FFDNet: Image Denoising Based on Convolutional Neural Network (CNN)

Dr. Murugaanandam S[1], Dr.Fancy C[2]
[1]Associate Professor, Department of Networking and Communications,
SRM Institute of Science & Technology, Kattankulathur, Chennai, India.
[2]Assistant Professor, Department of Networking and Communications,
SRM Institute of Science and Technology, Chennai, India.

Article Info

Article history:

Received Jan 9, 2025
Revised Feb 20, 2025
Accepted Mar 11, 2025

Keywords:

Convolutional neural network
FFDNet
Image denoising
Peak signal-to-noise ratio
Noise level

ABSTRACT

Image denoising is a technique that aims to reduce or eliminate unwanted sound from images, improving the worth and clearness of the image. During capture and distribution, images frequently have several types of noise, including Gaussian, speckle, and salt-and-pepper noise. Generative adversarial networks (GAN) are identified by high processing costs, improper fitting, and the capacity to introduce mistakes in the training data. These algorithms lose flexibility since their parameters are set and can't be changed while filtering. Based on this, we introduce FFDNet, a quick and adaptable denoising convolutional neural network that uses a stimulus with a programmable degree of sound conjunction. Convolutional neural networks (CNNs), a kind of deep neural network with remarkable item classification accuracy, may be constructed using large databases. The more recent image-denoising method, FFDNet, is based on convolutional neural network architecture. CNN is utilized for spatially varied distortion because it is quick, adaptable, and can efficiently manage various sound stages (i.e., [0, 75]) with just one system. According to the results of experiments, the FFDNet depends on the peak signal-to-noise ratio (PSNR), which is 38.08 dB, 37.63 dB, 37.42 dB, and 36.49 dB, respectively. Based on the result, it is effective and efficient.

Corresponding Author:

Dr. Murugaanandam S,
Associate Professor, Department of Networking and Communications,
SRM Institute of Science & Technology, Kattankulathur, Chennai.
Email: murugaas@srmist.edu.in

1. INTRODUCTION

The Convolutional Neural Network (CNN) is a method for deep learning that employs layers of convolution and pools to extract information and categorize or recognize objects or scenes. It is mainly used for image and video analysis. During the recording, compression, and distribution processes, noise affects images. Noise can affect pictures via pathways, including transmission and environmental sources. One essential component that improves comprehension of processing image tasks is visual denoising [1]. Healthcare imaging, satellite observation, military monitoring, biometrics and criminal justice, industrial and agricultural automation, and person identification are among the fields that employ image-denoising techniques. Denoising methods are essential preliminary processes for healthcare and Eliminating healthcare noise with physiological imaging, including quantum, Rician, and speckle.

Denoising techniques are applied in satellite observation to eliminate additive white Gaussian noise and pepper and salt. Noise may be regarded as a signal distortion making it harder to observe images and recover data. A picture is a fascinating type of data consisting of several elements, such as borders, shapes, flat homogeneous areas, and spikes, all caused by progressively changing grey stage levels of pixel strength

[2]. In different kinds of applications, including automated vision and object identification, noise significantly affects image performance as it can lead to inaccurate divisions and false detections. The noise reduction techniques now in use have difficulty identifying every compromised pixel and target neutral picture noise. Device noise is generated by several factors, including malfunctioning pixels during image capture, and digital picture noise is caused by several factors, including pixels during image collection, device storage failures, and data transfer via noisy channels [3]. Among the numerous established methods for noisy reduction images are non-local processes (NLM) controls, wavelets, diffusion, overall variation, block-matching and three-dimensional in-form sorting (BM3D), limited participation, Markov processes chains, unplanned farm designs, neural network models, and more. The first kinds of filters used for pictures were non-adaptive, simple, and non-linear filters [4]. The filter is a tool that modifies the signal waveform in a manner. Filters are used in digital signal processing to improve signal quality by removing distortion and separating the necessary details from the signal. To minimize the impact of noise, a wide variety of filters are used [5].

Noise elimination filters may be divided into six types: wavelet-based, flexible, straight, irregular, partial differential equation (PDE), and total difference filters. Although a linear picture recovery method can provide a result directly from the observed data, it requires a reverse operator. Non-linear approaches do not directly implement the opposite direction. It employs an Iteration technique to gradually enhance the replacement until a termination condition is satisfied [6].

To minimize noise, linear filters link the necessary input pictures with appropriate exit pixels using the multiplication of matrix technique [7]. Because it loses edge data, linear processing is a bad filtering technique. Adaptive filters employ scientific properties (such as recursive mean square and lowest mean square) for real-time applications.

In particular, the formula for our FFDNet is a = F (b, N; θ), where N is a map of noise levels. While the DnCNN model's variables $\theta$are independent of noise level, the variance data map is a data source for the FFDNet model, whereas the DnCNN model $a = F(b, N; \Theta)$ has variables θthat vary as the signal level $\boldsymbol{\sigma}$ changes. As a result, FFDNet offers a versatile method for managing different kinds of noise with a single network.

FFDNet, a quick and adaptable denoising system, is suggested for selective picture denoising. A single FFDNet can handle noise on multiple levels and spatially variable noise by using an adjustable noise level map as input. Conventional methods require high computational requirements a large amount of labeled data and a large memory footprint we provide a quick and adaptable noise reduction convolutional neural network (FFDNet) to address the shortcomings of CNN techniques. FFDNet has a smaller memory footprint and a faster execution time.

FFDnet's primary benefits are its speed and adaptability as a denoising artificial neural network. FFDNet can handle dynamically varying noise and noise of different intensities. Noise-level maps are offered to preserve the relationship between detail protection and noise reduction. When applied to both artificial and real-world noise, FFDNet shows potentially attractive outcomes.

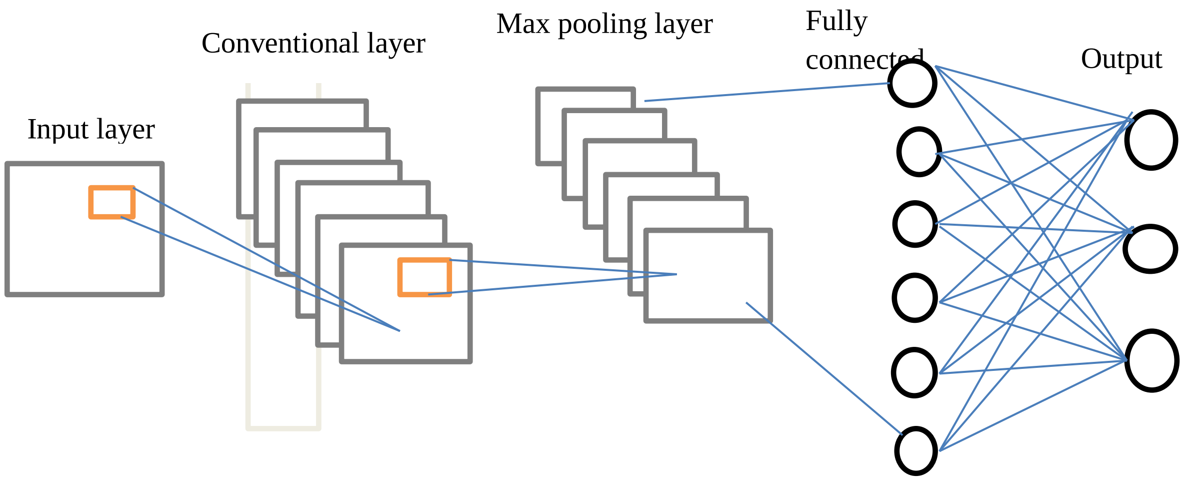

Figure 1. Basic Function of Conventional Neural Network

2. LITERATURE REVIEW

Ziyuan Wang et al [8] introduce a generative adversarial (GAN) network for picture noise reduction-based deep residual network that was suggested to finish the image-denoising task. In addition to being convincing for image denoising, the proposed approach network has remarkable and competitive outcomes for other image processing tasks including image defogging and medical CT denoising.

Yinan Chen et al [9] proposed an excellent self-supervised denoising of pictures technique built on CHRNet and SDDW-GAN. The primary application of SDDW-GAN is sound distribution on noisy input pictures. CHRNet was created to ascertain the linear relationship between the double-noise and single-noise data. The suggested technique increases a typical peak signal-to-noise ratio (PSNR) on two testing datasets with varying noise levels by 0.23 to 0.78 dB. Also, it includes super-resolution, deblurring, and low-light picture improvement.

Jiongkai et al [10] proposed object recognition in images of poor quality using reinforcement learning flexible enhancement. Deep reinforcement learning, or DRL, is recommended to improve the model's identification capabilities for poor-quality pictures. An image-improving utility chain (IETC) has been developed to offer a flexible and compact method selection for the Dueling Deep Q-Network-based Tool Selector (DDQN-TS). Through picture enhancement, the suggested technique turns low-quality photos into high-quality ones.

Luo, W et al [11] introduce a denoising regional heterodyne spectroscopy using deep convolutional neural networks. Convolutional neural network models for spatial heterodyne interferogram denoising in various brightness and Gaussian noise scenarios, and evaluates how well they perform in appraisal to other techniques. The proposed study provides new ideas and useful methods for enhancing signal identification quality and lessening the impact of noise on temporal heterodyne spectrum information.

Luo, Wei, et al [12] proposed the exploration of deep convolutional neural networks for denoising temporal heterodyne spectroscopy. The CFM is adaptable to different denoising systems and attains a superior trade-off between sound decrease and information retention. Because it works better than the state-of-the-art methods, the suggested CFMNet is quite attractive for image denoising.

3. METHODOLOGY

FFDNet eliminates Gaussian noise with spatial variations, typical in noisy real-world photographs. Quick and adaptable CNN is applied to noise that varies spatially. FFDNet uses the same network to handle a variety of noise levels ([0, 75]). The spatially variable noise is removed by specifying an irregular noise level map. It Outperforms reference B3MD in terms of speed not sacrificing discriminative learning Performance noise. The flexibility of CNN-based techniques is constrained; the well-educated model is often adjusted to a particular sound level [13]**.** One model CNN Gaussian denoising is taught on DnCNN-B within the signal value range of [0, 55]. Furthermore, DnCNN-B is not adaptable enough to handle spatially variable noise. FFDNet overcomes the drawbacks of the current CNN-based noise reduction techniques.

FFDNets Construction

Fig. 2 shows the FFDNet architecture. A noise level map and four segments created from the rearranged input picture are fed into the CNN. The initial layer samples the smaller images and aggregates them using a configurable noise level map M. Using a reversible down-sampling process, the input image of size S×R×T is transformed into four down-sampled sub-images of size (S⁄(2) × (R⁄(2))) ×4C. The number of layers in this instance is denoted by c, which is equal to 1 for a grayscale picture and 3 for a color image. The CNN's source is the tensor y of size, which is $(S/2) \times (R/2) \times 4T + 1$ [13]. Each layer is composed of three types of operations: Convolution (Conv), Rectified Linear Units (ReLU), and Batch Normalization (BN). More specifically, "Conv+ReLU" is adopted for the first convolution layer, "Conv+BN+ReLU" for the middle layers, and "Conv" for the last convolution layer. For each convolution, zero-padding is used to maintain the characteristic maps' size [14]. The last convolution layer is followed by the opposite operator of the sampling lower procedure used in the initial input process. The final output is reconstructed from the four denoised sub-images. FFDNet Can be modeled as

$$S^{\wedge} = F(b, N; \Theta) \qquad (1)$$

N is a noise level map, b is a noisy matrix, and the parameters θ are unchanged at different noise levels. FFDNet offers a simple method for controlling varying sound levels inside a particular network.

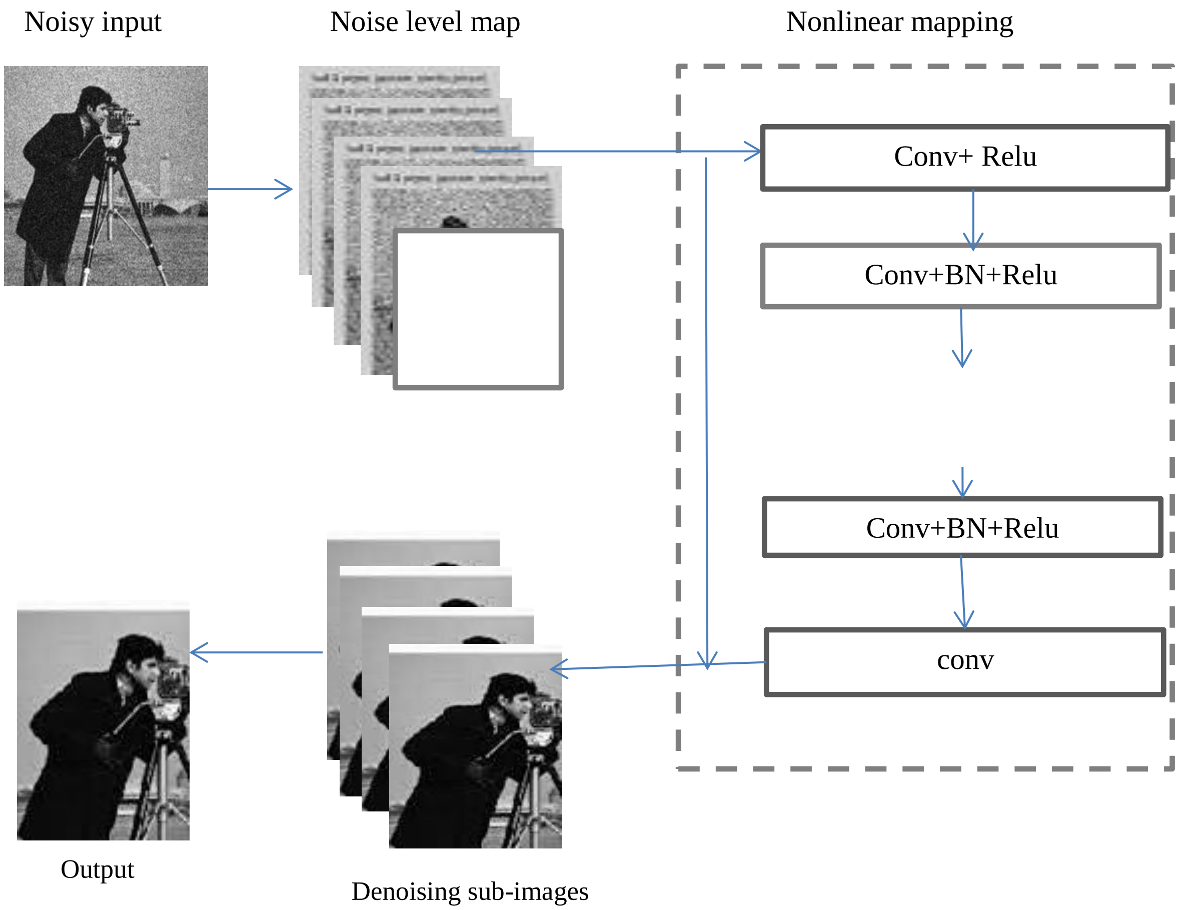

Figure 2. Architecture of FFDNet for image denoising

Additionally, the FFDNet network extends the amount of noise to connect with precisely the dimensions as the input picture**.** The noise level map provides more flexibility to the noise control network and kinds its range of noise level observation. After that, more samples are used to combine all of the denoised low-resolution sub-pictures into high-quality denoised images. Convolution (Conv), BN, and ReLU activating functions are the three operations used in the nonlinear routing of the FFDNet network. Two different networks are added by the FFDNet network model: sub-pixel convolution and picture downsampling, which are shown by the green and red dashed boxes, respectively [15]. Using two as the downsampling factor, an initial noisy picture, which has dimensions of S × R × T, is transformed by downsampling into sub-images of the same size, R× T, and (S/2) × (R/2) × 4T. We experimentally determined that there should be 17 convolution layers for grayscale images and 14 for shade pictures, considering the compromise between efficiency and complexity.

Regarding the map of paths, we set 98 for color images and 66 for grayscale images. We employ distinct settings for color and grayscale photos for two reasons. First, since the R, G, and B channels have four important connections, using fewer convolution layers encourages the approach to establish the inter-channel dependency. Second, as color images require multiple channels for input, more characteristics are essential.

FFDNet may offer an average increase of 0.25dB by PSNR. The 14-layer FFDNet for color pictures functions somewhat more slowly than the 17-layer FFDNet for grayscale images. We set the total number of layer transformations at 12 and the number of maps of features at 96 for shade picture denoising, taking into consideration both denoising efficiency and performance.

Noise level map

To improve CNN-based denoiser's flexibility, noise level map and model-based image denoising approaches will be used to investigate the factors that contribute to their capacity to adjust to changes in noise. The majority of model-based noise reduction techniques seek to address the following issue.

$S = \arg min \times 1\ 2\sigma\ 2kb - sk + \lambda\phi(s)$ (2)

Where ϕ(s) is a regularization parameter related to the picture prior, the information's quality variable with the level of noise σ is 1 2σ2 kb − sk.

Equation (2)'s solution, using certain efficiency procedures, defines a hidden function provided by

S^=F (b, **σ**, λ; Θ). (3)

Eqn. (3)can be expressed as follows since λ is able to dissolve into σ:

$S^\wedge = F(b, \sigma; \Theta)$. (4)

In this way, adjusting the noise level σ also controls the trade-off between detail protection and reducing noise by adjusting λ.

Non-linear mapping

Convolution (Conv), BN, and ReLU activation functions are the three operations employed in the nonlinear transformation of the FFDNet network; the main concepts behind these operations are well explained here. Back-propagation is used to adjust the variables of each convolutional layer's many neural units. The primary aim of the convolution process is to identify the picture's characteristics, such as texture and edge features, and they are each assigned to a pixel in the image either alone or in combination. During development, the network's temporal difficulty increases since each layer's settings directly affect the input distribution of the layer [16].

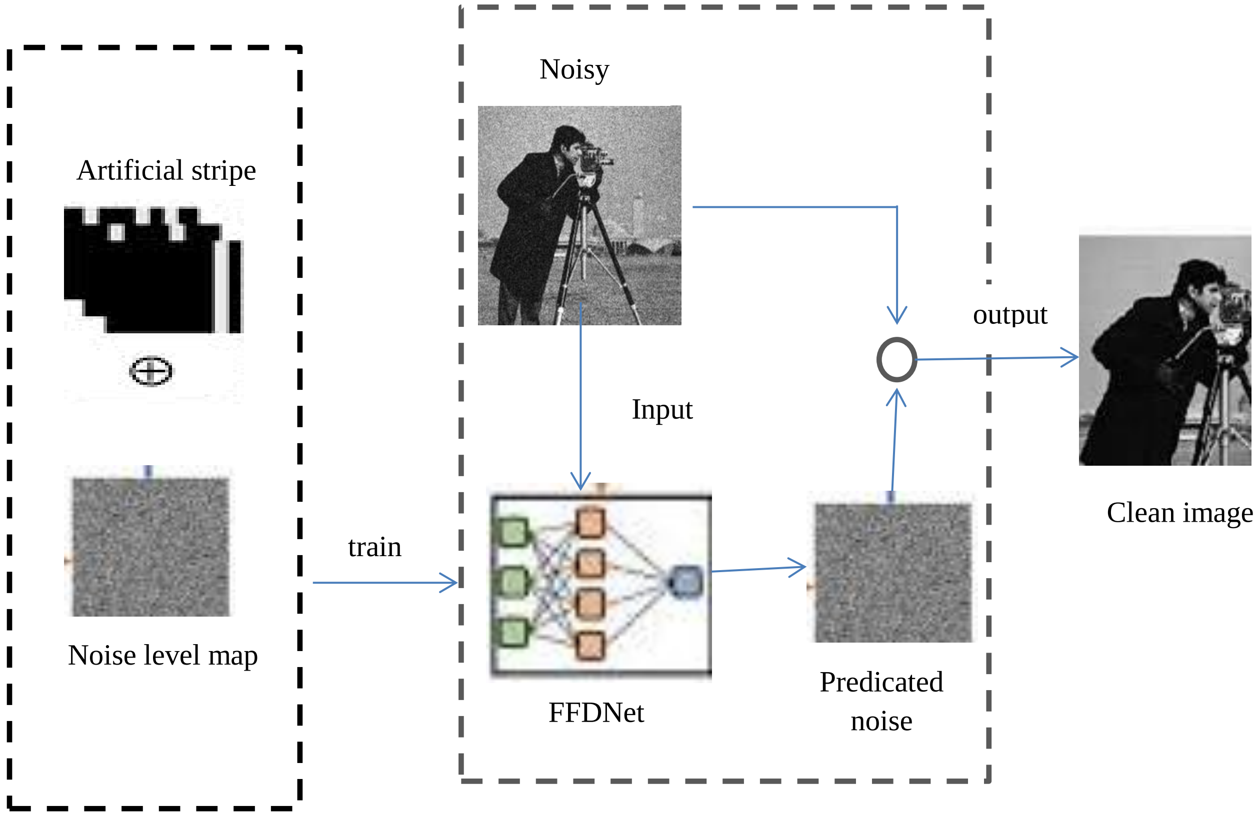

Figure 3. FFDNet Flow chart-based denoising method

Figure 3 shows the process of denoising based on the FFDNet network; FFDNet is utilized to address the shortcomings of current CNN-based noise reduction techniques. Both the sets for training and testing of the data required to train the network are produced by a computer. The input data set should not be too large or too small to prevent the trained network from being overfitted or underfitted. Consequently, following several experiments, 600 greyscale photos were selected as the number of data sets, with the images' sizes fixed at 200 × 200. FFDNet is designed to address the shortcomings of existing CNN-based denoising methods. It is expressed as follows: x = F, N; Θ, where x is the de-noised image, b is the unwanted image, N is a noise level map, and ˛ denotes model parameters. FFDNet is the name of the neural network used in this study that was learned on images that were neither compressed nor clipped [17]. We also train an FFDNet model, FFDNet-Clip, with the cropping parameters of a noisy image as a reference to precisely evaluate the proposed method. FFDNetClip quantizes the noisy pictures into an eight-byte representation for testing and training purposes.

4. EXPERIMENTAL RESULT AND ANALYSIS

An input-output pair training dataset {(bi, Ni; di)} RI=2 must be prepared for the FFDNet model. Here, Ni is the noise level map, and bi is produced by appending AWGN to latent image di. The noisy pictures $yi = bi = ui$ are trained in the FFDNet model with 8-bit integer numbers without normalization. Even though the learned model is often 8-bit quantized, we discovered that it continues to work well on actual noisy photos.

Table 1. Specification of FFDnet

FFDNet models	No. Of layers	No Of channels	Noise level image	Training patch size
Grayscale	17	66	[0,75]	80*80
Color	14	98	[0,75]	60*60

Several source photos comprised 4,844 images taken from the Waterloo Exploration databases, 400 BSD images, and 400 images chosen from Image Net's verification set. We randomly cut N = 128×8,000 sections of these images for training in each epoch. For grayscale and color pictures, we set the patch size to 70×70 and 50×50, respectively, because it must be more than the receiving field of FFDNet. The messy regions are created by adding AWGN with noise level σ ∈ [0, 75] to the clear sections. The regional connection feature of FFDNet, a fully convolutional neural network, shows that the final output pixel is affected by both local and regional noise levels. Spatially varying noise may be accommodated by the learned FFDNet by constructing an irregular noise level map. For clarity, the primary FFDNet model specifications are listed in Table 1.

Running time

The running time results of BM3D, DnCNN, and FFDNet for denoising color and grayscale level pictures with sizes of 256×256, 512×512, and 1,024×1,024 are shown in Table 2. A sophisticated deep-learning package is used to speed up DnCNN and FFDNet calculations. It also counts the duration that it takes for the CPU and GPU to transfer memory [19]. CPUs that contain multitasking (MT) and single-threaded (ST) implementations of DnCNN and FFDNet are possible.

Table 2. Running times (in seconds) of several denoising techniques for 256x256, 512x512, and 1,024x 1,024 pictures

Method	**Device**	**256x 256 Gray**	**Color**	**512x512 Gray**	**Color**	**1024x1024 Gray**	**Color**
BM3D	CPU(ST)	0.63	1.23	2.60	3.72	11.88	21.25
DnCNN	CPU(ST)	2.22	2.44	8.65	9.88	32.89	38.13
	CPU(MT)	0.97	1.23	3.17	4.18	12.14	15.84
	GPU	0.017	0.023	0.036	0.127	7.25	0.168
FFDNet	CPU(ST)	0.49	0.54	1.84	2.18	7.27	8.54
	CPU(MT)	0.19	0.25	0.78	0.83	2.98	3.18
	GPU	0.010	0.014	0.023	0.029	0.038	0.058

Table 2, demonstrates that BM3D initially takes longer to denoize color photographs than grayscale ones. The explanation is that following luminance-chrominance color transformation, CBM3D requires more time to denoise the brightness components than gray-BM3D. Second, DnCNN has a similar CPU time to BM3D, even though it can use GPU processing for speedy implementation. Third, FFDNet converts color and grayscale images at almost the same time. In particular, FFDNet's multitasking solution outperforms DnCNN and BM3D by around three times on the CPU and significantly outperforms DnCNN on the GPU.

Implementation

The Gaussian denoiser is trained on 400 180 × 180 photos using a specified noise level. It has been discovered that, particularly on the BSD68 test, employing a bigger training dataset can only result in lower gains. Three models are trained using four noise levels (σ = 15, 25, 50, and 70). The spot is 40 by 40 in size. The system is trained using 600 128 × 1 patches, with a mini-batch size of 128. The average adjusted error among the desired remaining images and remembered images computed from noisy data.

Table 3. The PSNR (dB) results based on the BSD68 dataset

Noise level	**BM3D**	**DnCNN**	**FFDNet**
σ =15	31.07	31.73	38.08
σ =25	28.57	30.23	37.63
σ =50	25.62	27.23	37.42
σ =70	24.43	26.79	36.49

A dataset of 68 actual photos from the Berkeley segmentation dataset (BSD68) is used to assess the model. Table 3 displays the median PSNR outcomes of the various techniques. As can be seen, the approaches BM3D and DnCNN have a PSNR increase of around 0.35dB compared to the benchmark BM3D. Interestingly, on all three noise levels, FFDNet performs 0.6dB better than DnCNN. FFDNet is quicker than BM3D, even when implemented in a single thread. Considering the flexibility and denoising performance, In terms of real-world applications, FFDNet is extremely strong.

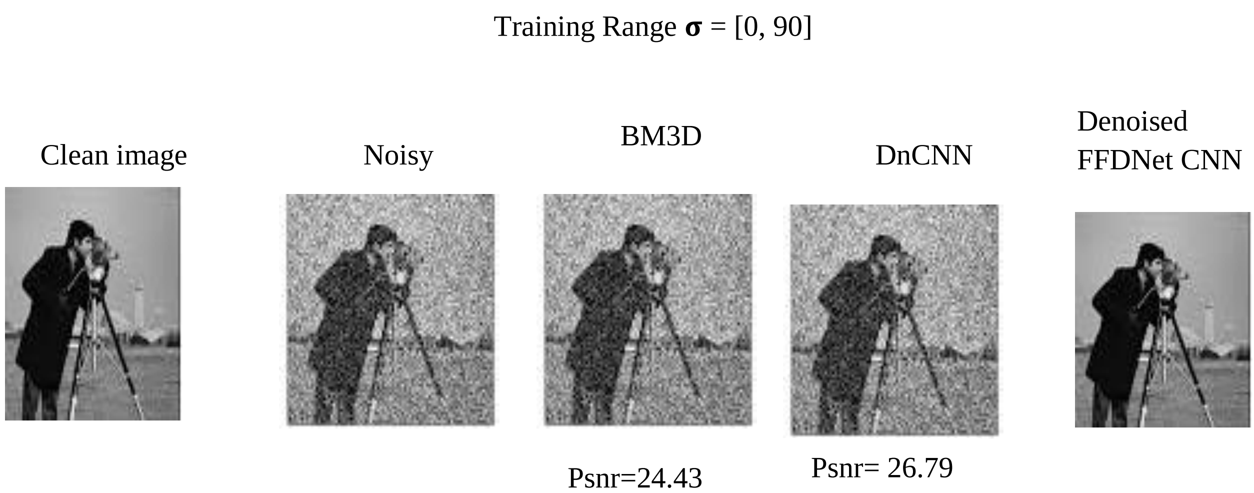

Figure 4. Denoising results proposed on noise level 70.

Figure 4 shows the visual outcomes of several techniques. In many cases, BM3D, DnCNN, and MLP provide too smooth textures and edges. DnCNN is likely to preserve delicate features and sharp edges while producing artifacts in the zone. On the other hand, FFDNet produces aesthetically pleasing results in the smooth zone in addition to recovering fine details and sharp edges.

Figure 5 shows PSNR results based on BSD68. BSD68 pictures with noise levels ranging from 0 to 60 are used to evaluate several techniques with various source noise values (for example, "FFDNet-15" denotes FFDNet with source noise level set at 15). When the input noise levels of BM3D and DnCNN are equal, FFDNet produces denoising results comparable across all noise levels. When the input noise level is fixed for all three methods, the PSNR value tends to stay constant at lower ground truth noise levels and begins to decrease with higher ground truth noise stages.

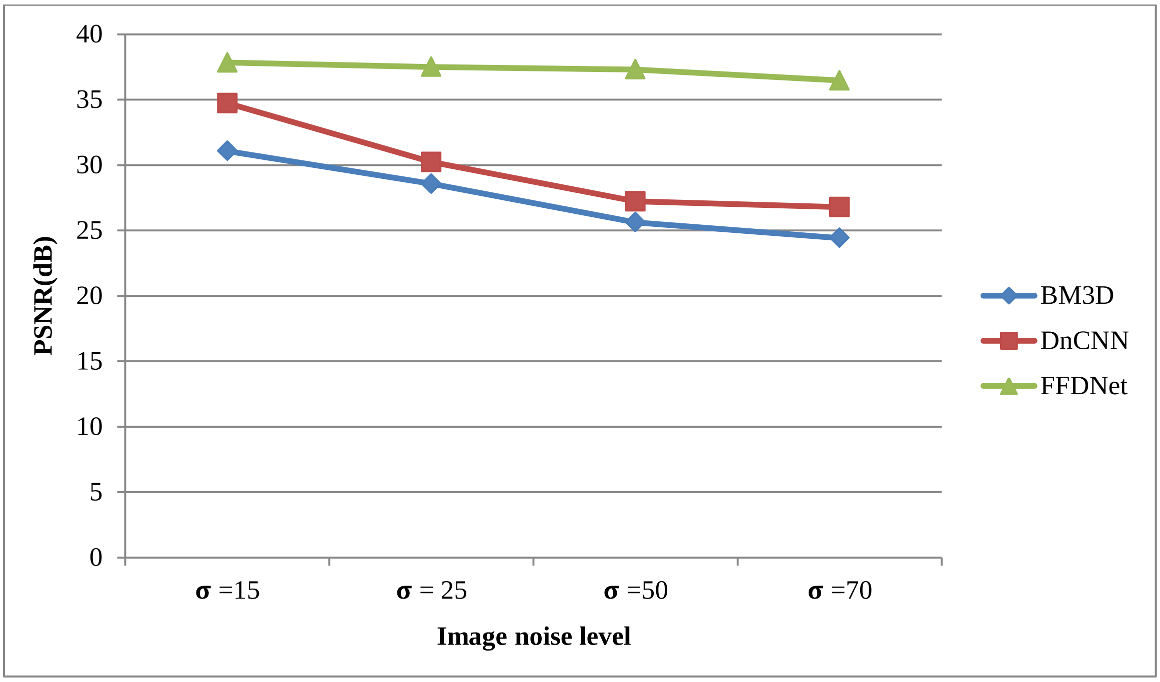

Figure 5. PSNR results based on BSD68

When the signal's noise level matches that of the ground truth, the best visual quality is obtained. While BM3D and FFDNet provide similar visual results at lower intake noise levels, they exhibit notable variances at higher input noise levels. Because BM3D uses non-local information, it can preserve the original patch independent of the input noise levels, although FFDNet may eliminate some low-contrast line structures.

In general, larger input noise levels can yield greater visual outcomes than less noise. Furthermore, the visual difference is minimal when the data noise level is marginally higher than the ground-truth value. Based on the results mentioned above, FFDNet performs similarly to BM3D and DnCNN noise level sensitivity when it comes to striking a balance between detail retention and a decrease in noise.

5. CONCLUSION

In this proposed work several CNN models for image denoising are shown, including FFDNet, which offers quick, efficient, and adaptable selective denoising. The PSNR on BSD-68 is used to compare the performance of each FFDNet CNN noise. Several methods were used in network design and training, including denoising in downsampled sub-image area and using a level of noise map as input. Lastly, the running time comparisons demonstrated FFDNet's superior performance over other techniques like BM3D. Average running time on a GPU, the FFDNet model is the fastest. For denoising, artificial pictures manually warped by users, DnCNN, FFDNet, and BM3D have been demonstrated to be more effective. According to the results, FFDNet is appealing for real-world denoising applications since it is effective and efficient.

REFERENCES

[1] Ilesanmi, A. E., & Ilesanmi, T. O. (2021). Methods for image denoising using convolutional neural network: a review. *Complex & Intelligent Systems*, *7*(5), 2179-2198.

[2] Goyal B, Dogra A, Agrawal S, Sohi BS, Sharma A (2020) Image denoising review: from classical to state-of-the-art approaches. Inform Fusion 55:220-244

[3] Farooq, Y., & Savaş, S. (2024). Noise removal from the image using convolutional neural networks-based denoising autoencoder. *Journal of Emerging Computer Technologies*, *3*(1), 21-28.

[4] Shen, Z., Qin, F., Ge, R., Wang, C., Zhang, K., & Huang, J. (2025). IDTransformer: Infrared image denoising method based on convolutional transposed self-attention. *Alexandria Engineering Journal*, *110*, 310-321.

[5] Magsi, H., Sodhro, A. H., Chachar, F. A., & Abro, S. A. K. (2018, March). Analysis of signal noise reduction by using filters. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)* (pp. 1-6). IEEE.

[6] Agrawal, P., & Verma, J. S. (2013). A survey of linear and non-linear filters for noise reduction. *International Journal of Advance Research in Computer Science and Management Studies*, *1*(3).

[7] Albelwi, S., & Mahmood, A. (2017). A framework for designing the architectures of deep convolutional neural networks. *Entropy*, *19*(6), 242.

[8] Wang, Z., Wang, L., Duan, S., & Li, Y. (2020, May). An image denoising method based on deep residual GAN. In *Journal of* Physics*: conference series* (Vol. 1550, No. 3, p. 032127). IOP Publishing.

[9] Chen, Y., Zhou, G., Li, L., Chen, A., Wang, Y., & Li, L. (2024). A high-quality self-supervised image denoising method based on SDDW-GAN and CHRNet. *Expert Systems with Applications*, *258*, 125157.

[10] Jiongkai, Y., Wu, Y., & Peng, D. Low-quality image Object Detection Based on Reinforcement Learning Adaptive Enhancement. *Available at SSRN 4493593*.

[11] Luo, W., Ye, S., Zhang, Z., Liu, S., Xiong, W., Wang, X., ... & Dong, B. (2024). A study on denoising with deep convolutional neural networks in spatial heterodyne spectroscopy. *Journal of Quantitative Spectroscopy and Radiative Transfer*, *316*, 108918.

[12] Du, J., Qiao, X., Yan, Z., Zhang, H., & Zuo, W. (2024). Flexible image denoising model with multi-layer conditional feature modulation. Pattern Recognition, 152, 110372.

[13] Thakur, R. S., Yadav, R. N., & Gupta, L. (2019). State-of-art analysis of image denoising methods using convolutional neural networks. *IET Image Processing*, *13*(13), 2367-2380

[14] Zhang, K., Zuo, W., & Zhang, L. (2018). FFDNet: Toward a fast and flexible solution for CNN-based image denoising. IEEE Transactions on Image Processing, 27(9), 4608-4622.

[15] Yang, X., Li, S., Xu, Y., Liu, Z., & Qin, Z. (2023, October). Multi-event location denoising scheme for φ-OTDR based on FFDNet network. In Photonics (Vol. 10, No. 10, p. 1114). MDPI.

[16] Nandihal, P., Sreenivas, V., & Pujari, J. (2019). Application of FFDNET for Image Denoising On Microarray Images [J]. International Journal of Recent Technology and Engineering, 8(3), 2691-2694.

[17] Karaca, N., & Çiftçi, S. (2023, December). Image denoising with CNN-based attention. In 2023 4th International Informatics and Software Engineering Conference (IISEC) (pp. 1-6). IEEE.

[18] Agarkar, A., Shrivastava, S., Bangde, D., & Bhise, D. (2024, March). CNN- based Image-Denoising Techniques-A Review. In 2024 3rd International Conference on Sentiment Analysis and Deep Learning (ICSADL) (pp. 146-152). IEEE.

[19] Zuo, W., Zhang, K., & Zhang, L. (2018). Convolutional neural networks for image denoising and restoration. Denoising of photographic images and video: Fundamentals, open challenges, and new trends, 93-123.