Convolutional Neural Networks for Noise Classification and Denoising of Images

Dibakar Sil, Arindam Dutta and Aniruddha Chandra
National Institute of Technology, Durgapur 713209, India
{ds.20150096, ad.20150021}@btech.nitdgp.ac.in, aniruddha.chandra@ieee.org

***Abstract*—The goal of this paper is to find whether a convolutional neural network (CNN) performs better than the existing blind algorithms for image denoising, and, if yes, whether the noise statistics has an effect on the performance gap. For automatic identification of noise distribution, we used two different convolutional neural networks, VGG-16 and Inception-v3, and it was found that Inception-v3 identifies the noise distribution more accurately over a set of nine possible distributions, namely, Gaussian, log-normal, uniform, exponential, Poisson, salt and pepper, Rayleigh, speckle and Erlang. Next, for each of these noisy image sets, we compared the performance of FFDNet, a CNN based denoising method, with noise clinic, a blind denoising algorithm. It was found that CNN based denoising outperforms blind denoising in general, with an average improvement of 16% in peak signal to noise ratio (PSNR). The improvement is however very prominent for salt and pepper type noise with a PSNR difference of 72%, whereas for noise distributions such as Gaussian, FFDNet could achieve only a 2% improvement over noise clinic. The results indicate that for developing a CNN based optimum denoising platform, consideration of noise distribution is necessary.**

***Index Terms*—Image denoising, noise statistics, convolutional neural networks, VGG-16, Inception-v3, FFDNet.**

I. Introduction

A. Background

Noise accumulation in images are caused at different stages of capturing an image, which includes acquisition, quantization, formatting and compression. In some cases, noise accumulation is unavoidable due to environmental constraints, for example, there is a maximum radiation limit for computer tomography (CT) scan which results in low-contrast images, or if we take the case of outdoor surveillance, haze or other weather conditions result in inconsistency in image quality. Noisy medical imaging may lead to inaccurate diagnosis of the patient [1]. On the other hand, if satellite images are not denoised, several crucial geographical information will be lost [2]. Denoising is equally important for mission-critical defence applications [3] and for non-critical industrial applications [4]. Noise classification allows us to identify the most probable noise distribution present in a noisy image after which an optimal denoising algorithm may be applied to denoise the image. The method is also used across other signal processing disciplines. In phase retrieval problems, if the system is influenced by Laplace noise, we would do ℓ_1-norm optimization rather than taking ℓ_2-norm, which is generally considered for the Gaussian case.

B. Related Work

Image denoising has gained a lot of importance over the last few years. A complete review of many possible methods that can be applied for image denoising is presented in [5], and more recently, in [6]. The methods range from image enhancement by virtue of independent component analysis (ICA) mixture models as demonstrated in [7] to image denoising based on support vector machine (SVM) classification [8]. Some denoising methods are also application specific. In the domain of hyperspectral remote sensing, unwanted environmental and atmospheric conditions degrade the hyperspectral image analysis, and simultaneous spatial and spectral low rank representations can effectively deal with the problem [9]. The development of image denoising algorithms for ultrasound images have rendered a lot of help to medical diagnosis of breast cancer [10]. The domain is ever evolving, and a recent addition in this regard is the use of convolutional neural networks (CNNs). In [11], authors describe a CNN-based image denoising method and showed its superiority over traditional methods.

C. Our Model

In this paper, we employed CNN-based architectures for both noise classification and denoising. Fig. 1 shows the complete method employed for the task in the form of a flow diagram. The first step in noise classification is transfer learning, i.e. training the neural network with a training image data set. It is followed up by passing images from the test data set, where the network predicts the type of noise distribution present in each image. After that, these noisy images are denoised using blind denoising algorithm and CNN-based image denoising algorithm, to obtain denoised images.

D. Contributions

The specific contributions of this paper are the following:

- We have classified noisy images using CNNs. In our work, we have used two deep learning architectures, VGG-16 and Inception-v3. The CNNs help to classify a total of nine types of noise distributions, Gaussian, log-normal, uniform, exponential, Poisson, salt and pepper, Rayleigh, speckle and Erlang. In contrast, most of the related text considered only two or three types of noise distributions [12]–[14].
- We have also denoised the images using two methods, blind denoising and FFDNet denoising, and compared

978-1-7281-1895-6/19/$31.00 ©2019 IEEE

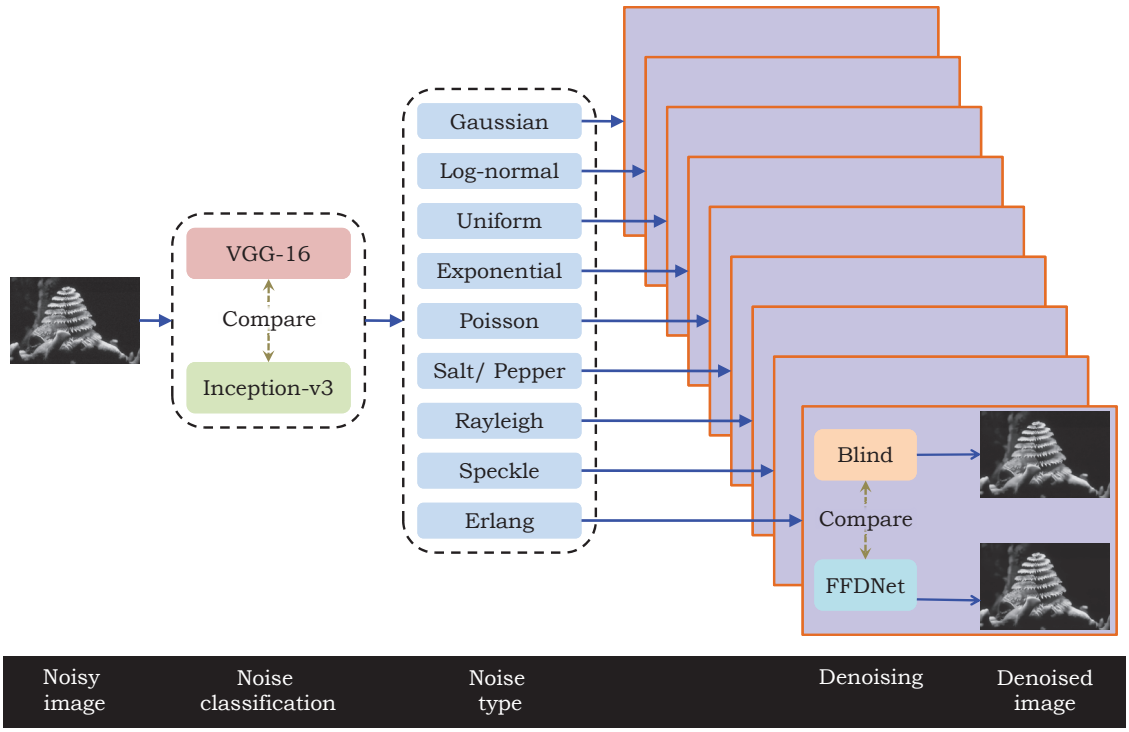

Fig. 1: Noise classification and denoising model.

the results. FFDNet uses the recently introduced CNN architecture for image denoising. Most of the existing work is centred around classical approaches to denoise the noisy images [15]–[17].

E. Organization

The rest of the paper is organized in the following fashion. In Section II, we focus on CNN-based noise classification. Architecture of the CNN-based frameworks, preparation of dataset, training and testing of the neural networks and related performance plots are included. In Section III, we have discussed the two denoising algorithms that have been used in this paper alongwith the comparative results. Section IV concludes the paper.

II. Noise Classification

Our first goal is to identify the type of noise distribution present in a given noisy image among nine possible noise distributions. Hence, we formulate the given problem as a classification problem, that is we put forward an effort to identify the type of noise present in any given random image. Dealing any classification problem through Deep Learning (DL) methods have been very popular nowadays as they produce far more accurate results than traditional machine learning (ML) algorithms. The basic logic to solve classification problems involve two crucial steps: training and testing. We prepared a data set [18] containing 14000 images, with one of the nine possible noise distributions present in each of the images. Out of the total 14000 images, 12000 of them were used for training and the other 2000 were used for testing.

A. CNN Architectures

A convolutional neural network (CNN), like any other deep neural network, consists of an input layer, an output layer and many hidden layers in between. Feature detection layers are one of the three following types, convolution, pooling and rectified linear unit (ReLU):

- Convolution activates certain features from the image by passing the image through convolutional filters.
- Pooling reduces the number of features that the network needs to learn about by virtue of non-linear down-sampling.
- ReLU further introduces non-linearity in the image by maintaining non-negative values and mapping others to zero, thus providing further scope for effective training.

These three operations are performed over a number of times after which the CNN shifts to classification, which is in turn done by the fully connected penultimate layer. The final layer of the CNN architecture uses a Softmax function to provide classification output.

During our study we trained and tested various CNNs, and found that VGG-16 and Inception-v3 are best suited for the purpose as their accuracy was well above other networks like CapsNet[1]. Noise identification accuracy with CapsNet was

[1]CapsNet realizes dynamic routing between capsules, where each capsule is a collection of neurons whose activity vector denotes the parameters of instantiation of an object. Translational invariance is achieved by removing the maxpooling layers and by relying on dynamic routing protocol between the capsules.

found to be totally random and we achieved an accuracy of 11% after 20 epochs[2].

B. VGG-16 and Inception-v3 Architectures

VGG-16 is a deep CNN architecture developed by visual geometry group (VGG), university of Oxford [19]. It has 16 convolutional layers and consists of very small convolutional filters with 3 fully connected layers. If input, output, pooling and activation are considered, then this model has 41 layers. This model has performed better than many well-known CNNs like Caffe Reference Model and AlexNet in ImageNet classification tasks.

The Inception-v3 [20] is derived from GoogLeNet. It was developed by Google Inc. and is a 42-layer deep neural network model, similar in complexity to VGG-16. The model has a variety of convolution windows and in this work we used 1×1, 3×3 and 5×5 along with a 3×3 max pooling. Following this, the resulting feature map has been passed through a ReLU and then the larger convolutions (5×5 or 3×3) are performed. This method significantly reduces the number of computations than what we got if we did the larger convolutions initially.

C. Noise Models

For noise models we consider the widely used Gaussian distribution along with some other common statistical models [21] such as exponential and log-normal. In laser based images, Erlang and speckle noise are encountered and we consider them as well. Poisson and Rayleigh noise are common in medical images and underwater images, respectively. Again, quantization noise introduced to an image is often uniformly distributed. Finally, we also include salt and pepper noise which accounts for the unanticipated and sharp disruptions, changing the pixel value to either the maximum or the minimum value.

D. Dataset Preparation

Noise with different power spectral densities (PSDs) are implemented randomly on the Berkeley segmentation dataset BSDS300 (321×481). NumPy library in Python is used to generate noise following a given distribution by varying mean and standard deviations of the noise PSD. For each of the noise distribution types, there are 1200 to 1500 images [18].

E. Results and Discussion

The noise classification involves both training and testing of the dataset. For training our CNN models, we used sequential training, where all the weights are updated after each training vector is sequentially passed through the model. This is different from batch training where all the samples are put through the model simultaneously in one epoch. The testing dataset is separate from the training dataset, and provides insight on how effective our model performs on unknown data. During training and testing, for VGG-16, images are resized to 224×244 and for Inception-v3, they are resized to 299×299. This is done because both these CNNs are very sensitive to the size of the input image and refrain to act on the images which do not fall into their specified resolution.

The data obtained from training and testing are used to plot Fig. 2-5, where accuracy and loss performance of VGG-16 and Inception-v3 are shown. For VGG-16, both the accuracy and loss performances show erratic changes whereas for Inception-v3, both accuracy and loss behaviours are relatively steady. After 30 epochs, VGG-16 architecture gives a training accuracy of 74.17% and a test accuracy of 67.99%. We can see that the Inception-v3 architecture performs better by reaching a training accuracy of 77.08% and a test accuracy of 68.35% in 30 epochs. Hence, we can conclude that Inception-v3 architecture performs better for the given noise classification task. There is, however, one concern regarding Inception-v3; the gap between training and testing performance increases with the number of epochs, thus implying that the model overfits the training data.

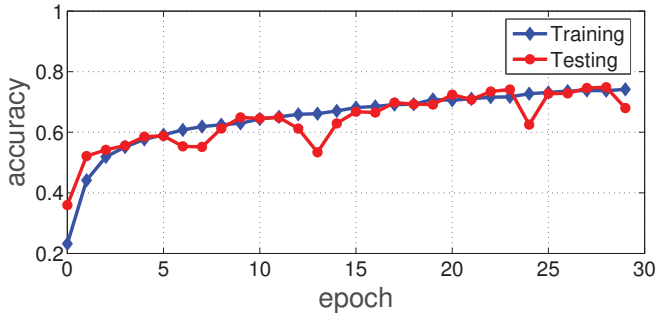

Fig. 2: Accuracy performance of VGG-16.

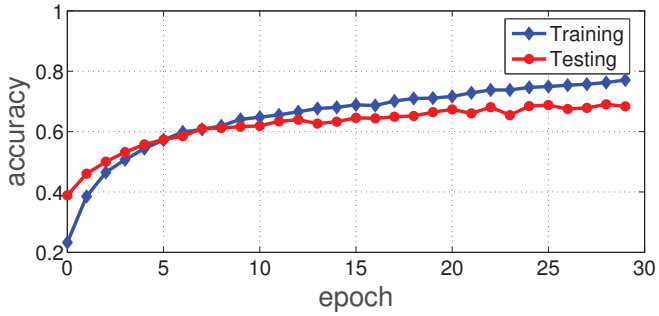

Fig. 3: Accuracy performance of Inception-v3.

III. Denoising Images

After the classification of noise distribution, the next step is to denoise the noisy images for each class of noise. We have used blind denoising algorithm first and then compared the results with a CNN-based image denoising algorithm.

A. Blind Denoising

For blind denoising, we have chosen Noise Clinic [22] which has proved itself to be superior than many well known

[2]An epoch measures the number of times the complete training dataset are used to update the weights.

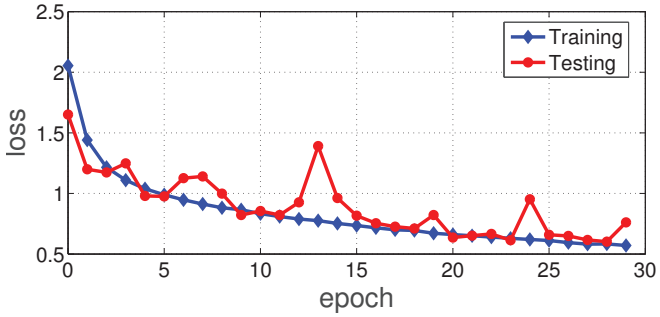

Fig. 4: Loss performance of VGG-16.

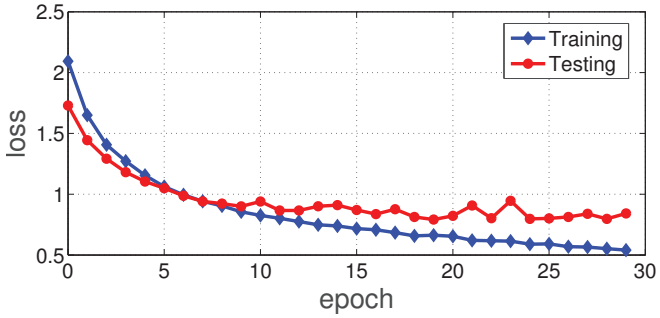

Fig. 5: Loss performance of Inception-v3.

image denoising softwares in the market. In this algorithm, after a digital image is taken in as input, a signal and frequency dependent (SFD) noise model is estimated. After that, the image is denoised by virtue of multi-scale adaptation of a non-local Bayes denoising method. The algorithm works well even with images in low light and with low quality old images. The algorithm is fully automatic and the only major user defined parameter that the algorithm takes in is the number of scales.

B. CNN-based Denoising

For CNN-based image denoising, we have chosen FFDNet, which stands for fast and flexible denoising convolutional neural network. Extensive experimentation [23], [24] proves that FFDNet is highly reliable for practical blind denoising applications. It works on down-sampled sub-images, hence achieving an optimum balance between computational time and denoising performance. It can handle wide range noise levels and can deal effectively with spatially variant noise. It is much faster and accurate in performance as compared to the benchmark block-matching and 3D filtering (BM3D) algorithm.

C. Results and Discussion

Fig. 6 compares the performance of algorithms following blind denoising and CNN-based denoising. We have included results for only a single class of noise. The full graphical comparison for all nine classes of noise is uploaded to IEEE DataPort site [25] due to space constraints. A simple visual inspection of Fig. 6 reveals that the CNN-based algorithm performs better in removing noise, although some details (see the zoomed in regions) are preserved better in blind denoising.

TABLE I: PSNR and SNR with respect to the clean image.

	Noisy image		Blind denoising		CNN-based denoising	
Noise Type	PSNR	SNR	PSNR	SNR	PSNR	SNR
Gaussian	18.60	11.54	19.07	12.01	19.44	12.37
Log-normal	13.07	6.00	14.98	7.92	15.98	8.91
Uniform	14.60	8.13	15.26	8.79	15.75	9.28
Exponential	17.70	11.10	18.53	11.92	19.99	12.38
Poisson	28.03	21.42	29.30	22.69	31.28	24.67
Salt/ Pepper	14.26	7.65	14.64	8.03	25.18	18.57
Rayleigh	16.46	9.91	18.57	10.09	19.82	11.22
Speckle	15.06	8.51	16.59	9.08	19.71	13.16
Erlang	25.74	17.20	26.54	18.00	31.09	22.54

The difference between the two methods is better illustrated in a quantitative manner through Table I, where average signal to noise ratio (SNR) and average peak signal to noise ratio (PSNR) is noted for all nine different noise distributions. It can be observed from the PSNR values that CNN-based denoising clearly performs better than blind denoising for all the noise types. The degree of improvement, however, varies widely. CNN-based denoising performed much better than blind denoising for salt and pepper, speckle and Erlang noises. For log-normal, exponential, Poisson and Rayleigh the improvement was moderate, whereas for Gaussian and uniform noise, the improvement was quite low.

IV. Conclusion

The major conclusions of this paper are twofold. First, we compare the performance of two CNN-based architectures for detecting noise distribution in noisy images and demonstrate that Inception-v3 architecture performs better than VGG-16. After 30 epochs, the training accuracy of Inception-v3 is about 3% better than VGG-16 and for the same epochs, the testing accuracy is improved by 1%, when Inception-v3 is chosen over VGG-16. Second, we show that knowledge of the particular noise distribution is helpful in denoising of images. When we trained a CNN-based architecture, FFDNet, seprately for different classes of noise distribution, it showed an improvement of 16% in PSNR over blind denoising. However, it is observed that in most of the cases, PSNR values are well below 30 dB, which implies that there is an ardent need for a specific neural network architecture to serve the purpose of detection of particular noise distribution type present in the image and act accordingly to denoise it.

References

[1] Q. Yang, P. Yan, Y. Zhang, H. Yu, Y. Shi, X. Mou, M. K. Kalra, Y. Zhang, L. Sun, and G. Wang, "Low dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1348–1357, Jun. 2018.

[2] A. Masse, S. Lefèvre, R. Binet, S. Artigues, G. Blanchet, and S. Baillarin, "Denoising very high resolution optical remote sensing images: Application and optimization of nonlocal Bayes method," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 3, pp. 691–700, Mar. 2018.

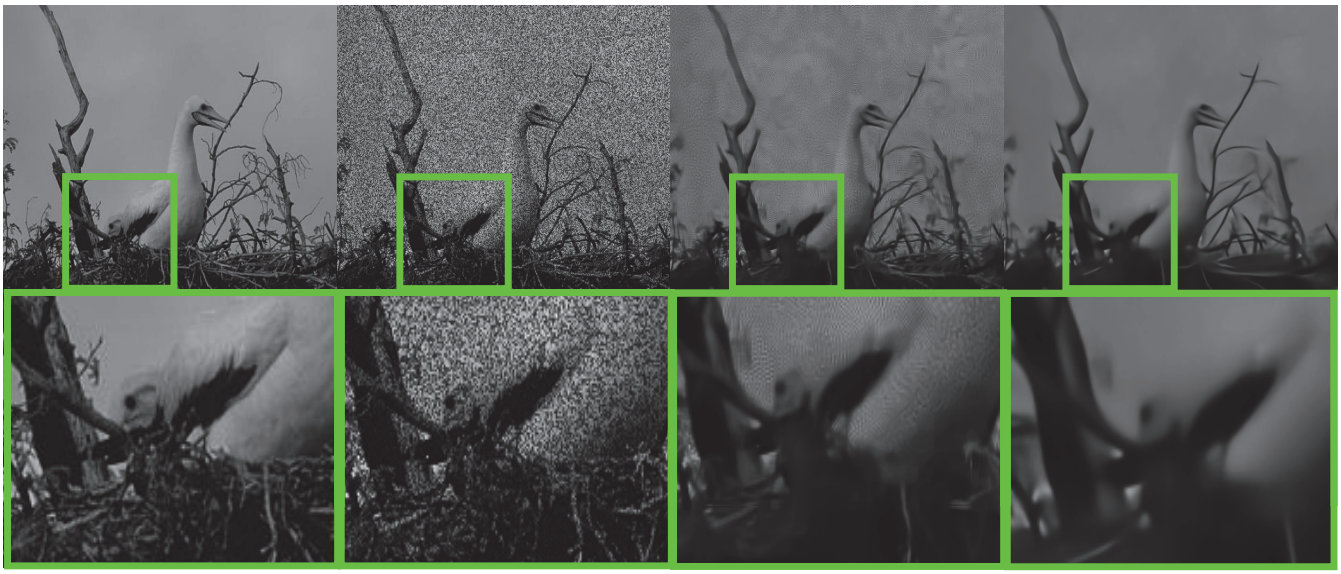

Fig. 6: Denoising results for speckle noise. From left to right: original image, noisy image, blind denoising, CNN-based denoising. The bottom row shows zoomed-in version of the region inside the green square. Image used: image 8049 [gray] from BSDS300 dataset. Images for all nine noise types are made available in [25].

[3] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu, "Defense against adversarial attacks using high-level representation guided denoiser," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, Utah, USA, Jun. 2018, pp. 1778–1787.

[4] L. B. A. Hamid, N. R. Rosli, A. S. M. Khairuddin, N. Mokhtar, and R. Yusof, "Denoising module for wood texture images," *Wood Science and Technology*, vol. 52, no. 6, pp. 1539–1554, Nov. 2018.

[5] M. C. Motwani, M. C. Gadiya, R. C. Motwani, and F. C. Harris, "Survey of image denoising techniques," in *Proc. Global Signal Processing Expo and Conference (GSPx)*, Santa Clara, California, USA, Sep. 2004, pp. 27–30.

[6] M. Mafi, H. Martin, M. Cabrerizo, J. Andrian, A. Barreto, and M. Adjouadi, "A comprehensive survey on impulse and Gaussian denoising filters for digital images," *Signal Processing*, vol. 157, pp. 236–260, Apr. 2019.

[7] T. W. Lee and M. S. Lewicki, "Unsupervised image classification, segmentation, and enhancement using ICA mixture models," *IEEE Transactions on Image Processing*, vol. 11, no. 3, pp. 270–279, Mar. 2002.

[8] S. Routray, A. K. Ray, C. Mishra, and G. Palai, "Efficient hybrid image denoising scheme based on SVM classification," *Optik*, vol. 157, pp. 503–511, Mar. 2018.

[9] S. Mei, J. Hou, J. Chen, L. P. Chau, and Q. Du, "Simultaneous spatial and spectral low-rank representation of hyperspectral images for classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 5, pp. 2872–2886, May 2018.

[10] H. D. Cheng, J. Shan, W. Ju, Y. Guo, and L. Zhang, "Automated breast cancer detection and classification using ultrasound images: A survey," *Pattern Recognition*, vol. 43, no. 1, pp. 299–317, Jan. 2010.

[11] Z. Liu, W. Q. Yan, and M. L. Yang, "Image denoising based on a CNN model," in *Proc. IEEE International Conference on Control, Automation and Robotics (ICCAR)*, Auckland, New Zealand, Apr. 2018, pp. 389–393.

[12] A. J. Ahumada, "Classification image weights and internal noise level estimation," *Journal of Vision*, vol. 2, no. 1, pp. 121–131, Jan. 2002.

[13] S. H. W. Scheres, M. Valle, R. Nuñez, C. O. S. Sorzano, R. Marabini, G. T. Herman, and J. M. Carazo, "Maximum-likelihood multi-reference refinement for electron microscopy images," *Journal of Molecular Biology*, vol. 348, no. 1, pp. 139–149, Apr. 2005.

[14] O. Magud, E. Tuba, and N. Bacanin, "An algorithm for medical ultrasound image enhancement by speckle noise reduction," *International Journal of Signal Processing*, vol. 1, pp. 146–151, Aug. 2016.

[15] Q. Lin and J. Allebach, "Combating speckle in SAR images: Vector filtering and sequential classification based on a multiplicative noise model," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Vancouver, Canada, Jul. 1989, pp. 1016–1019.

[16] S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1532–1546, Sep. 2000.

[17] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.

[18] D. Sil, "Noisy Image Dataset," Nov. 2018, Kaggle. [Online]. Available: https://www.kaggle.com/dibakarsil/9-classes-noisy-image-dataset

[19] W. Yu, K. Yang, Y. Bai, T. Xiao, H. Yao, and Y. Rui, "Visualizing and comparing AlexNet and VGG using deconvolutional layers," in *Proc. International Conference on Machine Learning (ICML)*, New York City, NY, USA, Jun. 2016, pp. 1–7.

[20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2818–2826.

[21] C. Boncelet, "Image noise models," in *The Essential Guide to Image Processing*, A. C. Bovik, Ed. Academic Press, Jul. 2009, ch. 7, pp. 143–167.

[22] M. Lebrun, M. Colom, and J. M. Morel, "The noise clinic: A universal blind denoising algorithm," in *Proc. IEEE International Conference on Image Processing (ICIP)*, Paris, France, Oct. 2014, pp. 2674–2678.

[23] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN based image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[24] M. Tassano, J. Delon, and T. Veit, "An analysis and implementation of the FFDNet image denoising method," *Image Processing On Line*, vol. 9, pp. 1–25, Jan. 2019.

[25] D. Sil, A. Dutta, and A. Chandra, "CNN based noise classification and denoising of images," Mar. 2019, IEEE Dataport. [Online]. Available: http://dx.doi.org/10.21227/3m26-dw82