtering in the case of the spatial domain. But in the case of the frequency domain, after visual decomposition, coefficients of multiple frequencies are used [1]. Hence, the image denoising methods based on the spatial domain would deal with the intensity of each pixel in an image, while the denoising methods based on the frequency domain, after image decomposition, focus on adjusting the coefficients of multiple frequencies.

In digital image processing, for pattern classification, Artificial Neural Networks were the preferred classifiers. Examples would be image restoration, facial recognition, image enhancement, etc. Before proceeding to this paper, the merits of neural networks have been thoroughly investigated. Hence, neural networks (in specific Convolutional Neural Networks CNN) are used in this paper for image denoising. In this paper, majorly four methodologies have been involved. Firstly, the use of deep CNN with increased width increases the networks' learning ability. Secondly, the dilated convolutions are used to enlarge the receptive field, which would reduce the computational cost along with the increased extraction of more contextual information. Thirdly, the Residual Learning concept would enhance the image-denoising performance. Lastly, with the Batch Normalization concept, inputs would be standardized for each batch. The test outcomes reveal that our model gives fine results for both real and synthetic noisy images.

II. LITERATURE SURVEY

A. Traditional Denoising Approaches

There are some popular denoising methods named Markov Random Field (MRF) [8], total-variation (TV) methods [12],[13], Average filtering (2007), Wiener filtering (2007), gradient methods (2009), Median filtering (2011), Non-locally Centralized Sparse Representation (NCSR) (2013) and Weighted Nuclear Norm Minimization (WNNM) (2014). Average filtering is a typical linear filter [2], which is combined with wavelet transform to obtain good denoised results. The Wiener filtering method is especially used for motion blur removal. In this algorithm, the images with Gaussian noise have their directional coefficients subjected to normal distribution [3]. Gradient descent is a process that occurs in the backpropagation phase where the continuous resampling of the gradient of the models' parameter is done. Median filtering performs blurring operations for image denoising, which is based on the traditional median filtering algorithm [4]. NCSR centralized the sparse coding to overcome the sparse-coding noise [5]. Each singular value is uniformly regulated to pursue the convexity of the objective function and it is called standard nuclear norm minimization (SNNM). Whereas, in WNNM different weights are assigned to singular values [6]. These are the popularly known denoising methods earlier.

Though the mentioned methods have shown very good performance, they have suffered from two major problems [7]. 1) To improve the performance, they need complex optimization methods. 2) They have to manually tune the parameters to get the optimal results.

B. *Neural-networks based image-denoising*

The deep network architecture has flexible connection fashion and strong learning ability. Hence, the addressed problems of traditional denoising methods have been solved by deep learning techniques. In specific, Convolutional Neural Networks (CNNs) are used in this paper for image denoising. A CNN model can continuously optimize the weights of the kernel during the training phase of the network, and it is its major advantage.

Deep CNNs (CNNs with increased depth) are widely used nowadays in image-denoising methods. Residual and iterative ideas were combined to a CNN and used for image denoising [14]. CNN maps a low-resolution image directly into a high-resolution image [15]. Although the above methods were able to give good results, they faced two challenges. 1) Since, the networks are deep, it's difficult to train them. 2) The problem of vanishing and exploding gradients. Also, some of the methods mentioned above have high computational costs.

C. *Dilated Convolution*

Pooling operations are used in the traditional CNNs for reducing the dimensionality of the image data, which results in information loss. The receptive field enlargement would be the solution for this information loss. The receptive field can be enlarged in two ways [9]. Either by increasing depth or by increasing filter size. Increasing the depth would result in the degradation of network performance. While increasing filter size increases the parameters of the network and results in a high computational cost.

The convolution applied to the input data with defined gaps based on dilation rate k is called dilated convolution. For an image, if the dilation rate k=1, then it is a normal convolution. If the dilation rate k=2, then one pixel is skipped per input. If the dilation rate k=4, then 3 pixels will be skipped per input. It solves the problem of information loss after the pooling operation.

D. *Batch Normalization*

Since deep CNN has more layers, it is difficult to train it. The layers can be affected by the initial weights taken randomly and by the configuration of the algorithm. The input distribution deep in the layers of the network keeps changing after each batch whenever the weights are updated. The target for the learning algorithm keeps moving. This kind of change in the inputs present in the layers of the network is called an "internal covariate shift" [11]. The Batch Normalization technique standardizes the inputs to a layer for every batch. As it reduces the number of training epochs required for training, it is used in training deep CNNs.

E. *Residual Learning*

As we already discussed, the increased depth in the network would result in performance degradation. Hence, Residual Learning was proposed (2016) to have a trade-off between increase of depth and performance degradation. It solves the problem of vanishing and exploding gradients, with its capability to combine the input of many stack layers into the input of the current layer [10]. The Global residual learning technique was exploited for very deep super-resolution (2016) for image restoration. The Overfitting problem in image restoration can be addressed by a combination of recursive mechanism and global residual learning, through which a deeply recursive convolutional network was designed.

III. PROPOSED MODEL

In this section, a novel CNN-based network architecture is designed for image-denoising. The proposed network architecture diagram is shown in Fig 1. Unlike other networks, where the focus is on the depth of the network for optimal results, here the focus is on the width of the network, which enhances the performance of the network. The proposed network has two different sub-networks, an upper and a lower sub- network. The initial noisy image data is sent to both the networks, whose output results will be concatenated to obtain the optimal denoised image.

The upper sub-network has BN (Batch Normalization), Residual Learning, and Dilated convolution. There are in total 10 layers in the upper network. The first and ninth layers are convolution layers with Batch Normalization to normalize data and an activation function of Rectified Linear Unit (ReLU). The last layer is just a convolution layer. The layers from 2 to 8 are dilated convolutional layers. These layers have

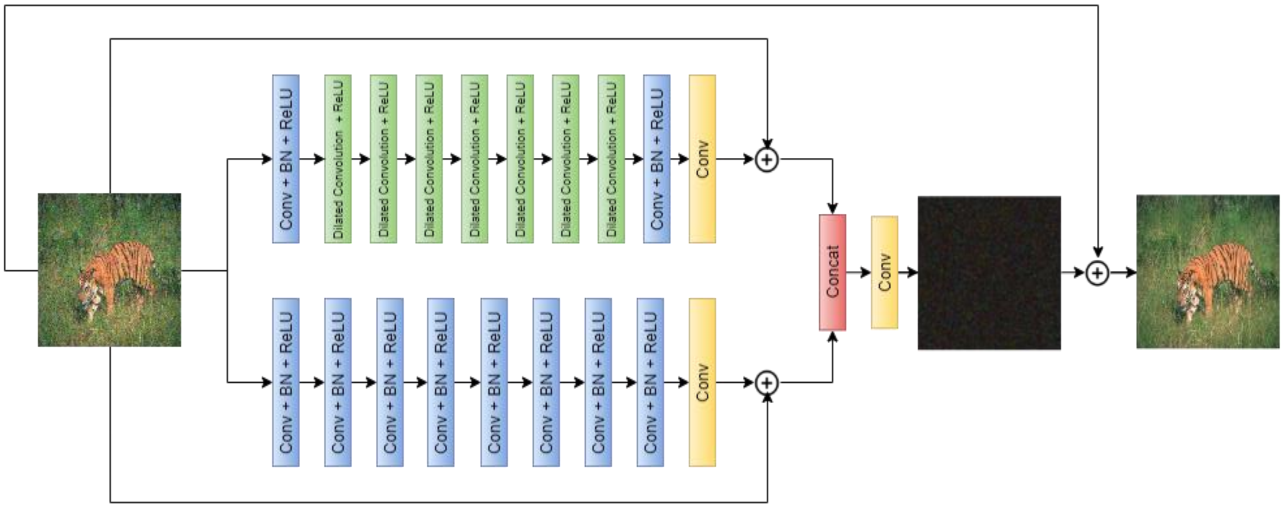


Fig 1. Proposed network architecture

a dilation factor, with the help of which, the receptive field can get more context information. If the dilation factor is 1, then the receptive field size would be (2l+1) X (2l+1), where 'l' be the number of layers. In the proposed network, the dilation factor is 2. The layers from 2 to 8, will receive more context information as their receptive field size will be increased after dilation. The receptive field of these layers is (4l+1) X (4l+1) (since the dilation factor is 2). Hence, the receptive field sizes of all the layers in the upper network are 3, (7, 11, 15, 19, 23, 27, 31), 33, and 35. This is comparable to the performance of a network having 18 layers under the same filter-size settings. Thus, having two sub-networks and dilated convolutions reduce the depth, as it can give similar results of a network having 18 layers, with just 10 layers. As there are only 10 layers in this network, the performance saturation (vanishing or exploding gradients) problem is resolved.

In the lower sub-network, there are 9 layers in total. The first 8 layers have Batch Normalization and ReLU concepts integrated into each. The last layer is just a normal convolution layer. In both the sub-networks, the data obtained at the last layer is subtracted from the noisy image data, which is the concept of Residual Learning. Then, the data from both the sub-networks after subtracting is concatenated and then passed to another convolutional layer to get a residual image. The residual image is finally subtracted from the noisy image data (Residual learning concept), to obtain the denoised image.

In the proposed model, the mean square error is the chosen loss function to attain the optimality in parameters of the network design. Let 'y' be a noisy image, 'x' be a clean image, and the training dataset $\{x_j, y_j\}$, where j=1, 2,…,N is given. The proposed model predicts a residual image f(y). The mapping is done in such a way that deducting f(y) from 'y' should result in a clean image 'x', i.e., x = y – f(y). The loss function is then minimized with the Adam function to get the optimal parameters.

IV. Experimental Results

Three aspects namely datasets, experimental setting, and performance evaluation are considered for showing the experimental results. The performance evaluation of the proposed model is done by comparing and analyzing the proposed model with four filters namely Gaussian filter, Mean filter, Median filter, and Bilateral filter (commonly used filters for denoising images). To compare and analyze the performance of the proposed model with the four available filters, measuring tools like Peak Signal-to-Noise Ratio (PSNR), and Mean Square Error (MSE) are used. If the PSNR value obtained for a denoising method is higher, it indicates that the method performs well. If the MSE value obtained is lower, then the method is said to have good performance.

A. Datasets

1. Training datasets

We used a dataset containing 5,544 images from Kaggle to train the proposed model for Gaussian image denoising. And, the images are rescaled to 180X180 dimensions.

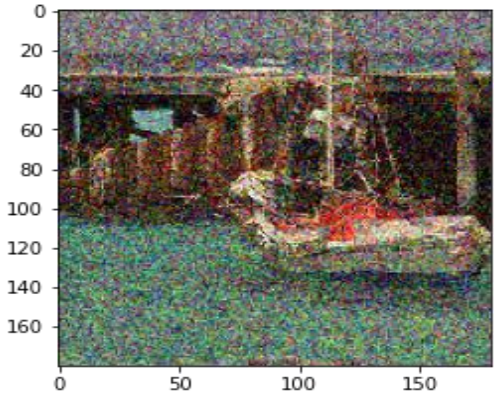
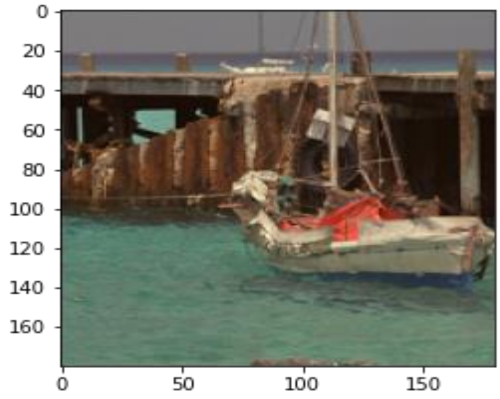
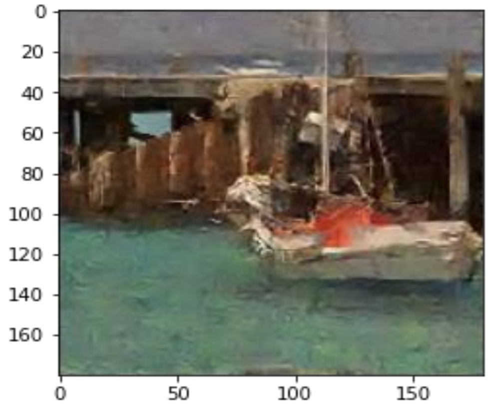
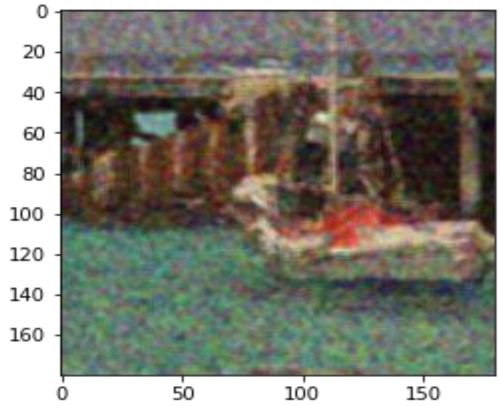
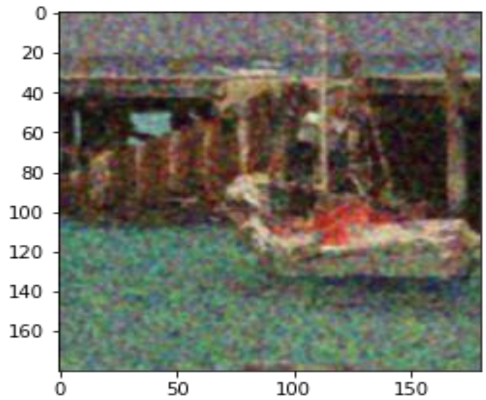
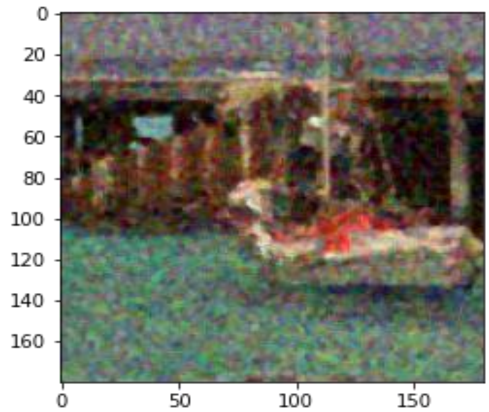
2. Test datasets

To test the proposed trained model, the test dataset consisting of three folders namely CBSD68, Kodale24, and McMaster, contains 68, 24, and 18 color images respectively. Each image inside them is converted from .bmp format to .png format and rescaled to 180X180.

B. Experimental setting

The proposed model has two sub-networks. The upper sub-network has a depth of 10 layers and the lower sub-network has a depth of 9 layers. Rectified Linear Unit (ReLU) is the activation function used. The batch size is 20. The number of epochs is 50. The learning rate of the model varies from 1e-3 to 1e-4 for 50 epochs. We have applied the Tensorflow with Keras package (Python language) to train the proposed model.

Table 1. Denoising results of the proposed model and the four filters, for test image-1

Noisy image 	Original image 
Denoised image (Proposed model)  PSNR: 26.181 MSE: 156.639	Mean filter (denoised image)  PSNR: 22.722 MSE: 347.402
Gaussian filter (denoised image)  PSNR: 22.661 MSE: 352.293	Median filter (denoised image)  PSNR: 21.348 MSE: 476.667

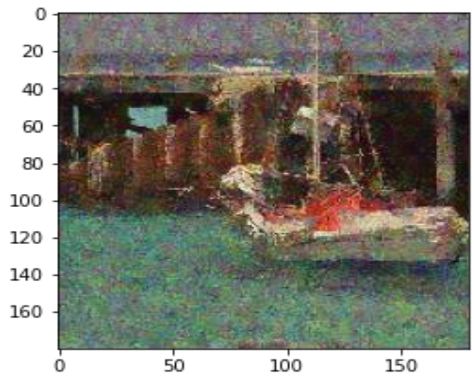
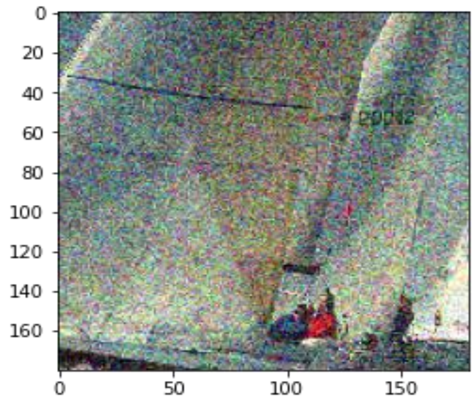
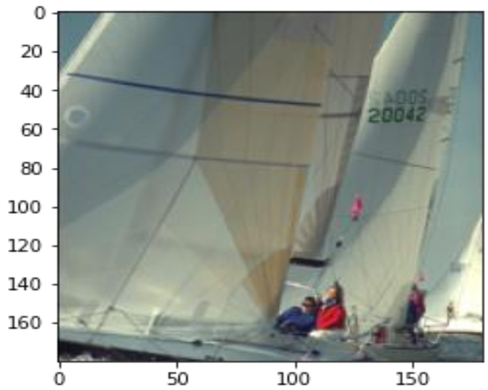
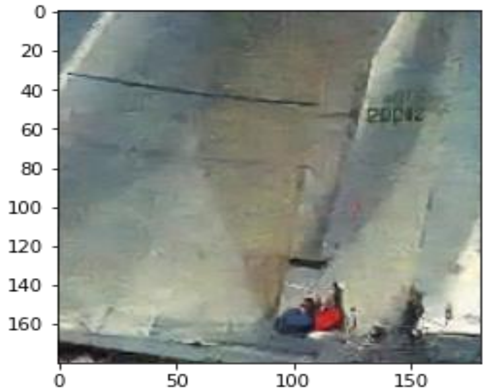
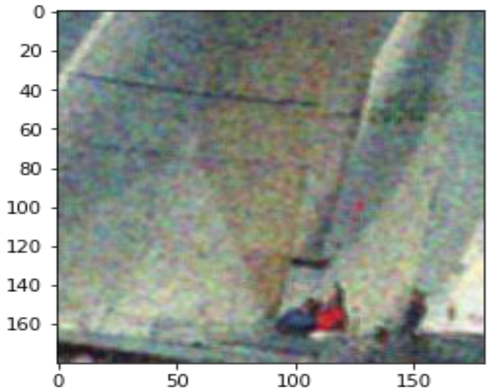
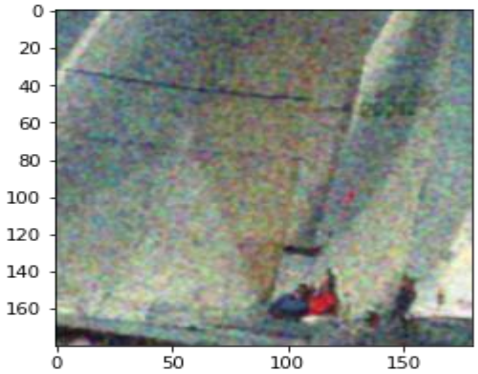
Bilateral filter (denoised image)  PSNR: 19.890198042428175 MSE: 666.8997530864198	The obtained PSNR for the proposed model is 26.181 and the MSE value is 156.639. The PSNR value is higher than the four available filter. Also, the MSE is lower than the four filters.

Table 2. Denoising results of the proposed model and the four filters, for test image-2

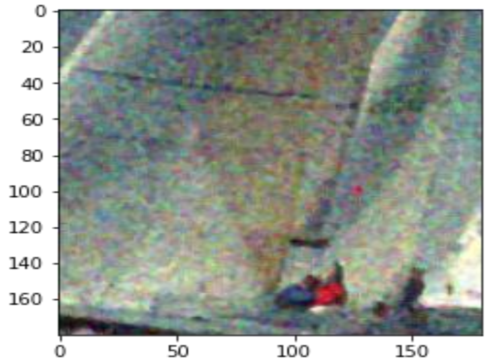
Noisy image 	Original image 
Denoised image (Proposed model)  PSNR: 28.407 MSE: 93.836	Mean filter (denoised image)  PSNR: 24.043 MSE: 256.270

Gaussian filter (denoised image)



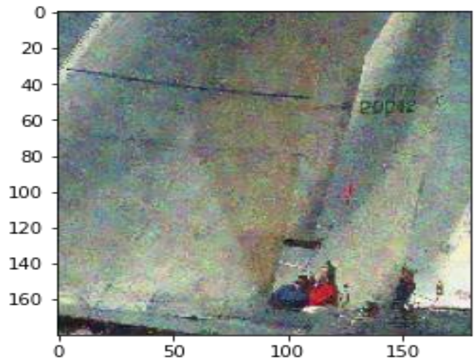
PSNR: 23.734
MSE: 275.207

Median filter (denoised image)



PSNR: 22.587
MSE: 358.393

Bilateral filter (denoised image)



PSNR: 21.561
MSE: 453.914

The obtained PSNR for the proposed model is 28.407 and the MSE value is 93.836.
The PSNR value is higher than the four available filters.
Also, the MSE is lower than the four filters.

Table 3. PSNR values of the denoised images using the proposed model and four other filters, for three sets of test images.

Test dataset Vs Model	Gaussian Filter	Mean Filter	Median Filter	Bilateral Filter	Proposed Model
CBSD68 (68 images)	25.47	24.49	23.99	25.28	27.84
Kodale24 (24 images)	26.03	25.14	24.34	25.50	28.05
McMaster (18 images)	25.74	24.69	24.32	25.45	27.62

Table 4. MSE values of the denoised images using the proposed model and four other filters, for three sets of test images.

Test dataset Vs Model	Gaussian Filter	Mean Filter	Median Filter	Bilateral Filter	Proposed Model
CBSD68 (68 images)	423.73	463.51	521.56	488.97	359.38
Kodale24 (24 images)	196.01	228.18	277.90	254.27	117.96
McMaster (18 images)	207.28	252.44	276.63	252.93	127.86

C. *Performance Evaluation*

The performance and efficiency of the proposed model are being compared with four filters named Gaussian filter, Mean filter, Median filter, and Bilateral filter. The PSNR and MSE values of the proposed model and four filters are compared. The quality of the image increases with the increase in PSNR value. Also, the lower the MSE value the higher the efficiency of the model. Two test images are taken and the denoising results of the test images using the proposed model and the four filters are shown in table 1 and table 2. Table 3 shows the average PSNR values of the denoised images for the test set having three subfolders namely CBSD68, Kodale24, and McMaster each containing 68, 24, and 18 images respectively using the proposed model and four other filters. Similarly, table 4 shows the average MSE values of the denoised images for the test set using the proposed model and four other filters.

CONCLUSION

In this paper, a novel CNN-based model is designed. In this network architecture, two subnetworks are present, to increase the width of the network. This increases the image-denoising performance. The concepts of Residual Learning and Dilated Convolution are integrated into the proposed model. Dilated convolutional layer helps to increase the receptive field. Residual Learning is implemented to isolate noise from the noisy images and to get the latent clean image. Experimental results have shown that the proposed model performs better than many available filters like Gaussian filter, Mean Filter, Median filter, and Bilateral filter.

REFERENCES

[1] Liu, Z., Yan, W. Q., and Yang, M. L., "Image denoising based on a CNN model," presented at 4th International Conference on Control, Automation and Robotics 'pp. 389-393, 2018.

[2] S. He, X. L. Pan, and Y. M. Li, "Optimization algorithm for average filtering," Information Technology, vol. 3, p. 41, 2012.

[3] Z. J. Wang, C. W. Qv, and L. Cui, "Images denoising with wiener filter in directionalet domain," Electronics Optics & Control, vol. 6, p. 8, 2007.

[4] Y. Ming and S. Li-hua, "The application of an improved fast algorithm of median filter on removing image noise," Engineering of Surveying and Mapping, vol. 20, no. 3, pp. 65–69, 2011

[5] Dong, W., Zhang, L., Shi, G., et al.: "Nonlocally centralized sparse representation for image restoration," Trans. Image Process 'pp. 1620–1630, 2013.

[6] Gu, S., Zhang, L., Zuo, W., et al.: "Weighted nuclear norm minimization with application to image denoising," Computer Vision Pattern Recognition, pp. 2862–2869, 2014.

[7] Zhang, K., Zuo, W., Chen, Y., et al.: "Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising", Trans. Image Process 'pp.3142–3155, 2017.

[8] Barbu, A., "Learning real-time MRF inference for image denoising," presented at IEEE conference on computer vision and pattern recognition 'pp. 1574-1581, 2009.

[9] Wang, T., Sun, M., Hu, K., "Dilated residual network for image denoising," arXiv preprint arXiv:1708.05473, 2017.

[10] He, K., Zhang, X., Ren, S., et al.: "Deep residual learning for image recognition," presented at IEEE Int. Conf. Computer Vision 'pp. 770–778, 2016.

[11] Ioffe, S., Szegedy, C., "Batch normalization: accelerating deep network training by reducing internal covariate shift," arXiv preprint arXiv:1502.03167, 2015.

[12] Chan, T.F., Chen, K, "An optimization-based multilevel algorithm for total variation image denoising," Multiscale. Model. Simul. 'pp. 615–645, 2006.

[13] Frohn, C., Henn, S., Witsch, K., "Nonlinear multigrid methods for total variation image denoising," Comput. Vis. Sci., 'pp. 199–206, 2004.

[14] Tai, Y., Yang, J., & Liu, X., "Image super-resolution via deep recursive residual network," In Proceedings of the IEEE conference on computer vision and pattern recognition: 1, (2), (p. 5), 2017.

[15] Dong, C., Loy, C. C., He, K., and Tang, X., "Image super-resolution using deep convolutional networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(2), 'pp. 295-307, 2016.