International Journal of Business Data Communications and Networking
Volume 17 • Issue 2

Outlier Detection Using Convolutional Neural Network for Wireless Sensor Network

Biswaranjan Sarangi, Biju Patnaik University of Technology, India*

Arunanshu Mahapatro, Veer Surendra Sai University of Technology, India

https://orcid.org/0000-0002-4171-3926

Biswajit Tripathy, GITA Autonomous College, Bhubaneswar, India

ABSTRACT

Over the recent years, deep learning has been considered as one of the primary choices for handling huge amounts of data. Having deeper hidden layers, it surpasses classical methods for detection of outliers in wireless sensor networks. The convolutional neural network (CNN) is a biologically-inspired computational model which is one of the most popular deep learning approaches. It comprises neurons that self-optimize through learning. EEG generally known as electroencephalography is a tool used for investigation of brain function, and EEG signal gives time-series data as output. In this paper, the authors propose a state-of-the-art technique designed by processing the time-series data generated by the sensor nodes stored in a large dataset into discrete one-second frames, and these frames are projected onto 2D map images. A convolutional neural network (CNN) is then trained to classify these frames. The result improves detection accuracy.

KEYWORDS

Azimuthal Projection, CNN, EEG Classification, FFT, Frequency Binning, Hanning Window, Outlier, WSN

1. INTRODUCTION

Outlier detection of wireless sensor network (WSN) has been an active research area in critical real-life application scenarios like remote patient health monitoring, environmental monitoring, engineering structures, industrial process monitoring, fraud detection, target tracking and military operations. Sensor nodes that are wirelessly interconnected are densely deployed across a geographical area, collecting sensed data and sending it to a central server or sink.The quality of data collected is sometimes unreliable and inaccurate due to the imperfect nature of WSNs, such as low battery power, low memory, and low communication bandwidth (Wang et al., 2006). In the context of WSNs outlier also known as anomaly is defined by Hawkins as “An outlier is an observation that deviates so much

DOI: 10.4018/IJBDCN.286705

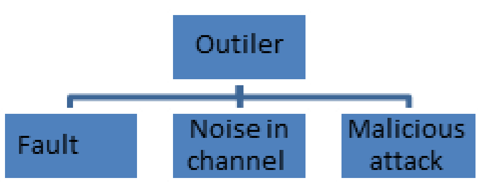
*Corresponding Author

from other observations as to arouse suspicion that it was generated by a different mechanism" (D.M. Hawkins, 1980). According to Barnett and Lewis, outlier is defined as "An outlier is an observation or subset of observations that appears to be inconsistent with the rest of the set of data" (Barnett and Lewis, 1994). Sadik, S. et al defined outlier as "An outlier is a data point which is significantly different from other data points, or does not conform to the expected normal behavior, or conforms well to a defined abnormal behavior" (Zhang et al., 2010). Outliers have an impact on the quality of information obtained from WSNs, which are classified as local or global depending on the data. In data dimension scenario the data streams can be univariate having single attribute or multivariate having multiple attributes. Outliers in data occur when there are any deviations in the sensed data that are correlated in both time and space.Temporal correlation implies temporal anomaly due to changes in data over time at a single node location. Spatial correlation denotes spatial anomaly caused by comparison with neighboring nodes, whereas spatiotemporal anomaly is caused by changes in data value over both time and space at a greater number of node locations.

The various sources of outliers are shown in Fig 1. As the nodes are deployed in harsh and hostile environment, faults in WSNs are likely to occur unexpectedly and frequently, ranging from simple permanent faults to the faults where the node behaves maliciously (Mahapatro and Khilar 2013). Faults are likely to occur unexpectedly due to fault in hardware or programs where a node becomes inactive gives erroneous outputs. Noise or error occurs from a noise-related measurement of a faulty. Malicious attacks can be passive attack where the data changed in the network without interrupting data communication or active attack which aims to minimize the functionality of the network by injecting false and corrupted data (Bhushan and Sahoo, 2018). This relates to the network security (Puri and Bhushan, 2019). Because of the low computing power, high-end security solutions cannot be implemented, making the nodes more vulnerable to security threats.Outliers are measurements taken by defective sensor nodes or nodes that have been hacked that vary considerably from the normal pattern of sensed data.Unattended outliers can have hazardous consequences in terms of environmental damage, human life, and economic hardship.Because a substantial number of WSNs will be used in safety-critical applications, outliers may cause misunderstanding or undesired alerts, resulting in life-threatening incidents.So the detection of outlier provides data reliability, effective and secures functioning of the network. Identifying outlier sources, node resource constraints, reducing communication overhead, routing (Bhushan, Sahoo, 2019), processing large amounts of distributed data online due to dynamic change behavior, and frequent communication failure between nodes due to large scale deployment in an unattended environment are major challenges in any outlier detection technique (Mahapatro and Khilar 2011).

To the best of our knowledge this paper presents a newly attempted methodology to detect outlier particularly in wireless sensor network, based on an unsupervised learning called CNN using EEG classification (Bashivan et al., 2016). Initially the data were separated into overlapping one-second frames. Then Hanning window is applied. Next, to transform data for each frame from time domain to frequency domain, First Fourier transform (FFT) is applied. Frequency binning is then used where the FFT amplitudes are classified into theta, alpha and beta ranges which gives 3 scalar values for each sensor per frame. These three values are converted as RGB color channel onto a 2D map by using 2D Azimuthal Projection. Then a convolutional neural network (CNN) is trained to classify the frames.

Figure 1. Outlier sources in WSNs (Zhang et al., 2010)



1.1 Motivation

Since the previous few decades, researchers have recognized outlier detection as a significant challenge in wired interconnected networks.Although the fundamentals of outlier detection are well established, their application to specific domains, particularly in WSNs, has received little attention. WSNs have the inherent ability to deploy low-cost sensor nodes in large numbers in uncontrolled or hostile environments.Sensor nodes are prone to becoming defective and unreliable.The regular operation of a WSN suffers from outliers because it reduces the base station's judgment accuracy, increases traffic in the WSNs, and wastes a lot of limited energy.Most of the current outlier detection techniques are not suitable in real time. There is little research on the real-time identification of outliers. After successfully detecting outliers in real-time data, outlier data can be prevented from entering the network, preventing unnecessary participation of the relay nodes in the transmission to the sink node and extending the network's lifetime.These issues motivate the need to design real-time outlier detection algorithms to address the aforementioned problems.

1.2 Objectives

The objectives of this work are as follows:

- To detect outlier in WSN by obtaining data from standard publicly available datasets such as the sensor-scope-Lausanne Urban Canopy Experiment (LUCE).
- To uses a novel CNN to detect the outliers in real-time.
- To validate the suitability of the proposed outlier detection model by comparing over the existing models.

In this paper we look at definition of outliers in WSN and its desirable properties. The attributes of input sensor data, correlations, type of outliers and identity of outliers including the major challenges of outlier detection techniques are briefed. The rest of paper is organized as follows: Section 2 describes the work related to various classified outlier detection techniques; In Section 3, we propose a CNN model for outlier detection; Experimental results are discussed in section 4; and concluding remarks given in last section.

2. RELATED WORK

The literature survey on outlier detection techniques in WSN is presented in (Zhang et al., 2010) and (AyaAyadi et al., 2017) where the outlier detection techniques has been classified into statistical-based, nearest neighbor-based, clustering-based, classification-based, and spectral decomposition-based approaches. The statistical based methods were the first method used for outlier detection depends on the distribution model and further classified into two groups: parametric and non-parametric. In parametric technique, there is no prior data distribution information and non parametric method is not suitable for real time application. Unsupervised outlier detection based on data nearest for outlier detection (DNDO) is suggested in (Abid et.al, 2016). The nearest neighbor-based approach based on distance-based algorithms where the focus is on the computation of the distances between observations. A point is viewed as an outlier if it is far away from its nearest neighbors.

In case of density based algorithm, the focus is in the low-density region which needs improve in efficiency. The principle of clustering based method is the application of standard clustering technique and need not be supervised. In some clustering technique, the choice of cluster width and continuous change of data stream does not make suitable for outlier detection. However through incremental learning method, the recent clustering techniques tackled the issue. Authors in (Rajasegarar et al. 2006) proposed a clustering based global outlier detection technique to detect outliers in sensor nodes. An abnormal cluster is identified if the average inter-cluster distance of the cluster is higher than a threshold value of the inter-cluster distance set. The classification based method is suitable for multidimensional data but the main problem is the computational complexity. The classification based outlier detection techniques can be classified into Bayesian network-based, support vector machine (SVM) based and deep learning based approaches. The objective of classification is to identify a classifier considering a training data set which is a set of labeled data instances. Then a testing data set is classified with an unseen instance into one of the learned class either normal or outlier. Authors in (Hill et al., 2007) presented a dynamic Bayesian network approach. Two techniques (one is Bayesian credible interval and another is maximum posteriori measurement status) used in the dynamic network topology that evolved over time. This technique operates on several data streams at once to identify local outliers in environmental sensor data. (Rajasegarar et al. 2007) proposed a SVM based approach. This approach used one class quarter SVM. The sensor data outside of the quarter sphere is known to be outlier. Outlier detection using various one-class SVMs, namely hyper-plane, hyper-sphere, quarter-sphere and hyper-ellipsoid is presented and reviewed in (Shahid et al., 2015). Support vector data description based on spatio-temporal correlation (STASVDD) based approach is presented in (Chen et al., 2019). In reference (Luo and Nagarajan, 2018), the autoencoder neural network based approach is suggested and a time series based on recurrent autoencoder ensembles is proposed in (Kieu et al., 2019) .

Kernel principal component analysis (KPCA) based Mahalanobis kernel is a novel outlier identification approach that uses the Mahalanobis distance to implicitly compute the mapping of data points in the feature space, allowing us to distinguish outlier points from the typical pattern of data distribution. (Ghorbel et al., 2015). For nonlinear cases, KPCA was utilized to extract higher order statistics.It uses a nonlinear function to map the data onto another feature space. In (Zhang et al., 2019), an outlier detection and recovery technique based on an artificial neural network (ANN) is proposed that may be used to identify whether temperature readings recorded by WSN sensors are outliers. Authors in (De Paola et al., 2015) proposes an adaptive distributed Bayesian technique for identifying outliers in data gathered by a wireless sensor network, with the objective of maximizing classification accuracy, time complexity, and communication complexity while also taking externally imposed constraints into account.An isolation-based distributed outlier detection framework using nearest-neighbor ensembles (iNNE) to detect outliers in WSN is suggested in (Wang et al., 2019). The iNNE method constructs local detectors in each node in this architecture.The approach is based on the concept of weighted voting.A sliding window is provided to update local detectors, allowing adaptation to dynamic changes in the environment.Mobile sinks are introduced in (Bhushan and Sahoo, 2019), which offer consistent energy usage and load-balanced data transmission over the sensor network. Authors in (Bhushan and Sahoo, 2020) propose a safe and energy-efficient Intelligent and Secured Fuzzy Clustering Algorithm with Balanced Load Sub-cluster Formation (ISFC-BLS) routing protocol for WSNs. In (Heinzelman et al., 2002), a low-energy adaptive clustering hierarchy (LEACH), a protocol architecture for micro-sensor networks is proposed that combines the ideas of energy-efficient cluster-based routing and media access together with application-specific data aggregation to achieve good performance in terms of system lifetime, latency, and application-perceived quality.

Recently more attention has been given in learning-based methods such as active learning and deep learning for outlier detection problems in WSNs (Chalapathy and Chawla, 2019). Convolutional Neural Network (CNN), Stacked Autoencoders (SAE), Deep Belief Networks (DBN), Long Short-Term Memory Recurrent Networks (LSTM) etc are several deep learning neural networks which are well suited for dealing with different classification problems. These neural networks deal with non-linear large-scale data with skewed properties (Kwon et al., 2017). Deep learning method is a good choice in the field of outlier detection because of the advantages of learning the features directly from the original data automatically.

We attempted to convert the sensor data streams generated by sensor nodes which are densely deployed over a large geographical area into images and used a CNN model for learning and testing. A CNN model recognizes complex function approximation by learning the deep nonlinear network structure. It represents the input-output mapping relationship and simultaneously learns the basic characteristic of a data set from a small sample set.

3. OUTLIER DETECTION USING CNN

3.1. Convolutional Neural Network (CNN)

The Convolutional Neural Network (CNN) is a multilayer neural network named from mathematical linear operation between matrixes known as convolution has four main layers: convolutional layer, *ReLu* layer (non-linearity), pooling layer, and fully- connected layer. CNN architecture is formed by stack of these layers. The convolutional and fully-connected layer has parameters but pooling and non-linearity layer does not have parameters (Albelwi and Mahmood, 2017). Figure 3 shows outlier detection process using CNN:

1. **Convolutional layer:** The convolutional layer plays a vital role in CNN operation. The parameters of this layer emphasize the use of learnable kernels which consists of several feature maps. It determines the output of neurons which are connected to local regions of the input through the scalar product calculation between their weights for each value in that kernel. Every kernel having an activation map stacked along the dimensional depth forms the output. When the data strike the convolutional layer, the layer convolves each filter of the input produces a 2D activation map. This layer's output volume ($M_{new}\times N_{new\times} Q_{new}$) is controlled by four hyperparameters given an input of size M×N× Q: It is computed as follows: (i) number of filters (X), (ii) filter size (F×F×R), (iii) quantity of zero padding (P), and stride (S), and is calculated as follows (Das et al., 2020):

$$\begin{aligned} M_{new} &= (M - F + 2p) \,/\, s + 1 \\ N_{new} &= (N - F = 2P) \,/\, S + 1 \\ Q_{new} &= X.(Q \,/\, R) \end{aligned} \tag{1}$$

2. **BN Layer:** The BN layer uses the mini-batch mean and standard deviation to normalize its inputs over a mini-batch.Following that, this layer shifts and scales the inputs using two learnable parameters, β and γ respectively. Assume B= {x_1, x_2,,...,x_m} is a mini-batch of size m (Das et al., 2020):

$$\begin{aligned} \mu_B &= \frac{1}{m}\sum_{i=1}^{m} x_i \\ \sigma_B^2 &= \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_B)^2 \\ \widehat{x}_i &= \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \\ n\_x_i &= \gamma \widehat{x}_i + \beta \end{aligned} \tag{2}$$

where μ_B and σ_B^2 indicate the mini-batch mean and variance, respectively. $\widehat{x}_i$ is the batch normalized output for a mini-batch B, while n_x_i is the normalized input with zero mean and unit variance.BN is often employed between the convolutional and ReLU layers.

3. **ReLu layer:** The rectified linear unit commonly referred as ReLu which is a non-linear activation function also referred as piecewise linear function, chosen for calculating the feature map generated by the filters. The feature map is calculated by the ReLu activation as:

$$h_i^k = \max\,(w_k x_i, 0) \tag{3}$$

where h_i^k is the kth feature map at a given layer, i is the feature map index, x_i is the input and w_k denotes weights.Because of the computational simplicity, representational sparsity and linear behavior, the ReLu activation function has become default activation function used in almost all convolutional networks.

4. **Pooling layer:** The main concept of this layer is to reduce the dimension of the feature maps which reduce the complexity for further layers and increase the robustness of feature extraction. The pooling layer does not affect the number of filters and performs down sampling along the spatial dimensionality of the given input. Max-pooling is considered as default pooling layer which reduces the number of parameters within that activation:

$$f_{max}(x) = \max\,(x_i) \tag{4}$$

where xi is input data vector with activation values.

5. **Fully connected layer:** After all of the features generated, they are passed to the softmax fully connected layer. Each node in this layer is directly connected to every node in both the previous and the next layer by taking all the neuron from the previous layer and combines them into one layer. The fully connected layer contains neurons of which are directly connected to the neurons in two adjacent layers, without being connected to any layers within them. The output of fully connected layer is the probability distribution of all classes which is the final result of classification.

3.2. Proposed Approach

The data generated from different sensor nodes deployed at different locations are collected with a specific epoch duration are stored in a dataset. The data preprocessing is to convert the data in the dataset to time series data and then outlier detection of this data by using CNN model with EEG classification. We proposed convolutional neural network which can deal with the structure of EEG data as it is able to learn two-dimensional representation of data.

In this paper, we proposed the methodology consists of the following five steps:

Step-1: In data processing, the data were separated into overlapping one-second frames. Windowing is the process of taking small subset of a larger dataset, for processing and analysis which alter the spectral properties of that dataset. Here a Hanning window is applied over the frames.
Step-2: To transform the data of each frame from time domain to frequency domain, First Fourier Transform (FFT) is then applied. Fourier transform is a function which transforms a time domain signal into frequency domain by accepting a time signal as input and produces the frequency representation as an output.
Step-3: Here the FFT amplitudes are grouped into theta, alpha, and beta ranges based on the FFT amplitudes of sensor node attributes (surface temperature, ambient temperature, humidity)

Figure 2. Illustration of convolution operation (Das et al., 2020)

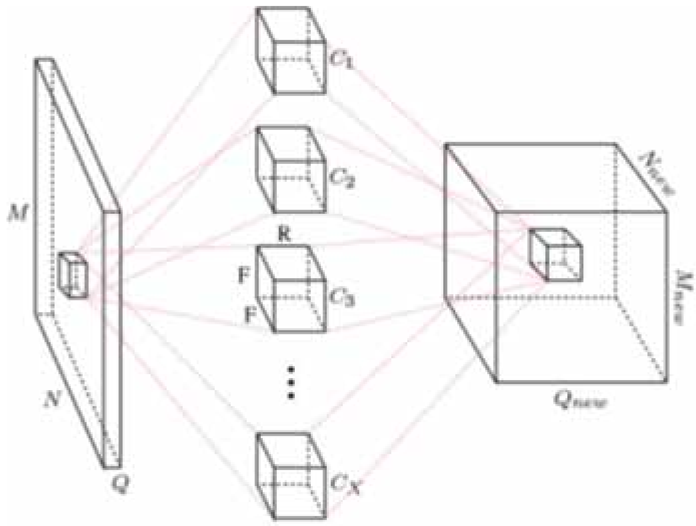
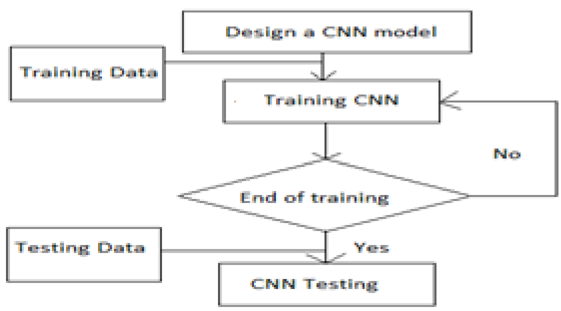


Figure 3. Outlier detection process using CNN



respectively, which gives three scalar values for each sensor node per frame. Real-world data tend to be noisy, so data cleaning routines attempt to smooth out noise while identifying outliers in the data. Here we are concerned with frequency binning for data smoothing. In this method the data is first sorted and then distributed into a number of buckets or bins. Sorted data values smoothed by confer with the neighbor data that is the value around it by binning method. In equal depth or frequency binning we divide the range of the variable into intervals that contain approximately equal number of points and equal frequency may not possible due to repeated values.

Step-4: The three scalar values generated from step-3 are converted as RGB color channels onto a 2D map by 2D Azimuthal projection.

Step-5: These 2D map projections of theta, alpha and beta ranges are trained in CNN to measure the validation accuracy and detection accuracy.

Figure 4. Logical steps of the methodology

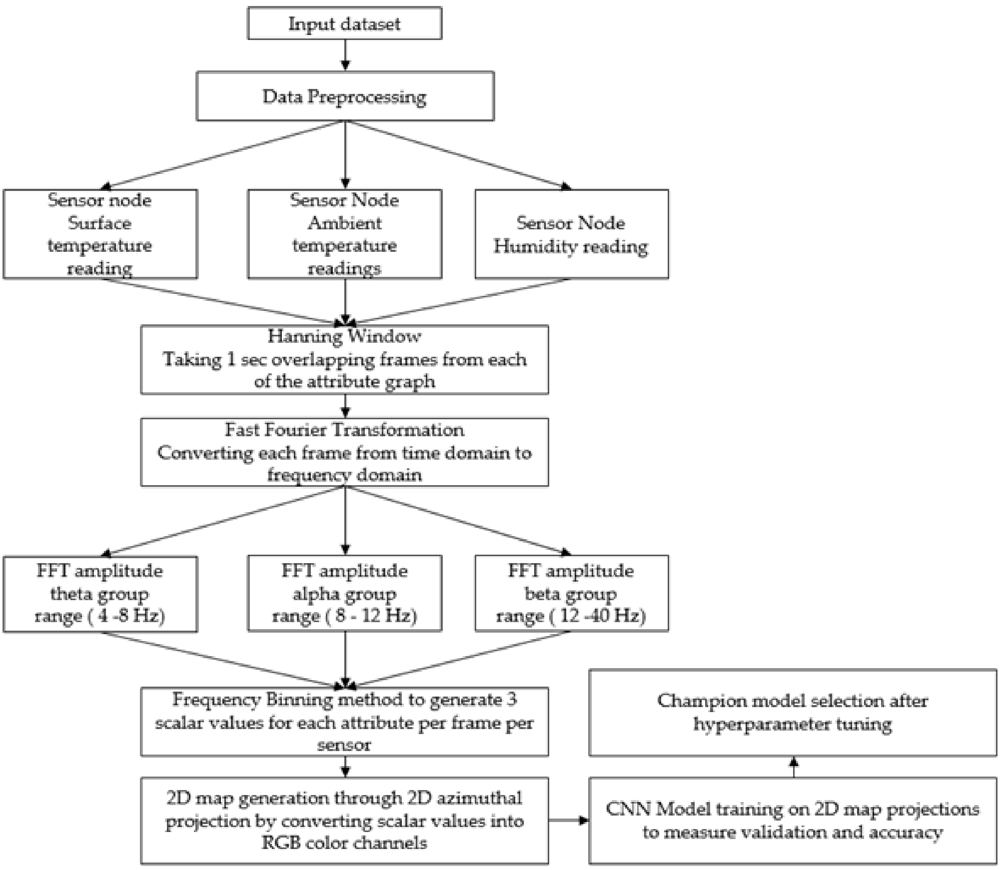
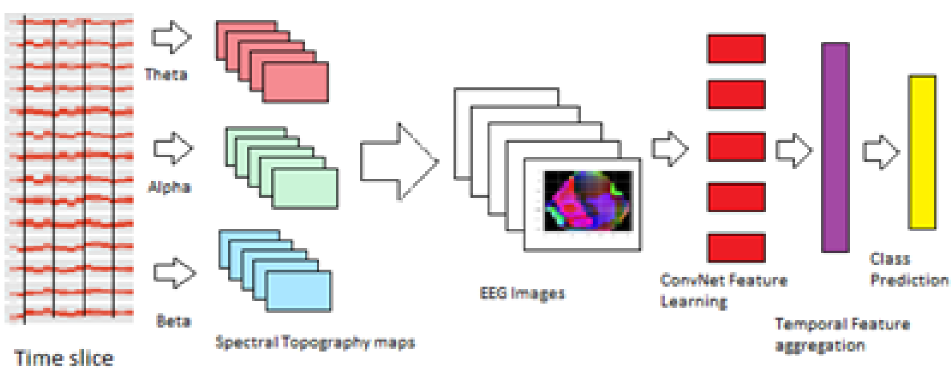


Figure 5. EEG classification architecture



4. EXPERIMENTAL RESULTS AND DISCUSSION

4.1. Dataset

In this study, we analyzed some data samples extracted from the dataset (Barrenetxea, 2019) of the sensor-scope-Lausanne Urban Canopy Experiment (LUCE), which was collected between July 2006 and May 2007 through a sensor scope project on the campus of the cole Polytechnique Fdrale de Lausanne (EFPL). In the central part of the campus, a network of 92 wireless weather sensor nodes covering an area of 300 × 400 m was deployed. The sensor nodes were deployed to measure ambient temperature, surface temperature, humidity, wind speed etc. From the sensor nodes with ID 10 are included in this work by taking data based on time. Synthetic outliers are generated by using fault models suggested in (Reece et al., 2009) and inserted into the dataset.The data in the dataset converted to waveform data and the waveform signals from 17 days with time slices are taken with sampling rate 128 Hz.

4.2. Data Processing

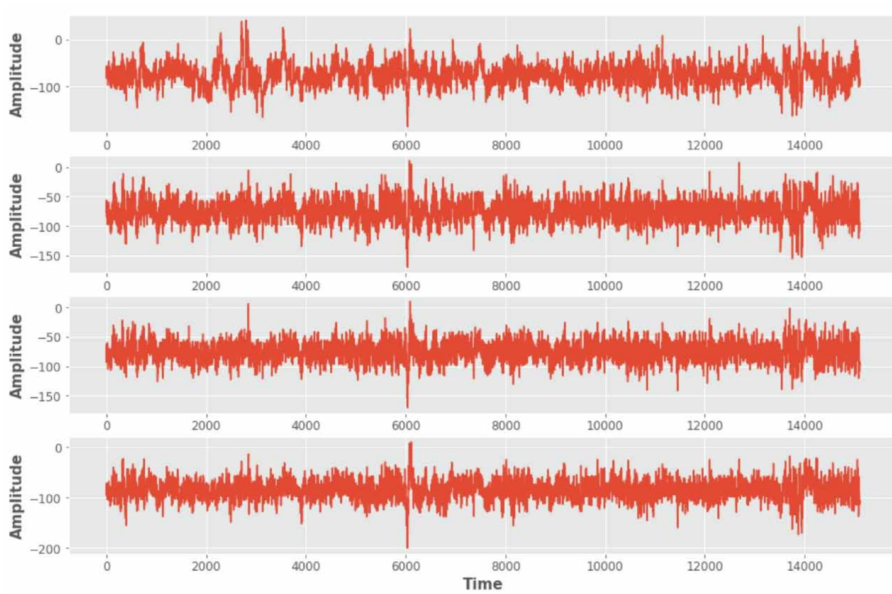
4.2.1 Hanning Window

Generally, windows are used to reduce spectral leakage when FFT is performing on time data. Any signal data can be periodic or non-periodic based on how it is captured by time measurement. A window can reduce the leakage present in a non-periodic signal. Hanning window is normally used for operational noise measurement in random data which is non-periodic.

The Hanning function named after Julius Von Hann is used to create a "window" for Fourier Transform filtering. The Hanning window is defined as:

$$w(n) = \alpha - \alpha * \cos\left(\frac{2\pi n}{M-1}\right), 0 \le n \le M\text{-}1 \quad (5)$$

Figure 6. Data in waveform from four of the 17 sensor scope time series data



where α = 0.5.

The factor 1/M is chosen rather than 1/M-1 to give the best behavior for spectral estimation of discrete data as the end points of Hann value just touch 0. It is also known as raised cosine as it is a member of cosine-sum and power-of-sine families.

4.2.2 First Fourier Transform and Frequency Binning

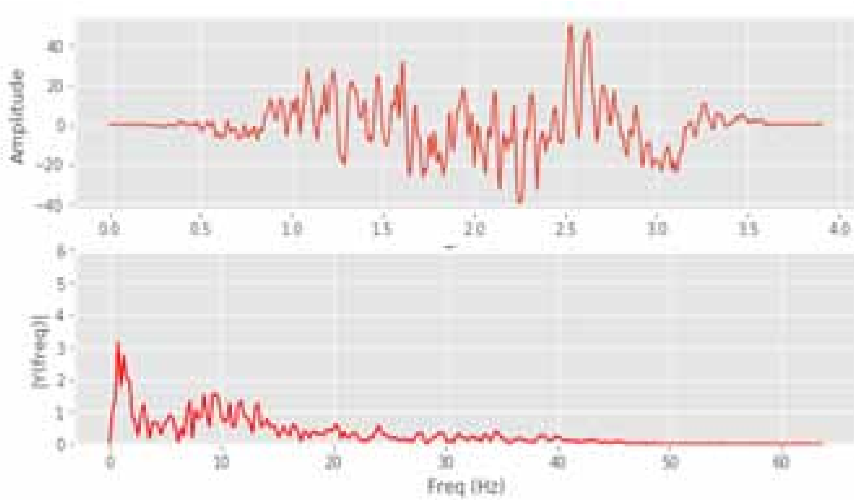
The Fourier transform takes a signal and breaks it down into sine waves of different amplitudes and frequencies. Fourier theorem states that any waveform in the time domain can be represented by the weighted sum of sine's and cosines. FFT is applied fortime domain to frequency domain transformation of each frame data (Herff and Krusienski, 2019). Looking at signals in the frequency domain can help for validating and troubleshooting the signals. The frequency domain is great at showing if a clean signal in the time domain actually contains noise or jitter. FFT amplitudes are classified into theta (4– 8 Hz), alpha (8-12 Hz), and beta (12-40 Hz) ranges giving three scalar values for each sensor per frame. Binning is a data smoothing technique. There are different smoothing techniques. Smoothing by "bin means" where each value in a bin is replaced by the mean value of the bin. Smoothing by "bin median" where each value in a bin is replaced by the median value of the bin. Smoothing by "bin boundary" where the minimum and maximum values in a given bin are identified as the bin boundaries and then each bin value is replaced by the closest boundary value.

4.2.3 2D Azimuthal Projection

Preservation of the directions, from a central point is the property of Azimuthal projection, where the straight lines on the map represent great circles through the central point. These projections have radial symmetry in the scales and also in distortions. A plane tangent to the map among the central point as tangent point can be imagined and visualized as the mapping of radial lines.

Whether the plane is tangent (plane having one point of contact) or secant (plane having entire line of intersection), we can minimize the level of choosing standard lines. The three values of theta, alpha and beta were converted as RGB color channels onto a 2D map projection. The function computes the Azimuthal equidistant projection of input point in 3D Cartesian Coordinates and the result returns projected coordinates using Azimuthal equidistant projection.

Figure 7. Hanning windowed frame and FFT



4.3. Evaluation Metric

The following metrics are used to evaluate the performance of the proposed method:

$$\text{Accuracy rate} = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$\text{Precision (P)} = \frac{TP}{TP + FP} \tag{7}$$

$$\text{True Positive Rate (Recall) TPR} = \frac{TP}{TP + FN} \tag{8}$$

$$\text{False Positive Rate FPR} = \frac{FP}{FP + TN} \tag{9}$$

$$\text{F1} = \frac{2\left(Precision \times Recall\right)}{Precision + Recall} \tag{10}$$

In equation 4, TP denotes the number of true-positive results, TN represents the number of true-negative results, FP is the false-positive results and FN represents the number of false-negative results. In equation 8, F1is the harmonic mean that measures the quality of classifications.

4.4. Experimental Setup

A high-level python library Pymote 2.0 (Shahzad, 2016) is used for simulation and Tensorflow to implement our CNN model. Seventy percent of dataset was used for training and thirty percent used for

Figure 8. 2D projections having theta, alpha and beta ranges

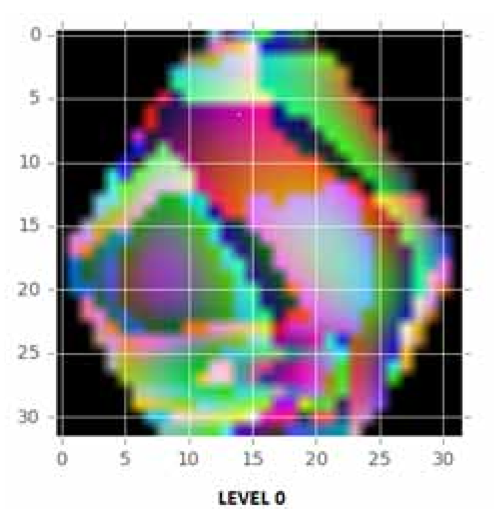
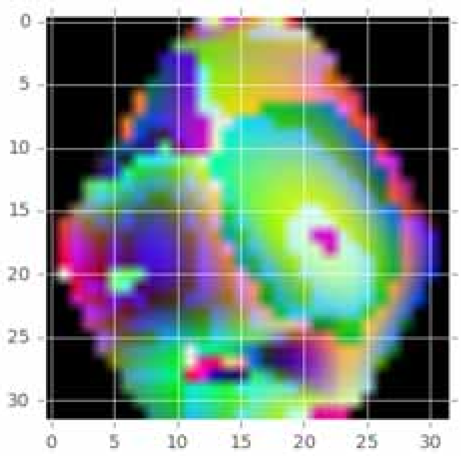


Figure 9. 2D projections having theta, alpha and beta ranges



testing.A fully-connected layer and a softmax layer preceded by a VGG style CNN network is used. A maxpool layer separates the stack and kernels number in each layer becomes double of previous stack. The complete network with maxpooling layer in time was build by inputting the EEG images (one image per time window), number of classes, size of the input images (a square input), number of color channels is 3 (RGB) and number of time window in the snippet returns a pointer to the output of last layer. The conv2D layer builds the complete network to integrate time from sequence of EEG images.

A sample training function was build which loops over the training set and evaluates the network on the validation set after each epoch. It evaluates the network on the training set by inputting the images, target labels, tuple of (train, test) index numbers, model type, batch size for training, number of epochs of dataset to go over for training.

4.5. Experimental Results

The images in Fig 6 show the data in raw waveform from four of the 17 EEG time series data from the dataset. Fig. 7 shows the Hanning windowed one-second frame and application of FFT for transformation of data for each frame from time domain to frequency domain. The image in Fig 8 shows 2D projections having theta, alpha and beta ranges. After frequency binning, the FFT amplitudes classified into theta (4-8), alpha (8-12) and beta (12-40) ranges giving three scalar values and then these three scalar values of theta, alpha and beta were converted as RGB color onto a 2D map projected images.

The EEG images used as input for CNN where we used three convolutional layers with 32 convolutional filters using small receptive fields of size 3×3, three ReLu, and three pooling layers with max-pooling block size 2×2. The dropout probability is 0.5. RMSprop optimizer with learning rate 0.001 was used. The time spent on training was 200 epochs with batch size 128. The classifier based on EEG classification is first trained with the training dataset and tested with the test dataset of sensorscope. A confusion matrix often used to define a classification model's performance where the true values are known. Fig. 9 shows the confusion matrix for the proposed approach. The performance of the proposed classifier is compared with the state-of-art classifiers taking into the performance parameters such as TPR and FPR as shown in Table 1. The proposed model is evaluated on the training dataset and validation dataset. The training and validation loss is shown in Fig. 10 and the training accuracy is shown in Fig 11.

Figure 10. Confusion matrix

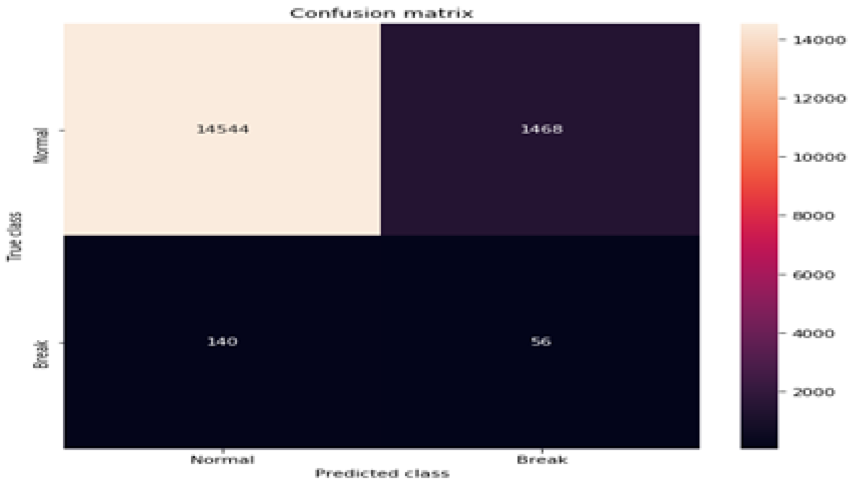


Figure 11. Training and validation loss

Figure 12. Training accuracy

Table 1. Comparison with experimental results

Method	AR	TPR	FPR	P	F1
SVM(12)	82.29	84.86	42.11	95.05	89.66
DNDO(9)	84.51	87.04	39.83	95.44	91.05
Proposed	85.21	89.45	27.10	95.62	92.54

5. CONCLUSION AND SUGGESTION

The main objective of the outlier detection is to identify the unruly nodes and to restrict the data reported by those nodes to enter into the network. In this paper, we have presented a CNN based online outlier detection method that is integrated with EEG classification. This approach transforms the sensor data into sequence of images in the preprocessing step. The performance according to Accuracy, TPR, FPR, Precision and F1 is compared with the state-of-art techniques which show significant improvements in detection accuracy. The research enables many more opportunities to improve further. The number of test images can be increased substantially and better libraries like fastai, PyTorch and techniques using different CNN architectures like DenseNet201, ResNet-101 and Inceptionv3 can be used in future scope of this work.

CONFLICTS OF INTEREST

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

Funding Statement

No funding was received for this work.

Process Dates:

Received: February 2, 2021, Revision: July 30, 2021, Accepted: August 3, 2021

Corresponding Author:

Correspondence should be addressed to Biswaranjan Sarangi, biswaranjan.sarangi@gmail.com

REFERENCES

Abid, A., Kachouri, A., & Mahfoudhi, A. (2016). Anomaly detection through outlier and neighborhood data in wireless sensor networks. In *Advanced Technologies for Signal and Image Processing* (pp. 26–30). ATSIP.

Albelwi, S., & Mahmood, A. (2017). A framework for designing the architectures of deep convolutional neural networks. *Entropy (Basel, Switzerland)*, *19*(6), 1–20.

Ayadi, A., Ghorbel, Q., Obeid, A. M., & Abid, M. (2017). Outlier detection approaches for wireless sensor networks: A survey. *Computer Networks*, *129*(1), 319–333.

Barnett, V., & Lewis, T. (1994). *Outliers in Statistical Data*. John Wiley Sons.

Barrenetxea, G. (2019). *Sensorscope Data*. Zenodo. Dataset. 10.5281/zenodo.2654726

Bashivan, P., Mohammed Yeasin, I., & Codella, N. (2016). Learning Representations from EEG with Deep Recurrent-Convolutional Neural Networks. *International conference on learning representations*.

Bhushan, B., & Sahoo, G. (2018). Recent Advances in Attacks, Technical Challenges, Vulnerabilities and Their Countermeasures. *Wireless Sensor Networks. Wireless Personal Communications*, *98*, 2037–2077.

Bhushan, B., & Sahoo, G. (2019). Routing Protocols in Wireless Sensor Networks. In *Computational Intelligence in Sensor Networks.Studies in Computational Intelligence* (pp. 215–248). Springer.

Bhushan, B., & Sahoo, G. (2019). E^2SR^2: An acknowledgement-based mobile sink routing protocol with rechargeable sensors for wireless sensor networks. *Wireless Networks*, *25*, 2697–2721.

Bhushan, B., & Sahoo, G. (2020). ISFC-BLS Intelligent and Secured Fuzzy Clustering Algorithm Using Balanced Load Sub-Cluster Formation, in WSN Environment. *Wireless Personal Communications*, *111*, 1667–1694.

Chalapathy, V., & Chawla, S. (2021). Deep learning for anomaly detection: A survey. *ACM Computing Survey, 54*(2), 1–38.

Chen, Y., & Li, S. (2019). A lightweight Anomaly Detection Method based on SVDD for wireless sensor network. *Wireless Personal Communications*, *105*, 1235–1256.

Das, D., Nayak, D. R., Dash, R., Majhi, B., & Zhang, Y.-D. (2020). *H-wordnet:* A holistic convolutional neural network approach for handwritten word recognition. *IET Image Processing*, *14*(9), 1794–1805.

Ghorbel, O., Ayedi, W., Snoussi, H., & Abid, M. (2015). Fast and Efficient Outlier Detection Method in Wireless Sensor Networks. *IEEE Sensors Journal*, *15*(6), 3403–3411.

Hawkins, D. M. (1980). *Identification of Outliers*. Chapman and Hall.

Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, *1*(4), 660–670.

Herff, C., & Krusienski, D. J. (2019). Extracting Features from Time Series. In *Fundamentals of Clinical Data Science*. Springer.

Hill, D. J., Minsker, B. S., & Amir, E. (2007). Real-time bayesian anomaly detection for environmental sensor data. the International Association of Hydraulic Engineering and Research, 1-10.

Kieu, T., Yang, B., Guo, C., & Jensen, C. S. (2019). Outlier detection for time series with recurrent autoencoder ensembles. *28th Int. Joint Conf. Artificial Intelligence (IJCAI)*, 2725-2732.

Kwon, D., Kim, H., Kim, J., Suh, S. C., Kim, I., & Kim, K. J. (2017). A survey of deep learning-based network anomaly detection. *Cluster Computing*, *10*, 1–13.

Luo, T., & Nagarajan, S. G. (2018). Distributed Anomaly Detection using Autoencoder Neural networks in WSN for IoT. *IEEE International Conference on Communications (ICC)*, 1-6.

Mahapatro, A., & Khilar, P. (2013). Fault diagnosis in wireless sensor networks: A survey. *IEEE Communications Surveys and Tutorials*, *15*(4), 2000–2026.

Mahapatro, A., & Khilar, P. M. (2011). Scalable Distributed Diagnosis Algorithm for Wireless Sensor Networks. In *Advances in Computing, Communication and Control. ICAC3* (pp. 400–405). Springer.

Paola, A. D., Gaglio, S., Re, G. L., Milazzo, F., & Ortolani, M. (2015). Adaptive Distributed Outlier Detection for WSNs. *IEEE Transactions on Cybernetics*, *45*(5), 902–913.

Puri, D., & Bhushan, P. (2019). Enhancement of security and energy efficiency in WSNs: Machine Learning to the rescue. *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, 120-125.

Rajasegarar, S., Leckie, C., Palaniswami, M., & Bezdek, J. C. (2006). Distributed Anomaly Detection in Wireless Sensor Networks. *IEEE Singapore International Conference on Communication Systems*, 1-5.

Rajasegarar, S., Leckie, C., Palaniswami, M., & Bezdek, J. C. (2007). Quarter Sphere Based Distributed Anomaly Detection in Wireless Sensor Networks. *IEEE International Conference on Communications*, 3864-3869.

Reece, S., Roberts, S., Claxton, C., & Nicholson, D. (2009), Multi-sensor fault recovery in the presence of known and unknown fault types. *12th IEEE International Conference on Information Fusion*, 1695-1703.

Sadik, S., & Gruenwald, L. (2011). Online outlier detection for data streams. *Proceedings of the 15th Symposium on International Database Engineering and Application*, 88-96.

Shahid, N., Naqvi, L. H., & Qaisar, S. B. (2015). One-class support vector machines: Analysis of outlier detection for wireless sensor networks in harsh environment. *Artificial Intelligence*, *43*, 515–563.

Shahzad, F. (2016). Pymote, 2.0: Development of an interactive python frame-work for wireless network simulations. *IEEE Internet Things J.*, *3*(6), 1182–1188.

Wang, Y., Attebury, G., & Ramamurthy, B. (2006). A survey of security issues in wireless sensor networks. *IEEE Communications Surveys and Tutorials*, *8*(2), 2–23. doi:10.1109/COMST.2006.315852

Wang, Z.-M., Song, G.-H., & Gao, C. (2019). An Isolation-Based Distributed Outlier Detection Framework Using Nearest Neighbor Ensembles for Wireless Sensor Networks. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 96319–96333.

Zhang, K., Yang, K., Li, S., Jing, D., & Chen, B. (2019). ANN-Based Outlier Detection for Wireless Sensor Networks in Smart Buildings. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 95987–95997.

Zhang, Y., Meratnia, N., & Havinga, P. (2010). Outlier Detection Techniques for Wireless Sensor Networks: A Survey. *IEEE Communications Surveys and Tutorials*, *12*(2), 159–170.

Biswaranjan Sarangi did his M.Tech in Computer Science & Engineering from F.M.U, Balasore, Odisha and pursuing his Ph.D in Computer Science & Engineering under BPUT, Odisha. His current research interest is machine learning & AI, neural network, and wireless sensor networks.

Arunanshu Mahapatro graduated from University of Pune in Electronics Engineering in the year 1999 and did his M.Tech. in Electronics and Communication Engineering from KIIT University and PhD from National Institute of Technology Rourkela. Presently, he is affiliated to Department of Electronics and Telecommunication Engineering, Veer Surendra Sai University of Technology (VSSUT), Burla. His current research interests include fault tolerant computing, wireless sensor networks and cognitive computing for wireless networks

Biswajit Tripathy was educated at Utkal University obtained his B.Tech (1991) and M.Tech from FM University (2008) and PhD from Biju Patnaik University of Technology(2018). He is currently working as Professor and Head in Dept of CST, GITA Autonomous College, Bhubaneswar (India). He is having total 29 years' experience both industries and teaching in the field of Computer Science. His area of interest is E-commerce, Software Engineering, Artificial Intelligence, IOT, Software Project Management and Network Security.