Convolutional Autoencoder based Deep Learning Model for Identification of Red Palm Weevil Signals

Parvathy S. R., Deepak Jayan P., Nimmy Pathrose, Rajesh K. R.
Centre for Development of Advanced Computing
A Scientific Society of the Ministry of Communication and Information Technology, Govt. of India
Vellayambalam, Thiruvananthapuram, Kerala, India
parvathysr@cdac.in, deepakjp@cdac.in, nimmy@cdac.in, rajesh@cdac.in

Abstract— **This paper presents a Convolutional Autoencoder based Deep Learning model for identification of Red Palm Weevil acoustic emissions from other background noise. Mel spectrogram of acoustic samples was chosen as the extracted feature for the proposed model. The designed Convolutional Autoencoder was trained using Mel spectrogram images of Red Palm Weevil acoustic activities which are regarded as the normal instances. Unbiased evaluation of the model was done with a test dataset composed of normal RPW acoustic emissions as well as anomalous acoustic samples. The model could achieve a very high classification accuracy of 95.85%. The results confirmed that the proposed method is highly efficient for the identification of Red Palm Weevil signals.**

Keywords—Convolutional Autoencoder, Deep Learning , Mel spectrogram, Red Palm Weevil, feature extraction, Mean Squared Error

I. INTRODUCTION

Red Palm Weevil (RPW) is a pest species regarded as a fatal enemy of coconut and date palms, posing major threat and severe economic losses to palm cultivation worldwide. The insect lays eggs inside the palm and its entire life cycle is completed inside the tree. The larvae actively feed on the soft tissues inside the stem or crown causing severe damage to the tree. During early stages of infestation, there will not be any visible symptoms and is difficult to be traced from outside the tree. However, the feeding activity of the larvae causes feeble acoustic signals, which can be captured and processed to identify its presence.

Identifying the feeble acoustic signals generated by the larvae is challenging, amidst the movement of petioles in wind, fibre breaking and other background noise. With Deep Learning techniques maturing with more accurate algorithms, its possibilities can be harnessed to effectively detect infestation. The paper proposes a technique to identify RPW generated signals from other sounds, based on Convolutional Autoencoder (CAE), a popular Deep Learning architecture.

This paper is organized as follows. First, the theoretical background of the techniques involved in this work is explained. Next, the methodology adopted for the work is presented. Following that, the experiments carried out and the results arrived at, are discussed. Finally, the paper concludes with the inference from the experiments carried out.

II. THEORETICAL BACKGROUND

In this section, the common signal processing workflow is introduced in brief. Until a few years ago, audio applications used to rely on traditional digital signal processing techniques for feature extraction. This required a lot of domain-specific expertise to solve these problems and tune the system for better performance. However, in recent years, as Deep Learning becomes more and more prevalent, it has seen enormous success in handling audio data. Most machine learning models do not directly operate on the raw audio signals. Instead, the common approach used is to convert the audio data into images for feature extraction, and then use any image processing architecture to process those images.

A. Mel Spectrograms

Recently, what has been prominent is that, the raw audio is first transformed from the time domain to the frequency domain, exploiting the fact that arbitrarily complex audio signals can be represented as a combinations of simple sinusoids. In practice, this is done using the Short-Time-Fourier Transform (STFT). The STFT applies the discrete Fourier transform on small overlapping blocks of the raw audio to account for signals whose frequency characteristics change over time. The output of the STFT is a matrix of dimension F×T with F frequency bins and T time frames and is called the spectrogram.

Humans do not perceive frequencies linearly. Most of what humans hear are concentrated in a narrow range of frequencies and amplitudes.

To account for the fact that the human perception of frequencies are logarithmic in nature, i.e. more discriminative at lower frequencies and less discriminative at higher frequencies, one further transforms the frequency bins of a spectrogram into the Mel-scale using the Mel-filter bank that is composed of overlapping triangular filters [1]. The formula for converting frequency f to Mel scale is:

$$m=2595 \log_{10}\left(1+\frac{f}{700}\right) \quad (1)$$

The Mel spectrogram is used to provide the models with realistic sound information similar to what a human would perceive. Mel filter bandwidth is small for low frequencies and increase in width for higher frequencies as shown in Fig. 1.

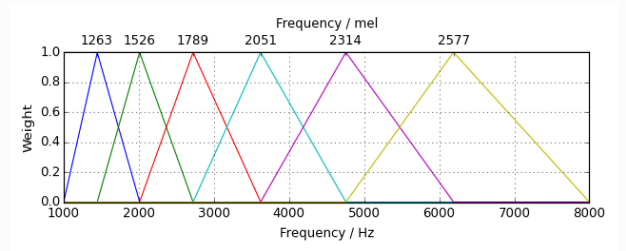

Fig. 1 Filter banks on a Mel scale

The filters combine the energy of consecutive frequency bins to describe how much energy exists in various frequency regions. Finally the log of the energies is taken to convert from amplitude to decibel (dB). The resulting Mel-spectrogram is a compact visual representation of the audio that can be used as an input to a machine-learning pipeline. Moreover, the Mel spectrogram can be treated as an image of the underlying signal and one can therefore utilize computer vision approaches for its processing.

B. Convolutional Autoencoder

As an unsupervised learning method, autoencoder (AE) is designed to extract useful features from unlabeled data, to remove input redundancies [2] and to carry out dimensional data reduction. An autoencoder consists of two parts: encoder and decoder as illustrated in Fig. 2.

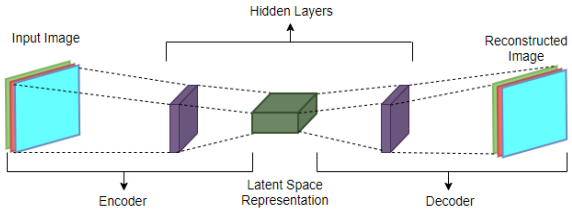

Fig. 2 The architecture of an autoencoder

The encoding function f with several hidden layers, will encode the input data ' x_i ' to a compressed domain representation.

$$y = f(W * x_i + b) \tag{2}$$

where, W is weights between input x_i and latent space representation y, and b is the bias. The hidden layer nodes learn the specific attributes of the input data.

The decoder function f' will try to reconstruct the input ' x_i ' from that compressed representation, which can be expressed as:

$$\hat{x}_i = f'(W' * y + b') \tag{3}$$

where, W' is the weights between latent space representation y and reconstructed output $\hat{x}_i$, b' is the bias.

The expectation from an autoencoder is twofold. Firstly, the autoencoder should be sensitive enough to the input for accurate reconstruction. The other expectation from an autoencoder is, it should be insensitive enough so that it does not memorize the input data. These conflicting requirements are decided actually by designing cost function.

The principle of training is to minimize the reconstruction error, which can be realized by minimizing the following cost function J_{AE} given as:

$$J_{AE} = 1/p \sum_{i=1}^{p} L(x_i, \hat{x}_i) \tag{4}$$

Where p is the number of input images, x_i is the i-th input image and $\hat{x}_i$ is the reconstructed image corresponding to x_i. $L(x_i, \hat{x}_i)$ represents the reconstruction error of the input image, which can be measured by mean squared error (MSE) or binary cross entropy. In this study, the MSE between the input image x_i (i=1,2,…p) and the reconstructed image $\hat{x}_i$ (i=1,2,…p) is used. Correspondingly, $L(x_i, \hat{x}_i)$ can be expressed as:

$$L(x_i, \hat{x}_i) = \| x_i - \hat{x}_i \|^2 \tag{5}$$

Convolutional Autoencoder (CAE) combines the local convolution connection with the autoencoder, which is a simple step that performs convolution operation to the inputs. The Convolutional layers are used to extract features from input images. Correspondingly, a convolutional autoencoder consists of convolutional encoder and convolutional decoder. The convolutional encoder, which includes convolutional filters and subsequent pooling operations, realizes the process of conversion from the input to the feature maps, while the convolutional decoder implements the conversion from feature maps to outputs. CAEs are more advantageous since it requires smaller memory because of the concept of parameter sharing [3].

III. METHODOLOGY

The proposed CAE based architecture regards RPW acoustic emissions as normal instances and all other acoustic samples other than RPW emissions as anomalous samples. This method is basically formulated on the idea that, when an anomaly occurs, the reconstructed images will be quite different from that of normal instances, which has been used for training an autoencoder model. The method involves three basic steps: training and generation of autoencoder model, determination of reconstruction error threshold and finally, testing the accuracy of the generated model as illustrated in Fig. 3, Fig. 4 and Fig. 5 respectively. The training is done with extracted Mel spectrogram input images of RPW signals. After training the model, it is expected that the reconstruction error of normal events must be lower than that of the abnormal events. Based on the statistical parameters of the reconstruction error distribution, an optimal anomaly detection threshold is computed [5]. Then this thresholding is applied to test images to classify it as normal or anomaly instance.

A. *Training and generation of Autoencoder model*

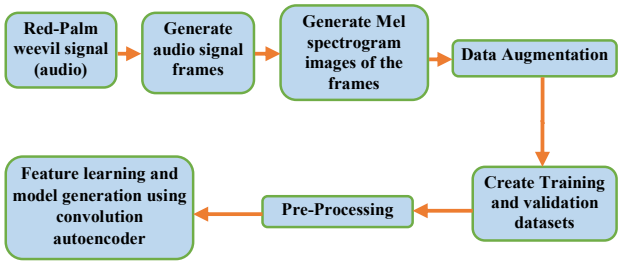

Fig. 3 Training and generation of autoencoder model

The RPW signal captured using data acquisition system was segmented into frames of 1 second duration. These frames were used to generate Mel spectrogram images. For training a model, ample amount of data was required. Data augmentation method was adopted to generate sufficient number of training and validation datasets. Both the training and validation datasets were subjected to pre-processing, which resizes the images and scales the pixel values. A series of Convolution, max pooling and up-sampling operations were used for learning the image features to develop the model.

B. *Determination of reconstruction error threshold for normal data*

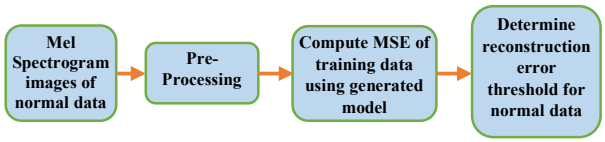

Fig. 4 Determination of reconstruction error threshold

The reconstruction error threshold for classifying acoustic samples as normal or anomalous ones is arrived at, by analyzing the statistical distribution of reconstruction errors with the training dataset. The MSE of these images were computed by comparing them to the output images reconstructed by the model, and from the relative frequency distribution of MSEs a suitable threshold was identified. Based on the statistical properties of this distribution curve, reconstruction error threshold for normal data was calculated.

C. *Testing the accuracy of generated model*

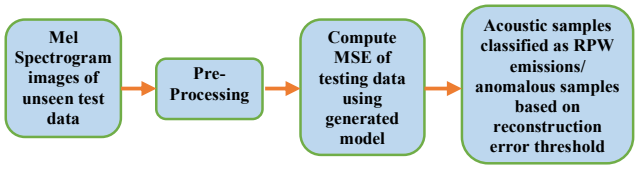

Fig. 5 Testing the accuracy of generated model

Accuracy of the generated model is then analyzed based on an unbiased evaluation with a test dataset composed of normal and anomalous acoustic samples. Using the model, the MSE of the test data images were computed. Based on the reconstruction error threshold, the test images were classified as normal data instances or anomalies.

IV. Experimental Study

A. *Dataset Preparation*

Deep learning algorithm learns from data. Dataset preparation is regarded as the most crucial stage in developing any DL algorithm. It is critical that right data in useful scale and format with meaningful features specific to the problem is prepared beforehand, to achieve consistent and better results. As with other deep learning applications, extensive datasets for conducting experimental study of red palm weevil acoustic activity as such is not available in public domain. This challenge was overcome by collecting and aggregating audio samples of weevil activity from infested palms with a custom-made data acquisition system. Suitable feature extraction techniques were then used to identify key features in data. Standard data augmentation methodologies were adopted to expand the dataset.

1. Data Collection

The effective frequency range of red palm weevil acoustic emissions lies in the human audible range of 800 - 4000Hz. A custom-made portable Data Acquisition System (DAQ) was developed in-house.

For training the autoencoder model, only a labeled set of normal (weevil acoustic) samples were needed. The performance of model was then evaluated on labeled set of RPW activities and an unlabeled set of anomalous samples which includes sounds due to petiole movements in wind, birds and animal sounds, human talks, machine sounds etc, which were also acquired using the custom-made DAQ.

2. Feature Extraction

As mentioned earlier, Mel Spectrograms are the most widely used feature extraction technique for deep learning applications using audio data. This method is particularly popular as it approaches the audio classification problem as a 2D image classification problem.

The audio signal acquired from infested palms using the custom-made DAQ at a very high sampling rate of 50 KHz, was segmented into frames of 1 second duration, around the occurrence of RPW acoustic activities. Mel Spectrograms of such segments were generated, making use of Python package 'Librosa'. Spectrograms were generated with a window length of 2048 samples (approx. 40 ms duration) and a hop length of 512 samples, with 196 Mel filter banks. Resulting spectrograms were saved as 196x196 images with 24 bit depth in RGB format.

Similar to preparing training and test normal datasets from infested palms, the test anomaly dataset for evaluating model was generated with audio data collected from healthy palms as shown in Fig. 6.

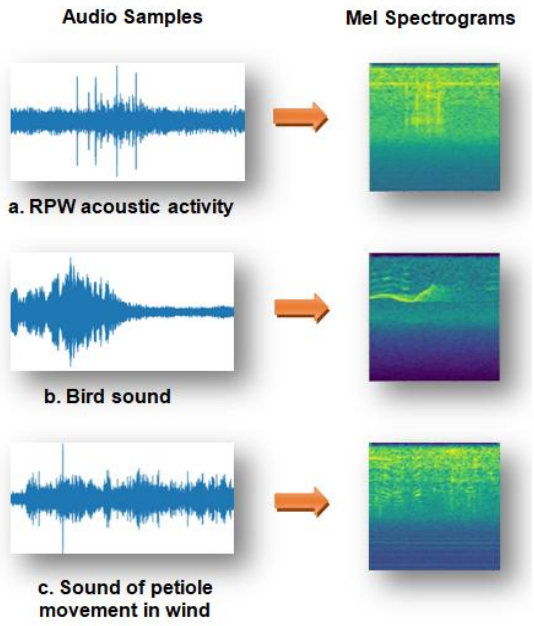

Fig. 6 Mel Spectrograms of normal and anomalous samples

3. Data Augmentation

The performance of any deep learning model depends on quality and quantity of data. In a broad sense all the deep learning algorithms can be considered data hungry. Data augmentation methodologies expand and create variations in the dataset thus improving model's generalization ability. It also prevents chances of overfitting, while training the model.

Two approaches were adopted for augmenting the training dataset:

1. Time shift - Generated Mel Spectrograms from audio data shifted left or right in time by random amounts.
2. Noise addition – Generated Mel Spectrograms from audio data after addition of some random noise.

B. Model Architecture Design

The input images to the proposed model are 3 layer RGB Mel Spectrogram images of dimension 196x196. The CAE model consists of Encoder-Decoder structure, mainly consisting of 2D Convolution layers with multiple filters aimed at extracting features from the input images. The proposed model was implemented with 3 such convolution layers as illustrated in Fig. 7. Several experiments were conducted by varying the number of filters from [8, 16, 32] to [32, 64, 128] with filter kernel sizes from (3x3) to (5x5). The effect of adding Batch Normalization after each convolution layer to reduce the number of training epochs, was also analyzed. Experiments were also done with addition of Fully Connected layers between the encoder and decoder, aiming to train the model with non-linear feature combinations. Model performance with dropout and early stopping regularization techniques were also assessed.

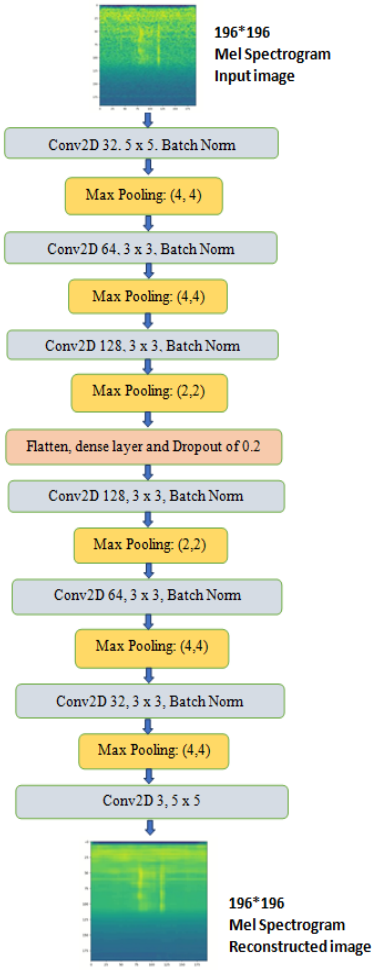

Fig. 7 Convolutional Autoencoder model Architecture

Based on the results of the above conducted experiments, an optimum CAE model architecture was arrived at, as shown in Fig 7. The model was then compiled with Adam optimizer, with a default learning rate of 0.001, with 'Mean Squared Error' loss function and with a batch size of 32 for 100 epochs.

V. RESULTS

The model was trained using an ample training dataset composed of Mel Spectrogram images of RPW acoustic emissions, which are regarded as normal samples. Since the model was trained on such samples alone, it better reconstructed Mel Spectrograms of RPW activities compared to any other acoustic samples. The model outputs were compared in terms of the MSE between input and reconstructed image. The more similar the reconstruction, the smaller was the reconstruction error.

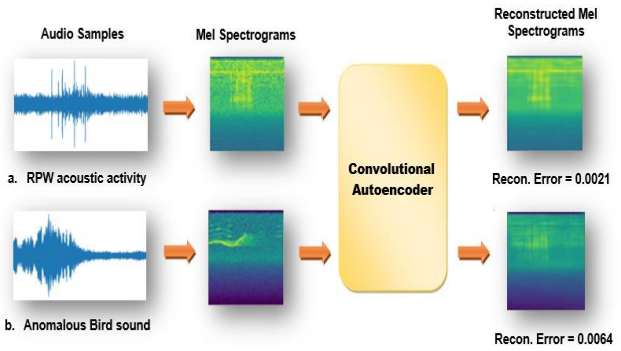

Fig. 8 Reconstruction of normal and anomalous samples using CAE model

From Fig. 8, it is clear that the Mel Spectrogram of RPW activity has been faithfully reproduced with a small reconstruction error, whereas that of anomalous bird sound was poorly reproduced with a noticeably high reconstruction error.

Reconstruction error threshold to discriminate between the normal and anomalous acoustic activities was arrived at by analyzing the distribution of reconstruction errors of the normal training dataset.

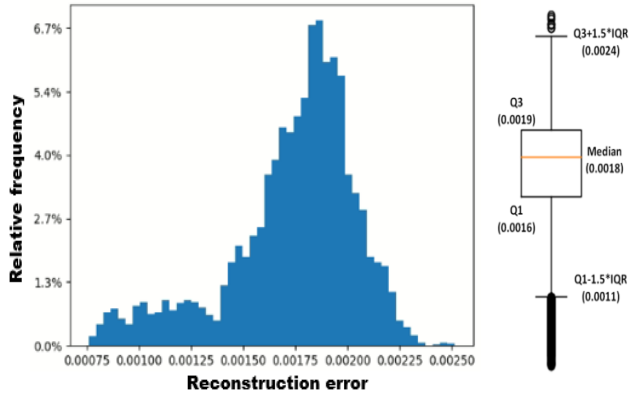

Fig. 9 Histogram and Box-plot of reconstruction error distributions of normal class

Because of skewed distribution of errors in the normal dataset, a reasonable threshold was initially chosen based on the five-number statistics, specified in the box-plot, which defines the outlier limit in the distribution as

$$\text{Outlier limit} = Q3 + 1.5 * \text{IQR} \quad (6)$$

An unbiased evaluation of final model was then performed on a test dataset composed of Mel Spectrogram images of both normal and anomalous samples. From the histograms of test data reconstruction errors, it is evident that the above chosen threshold gives satisfactory results.

To arrive at an optimum threshold the performance of the model was then evaluated on different thresholds nearer to this limit based on two statistical analysis measures [6].

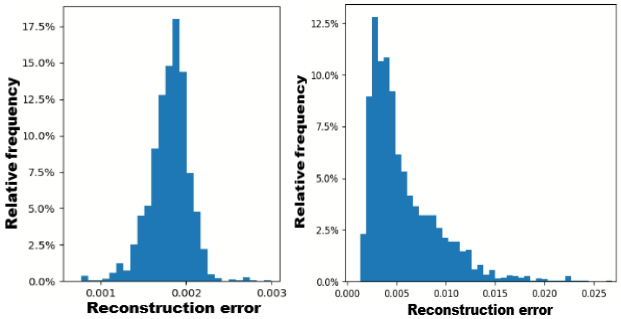

Fig. 10 Histogram of reconstruction errors of test dataset

1. F1-Score based on confusion matrix

The commonly used measures for evaluating model performance based on confusion matrix are Accuracy, Precision and Recall. Accuracy shows the number of correct predictions out of the total predictions made for the dataset. Sensitivity/Recall indicates how sensitive the model is in detecting true positives, which in our case is the normal RPW acoustic activity. This measure is important, since the cost associated with missing positive instances is too high as it leads to missing identification of infested palms. Precision on the other hand, gives the measure of correctly predicted positive samples. This measure is equally important since cost associated with detecting anomalies as RPW activities leads to unnecessary application of pesticides or cutting down of healthy palms.

F1-Score provides a way to combine both precision and recall into a single measure that captures both properties. Higher the F1-Score better the model performance. It is computed as

$$\text{F1-Score} = 2 * \text{Precision} * \text{Recall} / (\text{Precision} + \text{Recall}) \quad (7)$$

Accuracy, Precision, Recall, and F1-Score computed for different reconstruction error thresholds closer to the outlier limit is tabulated in Table 1.

TABLE 1. ACCURACY, PRECISION, RECALL, AND F1-SCORE AT DIFFERENT RECONSTRUCTION ERROR THRESHOLDS

Reconstruction Error Threshold	Accuracy %	Precision %	Recall %	F1-Score %
0.001940908	85.50	96.59	73.60	83.54
0.002109908	94.50	95.27	93.65	94.45
0.002278908	95.85	93.17	98.95	95.97
0.002309619	95.73	92.83	99.10	95.86
0.002447908	94.60	90.66	99.45	94.85
0.002616908	92.93	87.90	99.55	93.36

From the table, it is clear that the model better performs at a reconstruction error threshold of 0.002278 with very high F1-Score of 95.97%. The results also show that the compiled model in general shows a very high accuracy of 95.85%.

2. ROC curve and AUC value

The Receiver Operating Characteristic (ROC) curve in Fig. 11 shows the performance of model at all classification thresholds. It gives a trade-off between TPR and FPR for the model at different chosen thresholds. Area under ROC curve AUC, is a performance measure for the model which should be ideally 1.

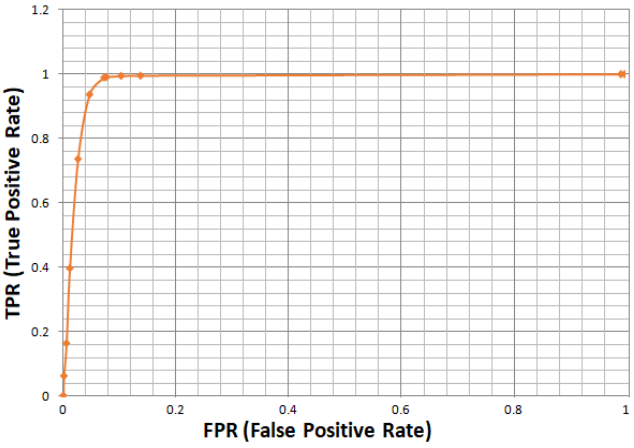

Fig. 11 ROC Curve of CAE model

From the ROC curve, it can be observed that the compiled model shows a very high AUC value of 0.8979.

VI. CONCLUSIONS

The work describes the application of CAE as the Deep learning technique for identifying RPW signals. Several experiments were done in tuning the model parameters so as to arrive at an optimum design of CAE for accurate results.

The results show that, the approach using CAE is highly robust to identify RPW signals from the background signals. The model could achieve a very high classification accuracy with an F1 score of 95.97% and AUC value of 0.8979. The technique provides a promising method to address the RPW identification problem.

ACKNOWLEDGMENT

The authors would like to express their appreciation for the financial support by, Department of Science and Technology, Govt. of India.

REFERENCES

[1] Robert Muller, Steffen Illium, Fabian Ritz and Kyrill Schmid, “Analysis of Feature Representations for Anomalous Sound Detection, ” in *ICAART 2021 - 13th International Conference on Agents and Artificial Intelligence*.

[2] Borui Hou, Ruqiang Yan, “ Convolutional Autoencoder Based Deep Feature Learning for Finger-Vein Verification, ” in *2018 IEEE International Symposium on Medical Measurements and Applications (MeMeA),*11-13 June 2018.

[3] Muyuan Ke, Chunyi Lin, Qinghua Huang, “Anomaly detection of Logo images in the mobile phone using convolutional autoencoder,” in *2017 4th International Conference on Systems and Informatics (ICSAI),* 11-13 Nov. 2017.

[4] Ragini Sinha, Padmanabhan Rajan,“ A Deep Autoencoder Approach To Bird Call Enhancement, ” in *2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS),*1-2 Dec. 2018.

[5] Hemant Dhole, Mukul Suta.one, Vibha Vyas,“ Anomaly Detection using Convolutional Spatiotemporal Autoencoder, ” in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 6-8 July 2019.

[6] Andrea Borghesi, Andrea Bartolini, Michele Lombardi, Michela Milano, Luca Benini, “Anomaly Detection using Autoencoders in High Performance Computing Systems, ” in *Proceedings of the AAAI Conference on Artificial Intelligence,* Vol. 33, pages 9428-9433, 2019.