



A Generalized Deep Learning Model for Denoising Image Datasets

Kurian Thomas, Pranav E., Supriya M.H.

Abstract: *Advance in technology world has lots of contributions from artificial intelligence which is a highly growing area. The failure of traditional algorithms has led to the employment of deep learning algorithms in various fields like pattern recognition, recommendation systems and classification systems. Removal of noise from images can be done using traditional noise removal filters. These filters can either remove more noise than wanted or leave unwanted noise than what is needed in the data. Utilization of Convolutional neural networks designed based on the dataset requirements along with the noise removal filter can yield better results. In this work, evaluation of the performance of convolutional neural network (CNN) against existing image denoising algorithms has been successfully executed. The proposed model is a generalized CNN model which can recognize and classify any type of noisy image given. Two types of model were compared where one model 1 uses the Adam optimizer and model 2 uses the Stochastic Gradient Descent (SGD) optimizer. The image dataset used here is MNIST handwritten dataset, which is trained, tested and validated with both the models by adding three different types of noise viz, Poisson, Salt and Pepper as well as Gaussian Noise. More accuracy and better results were given by the model 2 which uses the SGD optimizer.*

Keywords: Adam, Convolutional Neural Network, Classification, SGD.

I. INTRODUCTION

As technology progresses and the world becomes more and more virtual, there is a fear that we will lose the human connection and communication; but what if our devices could replace those interactions?

Acquisition, quantization, formatting and compression, the different stages of capturing an image, can lead to noise accumulation in images. Environmental constraints can lead to unavoidable noise accumulation. Typically a noisy medical imaging can result in inaccurate diagnosis of the patient. Without proper denoising of satellite images, crucial geographical information can be lost. Non-critical industrial applications as well as mission-critical defence applications need de-noising. On application of Optimal de-noising algorithm on images following classification of noise, identification of probable noise distribution present in abundance in the noisy image becomes possible. For the success of an image processing system, the pre processing stage which involves noise filtration will directly influence

the subsequent process of segmentation of objects, the detection of edges, extraction of features and so on. Spatial domain and frequency domain are the two forms of filtering which is usually applied on the 2D visual signals. Human brain inspired deep learning algorithms makes use an artificial neural network. One class of deep neural network, Convolutional Neural Network (CNN), is based on mathematical operation of convolution. 2D CNNs are basically used for the recognition task as the dataset consists of images. The proposed deep CNN model classifies and recognizes ten different classes of images of handwritten digits which have been added with noise. However, noise removal from images has been achieved using a linear CNN for image has been used because traditional de-noising methods which rely on image filtering failed in tuning of the parameters.

II. RELATED WORKS

Usage of denoising in convolutional neural networks can generate a modification signal in the info-communication system transmitting a image as proposed by O. Sheremet et al [1]. Fine-tuning for special filtering tasks can be carried out on a diverse image dataset using pre-trained correction elements. Loss function modification for Denoising Auto-encoders using entropy maximization principle, with constraint conditions of residual statistics has been proposed by Q. Xiang et al [2]. Improved Denoising Auto-encoders has been introduced using non-existent artifacts and is more robust than other state-of-the-art denoising methods with respect to evaluation parameters like SSIM and PSNR indexes. S. Suresh et al [3] has proposed a vision based system, a Convolutional Neural Network (CNN) model, which uses the images captured and can identify six different sign languages. The Stochastic Gradient Descent (SGD) and Adam are the different type of optimizers used for development of two CNN models. Z. Liu et al [4] designed and implemented the de-noising method based on a linear CNN model which will effectively remove Gaussian noise and improves the performance of traditional image filtering methods significantly. The Deep Convolutional Neural Network (DCNN) model created by E. Pranav et al [5] classifies five different human facial emotions. The manually collected image dataset has been used for training and has been tested and validated. Analysis on different signals contaminated with noise has been carried out by P. Dautov et al [6]. Daubechies family along with the usage of soft thresholding and application of mother wavelets for the studies has been rewarding.

Revised Manuscript Received on September 15, 2020.

* Correspondence Author

Kurian Thomas, Department of Electronics, Cochin University of Science & Technology, Kochi, India. Email: kurian@xtendtech.com

Pranav E., Department of Electronics, Cochin University of Science & Technology, Kochi, India. Email: pranaveswr96@cusat.ac.in

Supriya M.H.*, Department of Electronics, Cochin University of Science & Technology, Kochi, India. Email: supriya@cusat.ac.in

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Success of signal denoising applications is dependent on the correct selection of right thresholding technique and wavelet family.

III. PROPOSED GENERALISED DENOISER

Complex problems such as pattern recognition, video as well as image classification etc. need a better approach than traditional neural network. CNN while yielding good accuracy has achieved great success in these applications. Stacking of several building blocks like convolution layers, pooling layers typically max pooling, and fully connected (FC) layers build up the CNN. Loss function through forward propagation on a training dataset is calculated under particular kernels and weights and is used to analyse the model's performance. The learnable parameters, i.e., kernels and weights, are updated using loss function and gradient descent optimization algorithm. The features of interest are represented by each convolution filter and the algorithm learns from the feature. Based on the filter values, which transforms the entire image., the convolutional layer consists of small patches.

Feature maps, can be created using Equation (1)

$$G[m, n] = (f * h)[m, n] = \sum_j \sum_k h[j, k]f[m - j, n - k] \quad (1)$$

where input image is denoted by f , filter by h , (m, n) is the dimension of the output matrix. The pooling layer which gets its input from the convolution layer reduces the size of the feature map without any loss of information. The 2-dimensional arrays are modified to a single dimensional vector by flatten layer and then fed to the neural network for classification. The back-propagation algorithm is the logic employed in the neural network where the back propagation of errors will adjust the weights, thereby reducing the error (loss) function with weight updation as in the equation (2)

$$W_i = W_i + \Delta W_i \quad (2)$$

where W_i is the initial weight and ΔW_i is given by the delta rule as in (3).

$$\Delta W_i = n \frac{dE}{dw_i} x_i \quad (3)$$

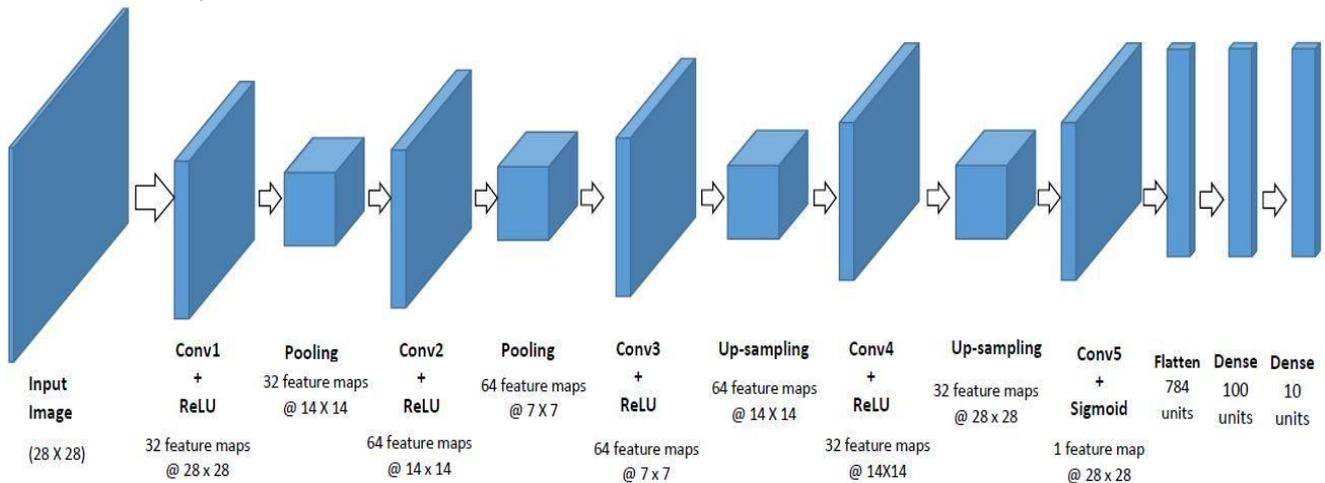


Fig. 1. The model for denoising image datasets

with learning rate n , error function E and input x_i .

A. Prototype developed

Fig. 1 depicts the architecture of the proposed model. The model has been designed with five convolution layers and with dropouts after each layer. The dataset of original images is resized to image of dimension 28 x 28 pixels. This becomes the input layer. The application of a series of two convolution operations to the image extract the features. An activation function, Rectified Linear Unit (ReLU) is applied to the first convolution layer (C1) which uses 32 kernels of size 3 x 3 to generate the feature maps.

To reduce the feature maps to a size of 14 x 14, subsampling is done by applying a Max-pooling (P1) layer of window size 2 x 2. In the second convolution layer (C2) 64 kernels of size 3 x 3 and ReLU activation is employed for generation of feature maps. To reduce the feature maps to a size of 7 x 7, subsampling is done by applying a Max-pooling (P1) layer of window size 2 x 2. This yields 64 feature maps. The procedure is followed with 64 kernels of size 3 x 3 and ReLU activation and this forms the third convolution layer (C3).

A window size of 2 by 2 Upsampling(U1) yields 64 feature maps. The dimension of the Feature map is 14 by 14. Procedure with 32 kernels of size 3 by 3 and ReLU activation forms the fourth convolution layer (C4). The window size of 2 by 2, Upsampling (U2) yields 32 feature maps of size 28 by 28. The procedure with 1 kernel of size 3 by 3 and Sigmoid activation is what constitutes the fifth convolution layer (C5). The flattened(F) feature maps result in a single dimension feature vector with 784 elements. This is converted to one dimension feature vector of size 100, by application of the first Dense layer with activation function of ReLU. The softmax activation forms part of the second dense layer (D2) and yields probabilistic output corresponding to the ten classes of digits.

The snapshot of the Keras DL Library built proposed system is as shown in Fig. 2.

```
Model: "sequential"
Layer (type)                Output Shape                Param #
-----
conv2d (Conv2D)              (None, 28, 28, 32)         320
max_pooling2d (MaxPooling2D) (None, 14, 14, 32)         0
conv2d_1 (Conv2D)            (None, 14, 14, 64)         18496
max_pooling2d_1 (MaxPooling2 (None, 7, 7, 64)         0
conv2d_2 (Conv2D)            (None, 7, 7, 64)           36928
up_sampling2d (UpSampling2D) (None, 14, 14, 64)         0
conv2d_3 (Conv2D)            (None, 14, 14, 32)         18464
up_sampling2d_1 (UpSampling2 (None, 28, 28, 32)         0
conv2d_4 (Conv2D)            (None, 28, 28, 1)          289
flatten (Flatten)            (None, 784)                 0
dense (Dense)                (None, 100)                 78500
dense_1 (Dense)              (None, 10)                  1010
-----
Total params: 154,007
Trainable params: 154,007
Non-trainable params: 0
```

Fig. 2. CNN model built using Keras

IV. DATA USED FOR EVALUATION

MNIST handwritten dataset with a total size of 70,000 images has been used for training, validation and testing. Biasing has been avoided by using equal number of different classes of images for training. Network performance has been validated along with the training phase so that proper learning is ensured. Out of the 70,000 samples in the database, 60,000 samples were considered for training, while 10,000 for testing as well as validation as shown in TABLE I. To reduce computational complexity without loss of relevant information, the images have been resized to 64 x 64.

Table- I: Dataset Considered

Classes under consideration	10 Classes
Training images under consideration	60000 Nos
Validation images under consideration	10000 Nos
Testing images under consideration	10000 Nos

V. METHODOLOGY

The programming language used for developing the model is Python, while coding and analysis has been executed using Jupyter Notebook environment. Keras, Scikit-learn, Matplotlib, and Numpy are some of the libraries and packages used. Open source deep learning framework for python, Keras, used to build the sequential network model and to add the convolutional layers of the model. Keras runs on top of open source machine libraries like TensorFlow, Theano or Cognitive Toolkit (CNTK). It is designed to quickly define deep learning models and is an optimal choice for deep learning applications. The most useful and robust library for Python based machine learning is Scikit-learn (Sklearn). Selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistent interface in Python can be achieved through the library. While the confusion matrix is plotted using the Sklearn library and the accuracy and loss plot are plotted using the matplotlib, which is a Python 2D plotting library.

VI. RESULT AND DISCUSSION

The optimizers Adam and SGD as well as utilization of the categorical cross-entropy as the loss function is used for training the noisy image dataset in the CNN. Three different types of noise viz Poisson, Salt and Pepper and Gaussian Noise have been used for testing the models. The following parameters has been finalised as the model parameters as shown in Table II.

Table- II: Parameters used for The Model

Model Parameters	Values
Images under consideration	70,000 Nos
Function of Activation	ReLU, Sigmoid and Softmax
Rate of Learning	0.01
Epochs used	10
Optimization Function	Adam, SGD
Loss function	Categorical Cross-entropy

The classical stochastic gradient descent procedure has been replaced by Adam the optimization algorithm to update network weights iteratively based on training data.

In the Stochastic Gradient Descent(SGD) method, a few samples are selected randomly instead of the whole data set for each iteration. Each iteration is performed using only a single sample, i.e., a batch size of one. The randomly shuffled sample is selected for performing the iteration. The selection of only one sample from the dataset and chosen at random for each iteration, the path taken by the algorithm to reach the minima is usually noisier than your typical Gradient Descent algorithm in SGD. Since path followed is immaterial as long as the minima is reached with significantly shorter training time, SGD is acceptable. The confusion matrix for the six test cases for the 2 models using the proposed CNN model is depicted in Fig. 3(a) to 3(f). The training accuracy and training loss of the model 1(Adam optimizer) respectively for added Gaussian noise for the entire epochs is depicted in Fig. 4 (a) and 4 (b). Fig. 4 (c) and 4(d) shows the images for model 2 (SGD optimizer). Analyzing the plots, it is obvious that the models are not overfitting.

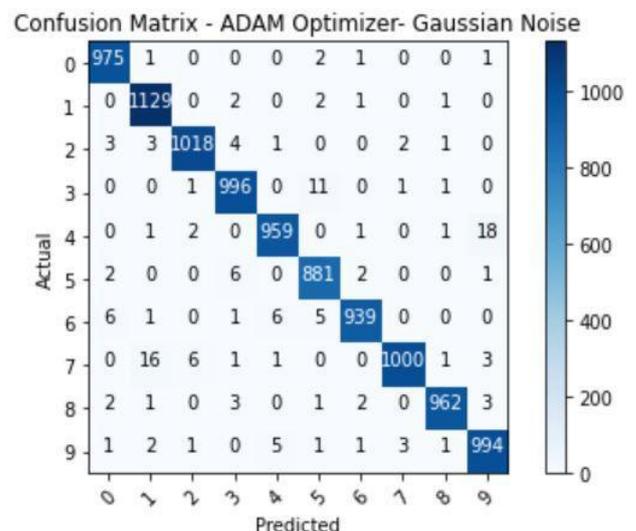


Fig. 3(a)

Confusion Matrix - ADAM Optimizer- Salt & Pepper Noise

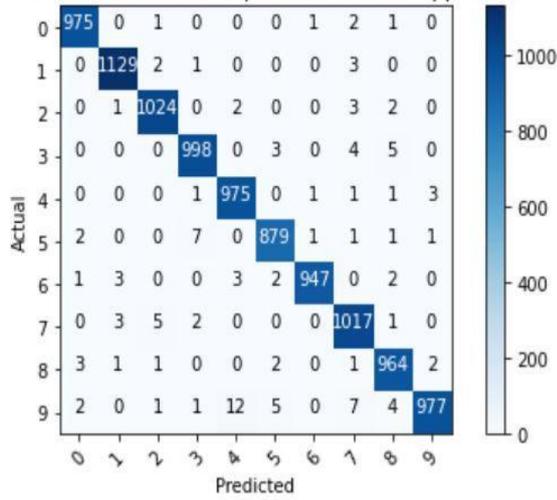


Fig. 3(b)

Confusion Matrix - SGD Optimizer- Salt & Pepper Noise

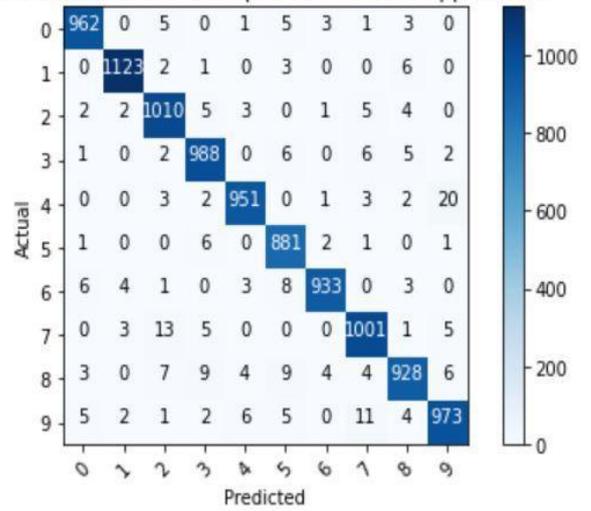


Fig. 3(e)

Confusion Matrix - ADAM Optimizer- Poisson Noise

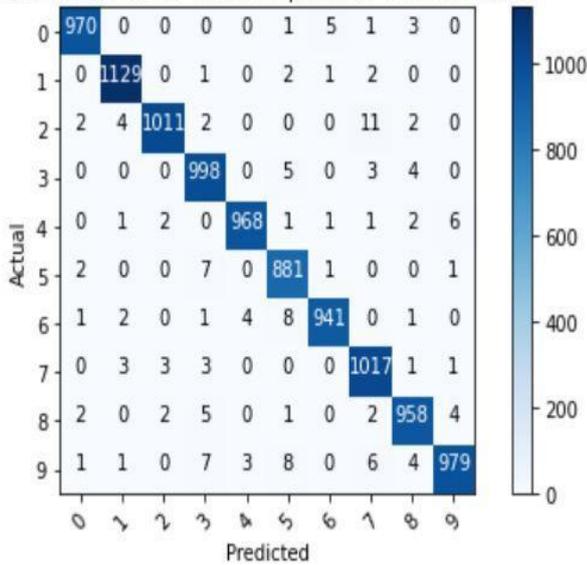


Fig. 3(c)

Confusion Matrix - SGD Optimizer- Poisson Noise

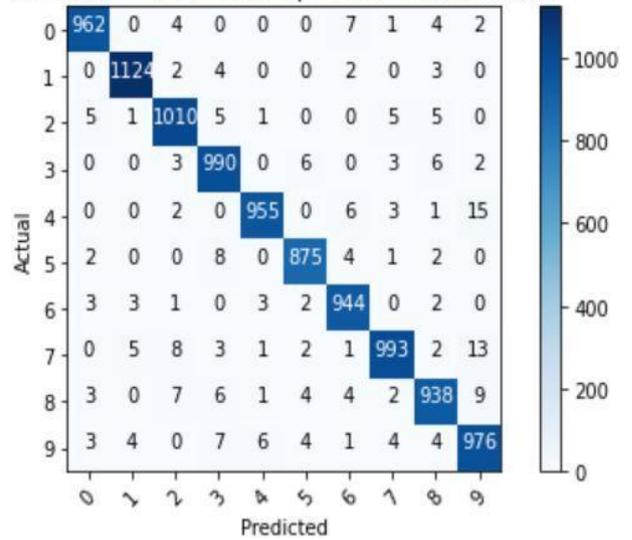


Fig. 3(f)

Confusion Matrix - SGD Optimizer- Gaussian Noise

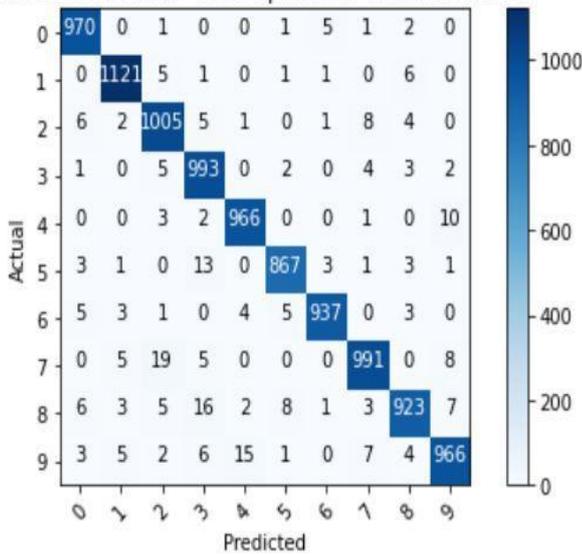


Fig. 3(d)

Model Accuracy : ADAM Optimizer-Gaussian Noise

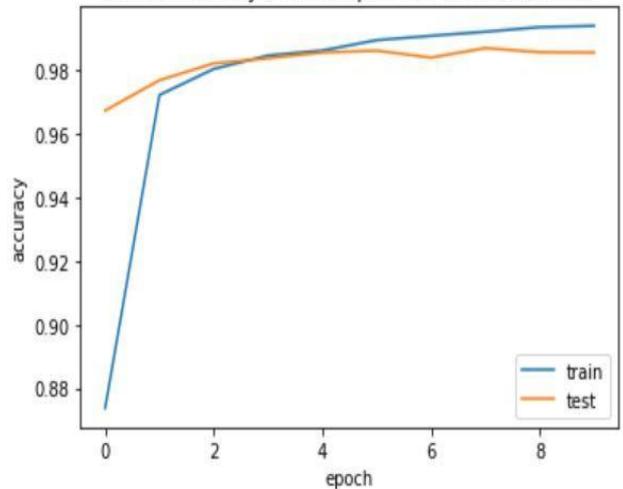


Fig. 4(a)



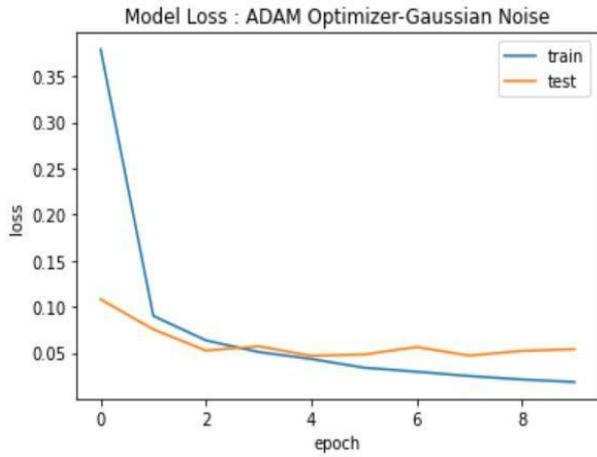


Fig. 4(b)

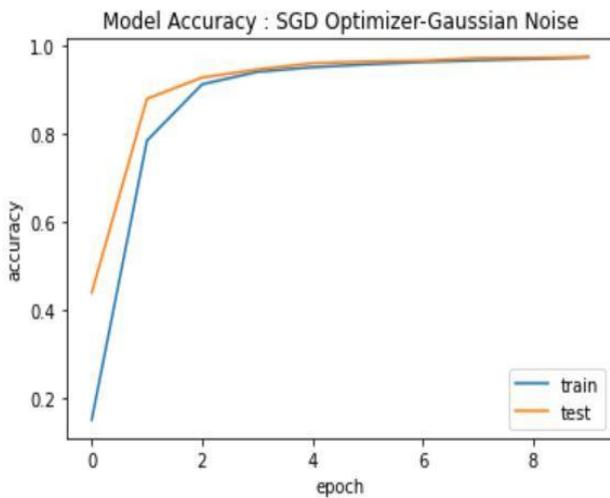


Fig. 4(c)

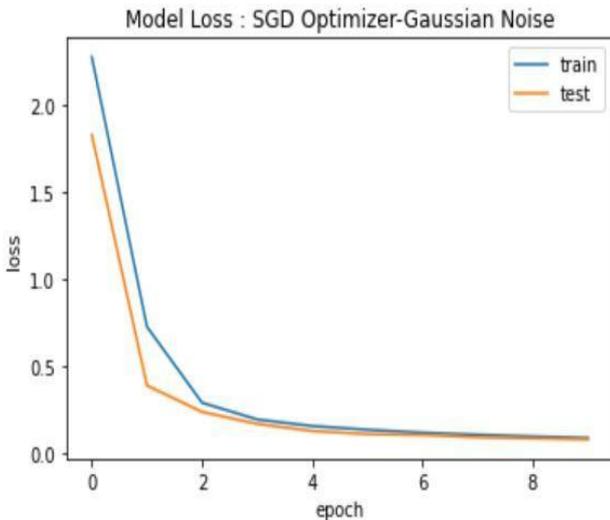


Fig. 4(d)

Fig. 5(a) shows a sample training image dataset without adding noise. Fig. 5(b) shows the training image which is added with Gaussian Noise. Fig. 5(c) shows the predicted output using the proposed Generalized CNN model. From the Fig. it is clear that the noise is eliminated completely.

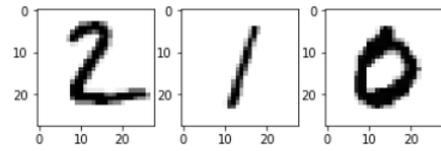


Fig. 5 (a)

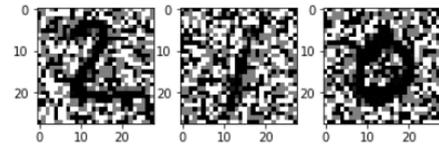


Fig. 5 (b)

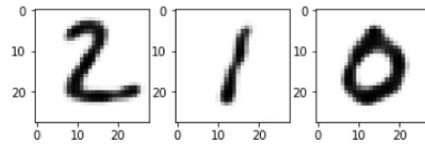


Fig. 5 (c)

Table- III: Prediction Accuracy for all Models

Noise	Model 1 (Adam Optimizer)	Model 2 (SGD Optimizer)
Gaussian Noise	0.9739	0.9853
Salt and Pepper Noise	0.9750	0.9884
Poisson Noise	0.9767	0.01

VII. CONCLUSION

This work proposes a generalized Deep Convolutional Neural Network model to classify and recognize any type of noisy images. Two models were compared, the first model uses an Adam optimizer and the second one uses SGD optimizer. Three types of noise viz. Poisson Noise, Gaussian Noise, Salt and Pepper Noise and were added to the input image and tested for the two models. From the training, test and validation results model 2, with SGD optimizer gives more accuracy when compared to the model Adam optimizer. Future work includes the improvement of the same model with high rate noise. Denoising video dataset using 3DCNN or LSTM(Long Short Term Memory Units) are also enhancements. Also this system can be implemented for denoising images in real time applications in medical field, space and research, etc.

REFERENCES

1. O. Sheremet, K. Sheremet, O. Sadovoi and Y. Sokhina, "Convolutional neural networks for image denoising in infocommunication systems," 2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), Kharkiv, Ukraine, 2018, pp. 429-432, doi: 10.1109/INFOCOMMST.2018.8632109.
2. Q. Xiang and X. Pang, "Improved denoising auto-encoders for image denoising," 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Beijing, China, 2018, pp. 1-9, doi: 10.1109/CISP-BMEI.2018.8633143.
3. S. Suresh, H. T. P Mithun and M. H. Supriya, "Sign language recognition system using deep neural network," 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 2019, pp. 614-618. doi: 10.1109/ICACCS.2019.8728411
4. Z. Liu, W. Q. Yan and M. L. Yang, "Image denoising based on a CNN model," 2018 4th International Conference on Control, Automation and Robotics (ICCAR), Auckland, 2018, pp. 389-393, doi: 10.1109/ICCAR.2018.8384706.
5. E. Pranav, S. Kamal, C. Satheesh Chandran and M. H. Supriya, "Facial emotion recognition using deep convolutional neural network," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2020, pp. 317-320, doi: 10.1109/ICACCS48705.2020.9074302.
6. Ç. P. Dautov and M. S. Özerdem, "Wavelet transform and signal denoising using Wavelet method," 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, 2018, pp. 1-4, doi: 10.1109/SIU.2018.8404418

AUTHORS PROFILE



Kurian Thomas is an Electronics and Telecommunication engineer, with an M.Tech in Digital Electronics and MBA in International Marketing. He has over twenty years of experience in the field of telecommunication, embedded systems and software development. He has been a member of the IEEE society for over two decades and is on the director board of a number of technology companies and educational

institutions. He is currently pursuing research in the field of secure communication.



Pranav E received his master's degree in Electronics and Communication with a specialization in Wireless Technology from Cochin University of Science and Technology in April 2020. He received his bachelor's degree in Electronics and Communication from Kerala University in April 2016. He has published a paper in the area of Deep Learning in IEEE Xplore Digital Library in the year 2020. His research interests are mainly Natural Language Processing and Machine Learning..



Dr. Supriya M.H. is currently working as Professor and Head in the Department of Electronics, Cochin University of Science and Technology, India, Kerala. She obtained her Ph.D. from the same department in 2008. With her 22 years of teaching experience and 4 years of Industrial experience, she has received various awards in the field of research and teaching. Her research and teaching areas of interest include Signal Processing, Machine Learning ,

Sonar Technology, as well as Underwater Target Recognition. She has guided the current work.