Synchrophasor Data Event Detection using Unsupervised Wavelet Convolutional Autoencoders

Jacob Buckelew, *Student Member, IEEE,* Sagnik Basumallik, *Member, IEEE,* Vasavi Sivaramakrishnan, Ayan Mukhopadhyay, *Member, IEEE,* Anurag K. Srivastava *Fellow, IEEE*, Abhishek Dubey, *Senior Member, IEEE*

***Abstract*—Timely and accurate detection of events affecting the stability and reliability of power transmission systems is crucial for safe grid operation. This paper presents an efficient unsupervised machine-learning algorithm for event detection using a combination of discrete wavelet transform (DWT) and convolutional autoencoders (CAE) with synchrophasor phasor measurements. These measurements are collected from a hardware-in-the-loop testbed setup equipped with a digital real-time simulator. Using DWT, the detail coefficients of measurements are obtained. Next, the decomposed data is then fed into the CAE that captures the underlying structure of the transformed data. Anomalies are identified when significant errors are detected between input samples and their reconstructed outputs. We demonstrate our approach on the IEEE-14 bus system considering different events such as generator faults, line-to-line faults, line-to-ground faults, load shedding, and line outages simulated on a real-time digital simulator (RTDS). The proposed implementation achieves a classification accuracy of 97.7%, precision of 98.0%, recall of 99.5%, F1 Score of 98.7%, and proves to be efficient in both time and space requirements compared to baseline approaches.**

***Index Terms*—convolutional neural network, hardware-in-the-loop, unsupervised machine learning, phasor measurement units**

I. Introduction

DETECTION and categorization of grid events, an example of which is shown in Fig 1, as well as locating the source of the event to expedite the recovery process, is extremely essential for ensuring the overall reliability and resiliency of the power system. Inadequate detection may lead to widespread outage incidents affecting hundreds of millions of customers. This is particularly evident from the 1996 west coast blackout, the 1997 New York City blackout, and the 2012 north-India blackout, caused due to a combination of overloaded line tripping, system faults, poor situational awareness, and human errors [1]. High-resolution Phasor Measurement Unit (PMU) data and machine learning (ML) applications have opened up opportunities to improve grid resiliency by constructing extensive data-driven models that facilitate better situational awareness and event detection [2], [3]. Various detection methods have been developed including supervised (with labeled data) [4]–[7], unsupervised (no labeled data) [8]–[14], or semi-supervised (few labeled data) [15]–[17].

J. Buckelew, A. Mukhopadhyay and A. Dubey are with Vanderbilt University, Nashville, TN. S. Basumallik, V. Sivaramakrishnan, and A. Srivastava are with the West Virginia University, Morgantown, WV, USA (e-mail: jacoboribuckelew@gmail.com). This work is funded in part by the National Science Foundation under the award numbers 1840052, 1840192, and 2238815.

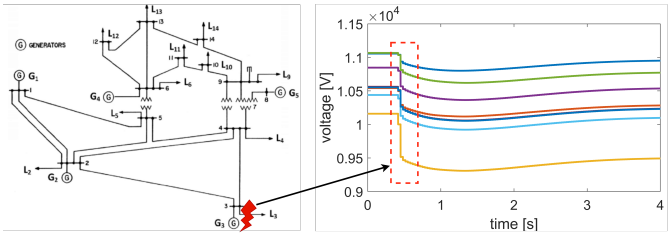

Fig. 1: (a) IEEE 14-bus power transmission system, (2) PMU data corresponding to an event on bus 3 at 0.5s.

There exist key challenges in current PMU-based event detection methods. First, supervised learning is not well-suited for most anomaly detection problems due to the need for label data, which may often be unavailable, and the high cost associated with labeling. The literature on unsupervised event detection is new and still limited [8]–[14]. Second, raw PMU data is generated in the form of a time series and numerous analyses [18]–[20] have shown that analyzing PMU data for events in the time-frequency domain is a more effective approach compared to simple time series analysis.

To address the above challenges, this paper presents the **W**avelet-**C**onvolutional **A**uto **E**ncoder (W-CAE) framework, shown in Fig. 2, for power transmission system event detection and localization. The framework has two main components: the Discrete Wavelet Transform (DWT) and the Convolutional Auto Encoder (CAE). We avoid Fourier Transform as it lacks time information of frequencies, which may lead to inaccurate event analysis. Instead, we employ the DWT that sends PMU data from the time domain to the joint time-frequency domain. DWT is utilized to determine the critical features of our ML model, as well as for the purpose of event localization. The decomposed signal from DWT is fed to the CAE learning model. By treating the decomposed signals as images, we leverage the benefits of convolutional operations in our framework to detect anomalous samples based on reconstruction error. If a reconstruction error for a particular sample is above a certain threshold, then the sample is flagged as an anomaly. By providing computationally efficient processing on PMU samples, the proposed W-CAE framework can be used by grid operators in real-time during the diagnosis phase following an event. The key contributions of this paper are as follows,

1) Proposed a fully unsupervised anomaly detection algorithm that incorporates discrete wavelet transform for

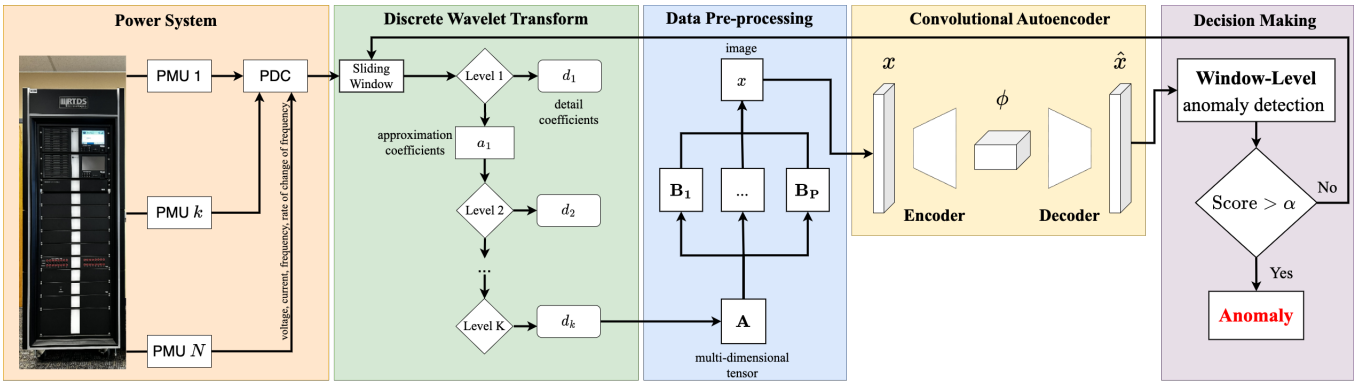

Fig. 2: Overview of the W-CAE framework for power system event detection.

feature extraction and a convolutional autoencoder for detecting several types of events.

2) Validated the algorithm using PMU data obtained from a hardware-in-the-loop real-time digital simulator (RTDS).
3) Evaluated the efficiency of the proposed anomaly detection algorithm compared to a baseline method using classification metrics such as accuracy, precision, recall, the area under the curve (AUC), and F1 score.
4) Evaluated the efficiency of the proposed anomaly detection algorithm in both time and space requirements against a baseline method.

The rest of the paper is organized as follows - Section II gives an overview of various PMU-based event detection methods. Section III formalizes the event detection problem statement and provides an overview of our method. Section IV discusses the proposed method in detail. Section V presents the evaluation results of our proposed method. Section VI concludes our work and addresses future research directions.

II. State-of-the-Art

PMUs provide a massive amount of high-resolution time-synchronized voltage and current phasors at the rate of $30/60/120/240$ frames/s. With the increase in governmental support [21], it is anticipated that PMUs will further proliferate in the near future for wide-area grid monitoring systems to improve situational awareness, detect abnormalities, analyze their root causes, and take prompt appropriate actions. The ML-based anomaly detection problem using PMU measurements can be broadly classified into three categories: supervised, semi-supervised, and unsupervised methods.

Supervised methods use labeled data to train a classifier. In [4], a convolutional neural network (CNN) is used with scalograms generated from a pseudo-continuous quadrature wavelet transform. The authors in [5] exploit the temporal structure of the PMU data through the correlation matrix with CNN. Other approaches used for detecting events and attacks include Extreme Learning [6] where a state reconstruction method was constructed and a text-mining method [7] through symbolic aggregation approximation.

Semi-supervised approaches make use of both labeled and unlabeled data for event detection. In [15], a Generative Adversarial Network-based (GAN) model is combined with a signal-processing sorting algorithm to detect anomalies through highly correlated PMU samples. A Deep Autoencoder is constructed in [16] that detects physically induced anomalies in PMU data. Lastly, GANs and autoencoders are combined in [17] to create a Fault-Attention Generative Probabilistic Adversarial Autoencoder for the detection of anomalies.

Unsupervised learning does not require labeled data. Examples include adaptive and online Isolation Forest method [8] and principal component analysis (PCA) [9]. In [10], a change-point detection method is used for event detection, followed by PCA for event identification. The authors in [11] generate spectrograms via short-time Fourier Transform and use a convolutive dictionary model for event detection via generalized likelihood ratio tests. DBSCAN provides real-time event detection by leveraging the spatiotemporal differences of anomalous samples [12]. In [13], an ensemble approach uses multiple k-Means clustering models. In [14], a spatiotemporal graph autoencoder exploits spatial and temporal relationships of PMUs for event classification and localization.

This paper extends the research in unsupervised power system event detection by proposing a novel wavelet convolutional autoencoder. By transforming PMU data into the joint time-frequency domain using DWT, the autoencoder is able to detect events without needing data labels.

III. Overview of the W-CAE Framework

We formulate the problem of real-time monitoring as a general anomaly detection problem with the goal of finding diversions from the expected normal grid operations. Fig. 2. shows the proposed W-CAE framework.

Data Generation: PMU data is generated from a real-time RTDS testbed at a rate of 30 frames/s. The simulated data model the state of operation under normal conditions as well as the occurrence of power disturbances.

Feature Extraction: We perform DWT on windows of PMU signals to extract detail coefficients. DWT targets disturbances in non-stationary signals and provides time localization.

Feature Pre-processing: We structure the output of DWT as an image where each image has dimensions defined by the

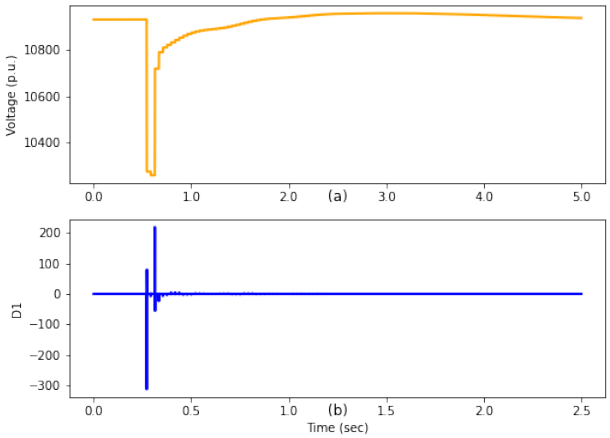

Fig. 3: (a) Voltage signal and (b) Detail coefficients from a one-level decomposition after a line-to-line fault event.

number of channels across all PMUs and the length of a time window. Thus, a single image covers the state of the grid in any given time window.

Learning Model: The deep learning model we employ is a CAE that learns an underlying manifold of the input space (decomposed PMU signals under normal grid operations) using convolutional layers. An encoder provides a mapping from the input to the latent space, while the decoder learns how to reconstruct the original input. An anomalous sample will be detected if its reconstruction error is higher than a threshold. By treating the input as images of detail coefficients, convolution will be useful in detecting large perturbations in frequencies.

IV. Proposed Method

This section provides details of the proposed W-CAE framework. Specifically, we focus on three aspects - discrete wavelet transform, data pre-processing, and convolutional autoencoder.

A. Discrete Wavelet Transform

DWT is used to convert a PMU signal $s(t)$ in the time domain to the joint time-frequency domain. The signal $s(t)$ is decomposed through the use of scaling functions, $\phi(t)$, and wavelet functions, $\psi(t)$, which act as low pass and high pass filters, respectively. Scaling functions filter out frequencies higher than half the highest frequency in the original signal and wavelet functions filter out lower frequencies [22]. Each level of decomposition outputs a set of approximation coefficients $a_i(n)$ and a set of detail coefficients $d_i(n)$, where i denotes the level of decomposition. Fig. 3 presents an example of detail coefficients obtained from PMU measurements during a $3-\phi$ line-to-line fault. The decomposition of $s(t)$ is represented as,

$$s(t) = \sum_{k} a_{j_0}(k)\phi_{j_0,k}(t) + \sum_{k}\sum_{j=j_0}^{\infty} d_j(k)\psi_{j,k}(t) \quad (1)$$

where j is the scale factor and k is the translation factor. Our model incorporates a sliding window component into wavelet analysis. Given a total period of time T, we define a window size M where the number of windows W is defined as $W = T - M + 1$. These windows are overlapping and let us capture patterns between contiguous time windows. We choose a window size that is small enough to pinpoint anomalies but large enough to avoid data loss caused by downsampling.

B. Data Pre-processing

The pre-processing stage transforms the DWT output into images. First, the output of the DWT algorithm is structured as a *multi-dimensional tensor* $\mathbf{A} \in \mathbb{R}^{E\times P\times C\times W\times D}$ where E is the number of simulations performed using the RTDS testbed, P is the number of PMUs on the 14-bus system, C is the number of PMU channels, W is the number of windows, and D is the number of detail coefficients. W is a tunable parameter, and $D = \left\lceil \frac{M}{2^K} \right\rceil$ is defined in terms of the window size M and the level of the decomposition K.

First, the DWT data undergoes min-max normalization. Next, in order to generate samples of 2-d images, we first slice the tensor $\mathbf{A}$ into smaller tensors that represent all of the channels for a single PMU. For each PMU p_i, we define a smaller matrix $\mathbf{B_i} \in \mathbb{R}^{C\times W\times D}$ that contains all of p_i's detail coefficients for a given simulation e_j. An aggregation of the detail coefficients is then performed that squeezes $\mathbf{B_i}$ into two dimensions. We consider mean and max magnitude aggregation as two heuristics. Compared to looking at the maximum magnitude, mean aggregation retains the highest variance in detail coefficients. An example is illustrated in Fig. 4. Note that $\mathbf{B_i}$ will have dimensions $C \times W$ after the aggregation. Given each p_i and their corresponding $\mathbf{B_i}$, we construct images by stacking each $\mathbf{B_i}$ so that the collection of $\mathbf{B_i}$'s gives one single matrix that contains all the aggregated values for each PMU's channels across time. Since $C << W$, we divide the W windows into smaller chunks with widths equal to C. A single chunk, or image sample, will be represented as $x \in \mathbb{R}^{PC\times PC}$. With the PMU data being sampled at a very high sampling frequency, each image will be small enough to allow for fast data processing, making our approach suitable for online stream processing.

C. Convolutional Autoencoder

Autoencoders are a feed-forward neural network primarily used for embedding input data into a latent space representation [23]. The basic structure of an autoencoder is composed of an encoder that maps input samples to a latent space and a decoder that reconstructs the encoded samples. A traditional autoencoder is built on hidden layers composed of neurons that contain activation functions such as ReLU or sigmoid functions [16]. A standard encoding of an input vector x into the latent space ϕ is represented as

$$\phi(\mathbf{x}) = h(\mathbf{W}\mathbf{x} + \mathbf{b}) \quad (2)$$

where $\mathbf{W}$ is a weight matrix, $\mathbf{b}$ is a bias vector, and $h(\cdot)$ denotes a non-linear activation function. The decoder, represented by $d(\phi)$, maps ϕ to a reconstructed vector $\hat{x}$ such that

$$d(\phi) = h(\mathbf{W}^*\phi + \mathbf{b}^*) \quad (3)$$

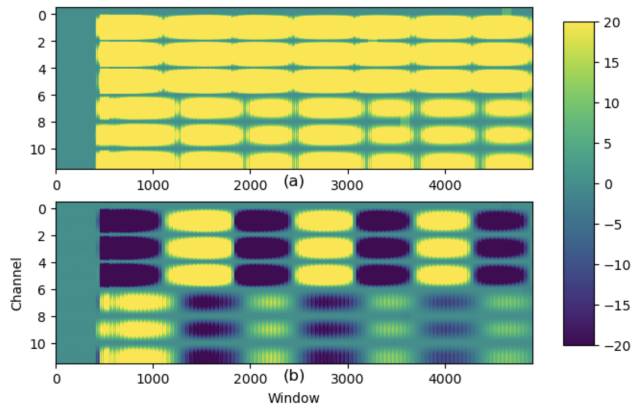

Fig. 4: (a) Max magnitude aggregation of detail coefficients. (b) Average aggregation of detail coefficients.

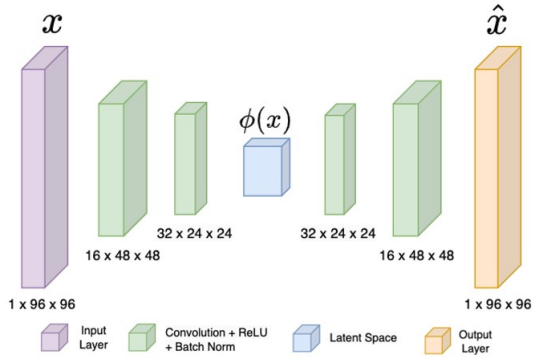

Fig. 5: Architecture of the proposed CAE.

where $\mathbf{W}^*$ and $\mathbf{b}^*$ represent the weight matrix and bias vector, respectively. In our encoder, there are two convolutional layers, each followed by a ReLU and Batch Normalization layer, as shown in Fig. 5. The decoder is composed of the same structure as the encoder. In each convolutional layer, re-sizing of images is performed by kernel filters, which detect features in local regions of an image. Let $n = PC$ be the height and width of x, and c be defined as the number of color channels in x. Function $\phi : \mathbb{R}^{c \times n \times n} \mapsto \mathbb{R}^{c^* \times m \times m}$ maps the input image of aggregated detail coefficients to a latent representation $\phi(x)$. Function $d : \mathbb{R}^{c^* \times m \times m} \mapsto \mathbb{R}^{c \times n \times n}$ then takes $\phi(x)$ as input and returns $d(\phi(x)) = \hat{x}$ as the reconstruction of x. The autoencoder learns the optimal ϕ that maps the inputs to an underlying manifold that is most representative of the input while also learning an optimal d that can reconstruct the samples. The training process, shown in Algorithm 1, involves minimizing the mean-squared error (MSE) between x and $\hat{x}$ which is defined as,

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2 \tag{4}$$

A high anomaly score (or high reconstruction error) above a threshold α will indicate the presence of an event. We optimize α to be a quantile of the distribution of reconstruction errors in the test set. If we define F to be the cumulative distribution function for the reconstruction errors, then the quantile function is defined by

$$F^{-1}(\beta) = \inf\{x : F(x) > \beta\} \tag{5}$$

In optimizing $\alpha = F^{-1}(\beta)$, we seek a β that provides high class separability. A low β will make the algorithm sensitive to anomalies; however, this may increase the number of false positives. Also, anomalies are localized in time with DWT.

Algorithm 1: Training Algorithm

Input: dataset $D = \{x_i\}_{i=1}^{N}$, learning rate η, batch size R, batches B, epochs S;
Output: Trained CAE Θ

```
/* Initialize the CAE model                    */
Θ := Build-CAE(η);
for s ∈ [1, ..., S] do
    for b ∈ [1, ..., B] do
        z := φ(x_b);
        x̂_b := d(z);
        MSE_b := (1/R) ||x̂_b − x_b||_2^2;
        /* Perform backpropagation             */
        Update-Weights(Θ);
    end
end
return Θ
```

V. Performance Evaluation

This section describes the results and discussion of the W-CAE framework for power system event detection. We focus on evaluating the model's classification accuracy, precision, recall, F1 score, and AUC. Ideally, the model needs to minimize the occurrence of false negatives due to the safety-critical nature of modern smart grids. We will also compare our approach, W-CAE, to a baseline approach, the Time-Convolutional Autoencoder (T-CAE), that does not implement DWT. The T-CAE approach contains the same CAE used in the W-CAE implementation; however, the image data will come directly from the original time series data generated by RTDS.

The machine learning models are built using `tensorflow` and DWT is implemented using `PyWavelets`. All experiments were performed on a Linux machine with an AMD processor and 32 CPU cores. A single NVIDIA Titan X GPU was used for the hyperparameter tuning and training.

A. Power System Data Generation

All simulations are performed on the IEEE-14 bus system which is modeled in the RTDS. The RTDS is capable of conducting electromagnetic transient simulations of a power system in real time and has several I/O and communication cards to facilitate external interfacing. The built-in GTNETx2-PMU firmware `PMU8` is used to obtain phasor measurements.

TABLE I: CAE Details

Type of Activation Layer	ReLU
Number of Convolutional Layers	4
Learning Rate	0.1
Optimizer	Adam
Learning Rate Decay	1×10^{-5}
Loss Function	MSE
Maximum Epochs	50
Batch Size	32

TABLE II: Evaluation Metrics

Model	Accuracy	Precision	Recall	F1	AUC	β
W-CAE	97.7	98.0	99.5	98.7	0.962	0.0645
T-CAE	89.4	90.2	98.8	94.4	0.460	0.01

A total of 8 PMUs are placed at buses $1, 2, 3, 4, 6, 9, 11$ and 13, which monitor lines $1-5, 2-1, 3-2, 4-5, 6-11, 9-14, 11-10$ and $13-12$. The PMU data includes three-phase voltage phasors, current phasors, frequency, and rate of change of frequency. For each PMU, 15 channels of data are obtained. A total of 5000 raw data points are obtained for each channel from the RTDS for 4 second. The total data size is 0.85 GB.

B. Experimental Settings

After experimentation, the `Haar` mother wavelet, decomposition level $K = 4$, and window size of 100 were chosen as parameters for performing DWT. Different $\beta \in [0, 1]$ are chosen to determine the value of α, which represents the classification threshold. The size of the training and test datasets for each model differ slightly due to data pre-processing. The W-CAE has a total of 3110 training samples and 3111 test samples, while the T-CAE has a total of 3120 training samples and 3172 test samples. An $80:20$ split on the training samples for each model produces training and validation sets. Training is unsupervised and is based on purity-approach. The loss on the validation set is calculated at the end of each epoch and the maximum number of epochs is set to 50. Early stopping is implemented based on validation loss and the loss function is optimized using the ADAM optimizer [24] with a decay of 1×10^{-5}. All hyperparameter optimization is done via grid search combined with cross-fold validation that chooses the optimal batch size, learning rate, and the number of training epochs. The details are summarized in Table I.

C. Experimental Results

Table II summarizes the evaluation results of the proposed W-CAE compared with the T-CAE. It is observed that W-CAE achieved higher performance on all classification metrics with $\beta = 0.645$. Although T-CAE achieves a high recall with $\beta = .01$, it fails to provide high precision and accuracy.

Fig. 6 shows the training and validation loss curves for the W-CAE and T-CAE. The W-CAE model performs very well in reconstructing training samples and only requires half of the total number of epochs in order for the validation loss to converge with the training loss. In Fig. 7, a confusion matrix demonstrates the ability of the W-CAE to classify nearly all true positives, thus leading to a high recall. Furthermore,

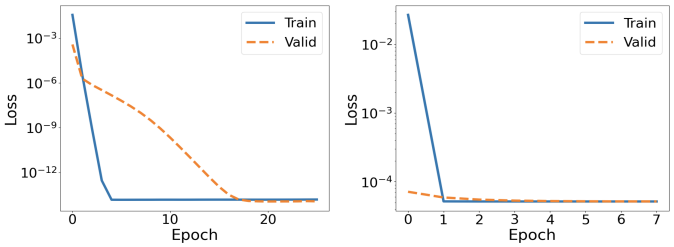

Fig. 6: Loss curves for (a) W-CAE and (b) T-CAE.

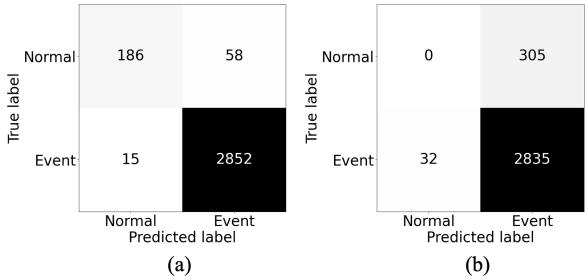

Fig. 7: Confusion matrices for (a) W-CAE and (b) T-CAE.

most of the true negatives are correctly classified, increasing precision and accuracy.

Fig. 8 presents the ROC curves for the W-CAE and T-CAE. T-CAE does not achieve a high AUC, indicating that it fails to provide good class separation, as compared to W-CAE, which is able to achieve an AUC closer to 1.0. Thus, T-CAE performs nearly as well as a random classifier, which would have an AUC = 0.50. This is due to the fact that T-CAE is unable to classify the true negative samples accurately.

The performance of T-CAE and W-CAE in analyzing time and space requirements was similar, with T-CAE requiring 14.6 s and W-CAE requiring 17.6 s on average for training, while both models only needed 1.245 GB of space (including the model, datasets, and GPU memory usage). Compared to the baseline CUSUM algorithm for change-point analysis, our model was much faster, as CUSUM required an average of almost 70 minutes to run on the 5856 signals in the test set, with each signal taking an average of 0.75 s to analyze.

The W-CAE also proves to be efficient for real-time streaming applications. The time complexity for DWT is $O(n)$, where n is the length of the input signal. For a window size of 100, performing DWT on a single PMU only takes 0.2 ms on average. The computation time can be further reduced by parallelizing across geographically clustered PMUs, making it fit for real-time processing. Note that the prediction takes negligible time and all training is offline. In summary, features extracted from the detail coefficients using DWT are more informative of events than time series signals. Thus, W-CAE outperforms T-CAE in event detection.

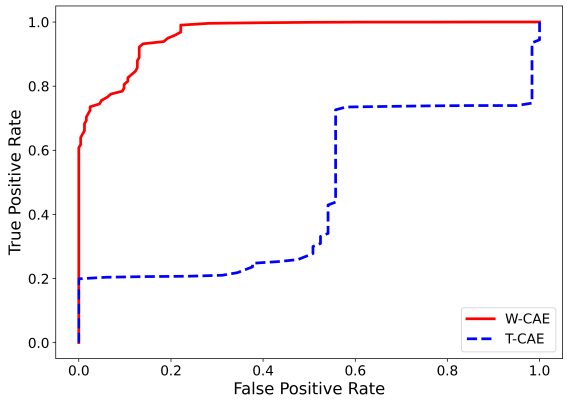

Fig. 8: ROC curve plotted at various values of α.

VI. Conclusion

In this paper, we present an unsupervised power system event detection algorithm using wavelet transform and a convolutional autoencoder. We demonstrate that incorporating discrete wavelet transform as a feature extractor prior to anomaly detection proves to be more effective than relying on time-series phasor data. Our approach leveraged coefficients obtained from wavelet decomposition of phasor data, transforming them into an image, and using an unsupervised deep learning method for event detection and localization. Our approach has high accuracy and provides a better sense of class separability compared to a baseline method as observed by a high AUC of 0.962, a low training time of 17.6 s, and reasonable space requirements of 1.245 GB, making it suitable for online applications. The proposed method was validated on real-time phasor measurements obtained from a real-time digital simulator for different power system events. In the future, we will consider additional events such as cyber-attacks and evaluate other deep learning models used for handling time series such as recurrent neural networks and transformers.

References

[1] H. Haes Alhelou, M. E. Hamedani-Golshan, T. C. Njenda, and P. Siano, "A survey on power system blackout and cascading events: Research motivations and challenges," *Energies*, vol. 12, no. 4, p. 682, 2019.

[2] J. De La Ree, V. Centeno, J. S. Thorp, and A. G. Phadke, "Synchronized Phasor Measurement Applications in Power Systems," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 20–27, Jun. 2010, conference Name: IEEE Transactions on Smart Grid.

[3] Y. V. Makarov, P. Du, S. Lu, T. B. Nguyen, X. Guo, J. W. Burns, J. F. Gronquist, and M. A. Pai, "PMU-Based Wide-Area Security Assessment: Concept, Method, and Implementation," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1325–1332, Sep. 2012, conference Name: IEEE Transactions on Smart Grid.

[4] S. Wang, P. Dehghanian, and L. Li, "Power grid online surveillance through pmu-embedded convolutional neural networks," *IEEE Transactions on Industry Applications*, vol. 56, no. 2, pp. 1146–1155, 2020.

[5] S. Basumallik, R. Ma, and S. Eftekharnejad, "Packet-data anomaly detection in pmu-based state estimator using convolutional neural network," *International Journal of Electrical Power & Energy Systems*, vol. 107, pp. 690–702, 2019.

[6] T. Wu, W. Xue, H. Wang, C. Y. Chung, G. Wang, J. Peng, and Q. Yang, "Extreme Learning Machine-Based State Reconstruction for Automatic Attack Filtering in Cyber Physical Power System," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 1892–1904, Mar. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9055170/

[7] R. Ma, S. Basumallik, and S. Eftekharnejad, "A pmu-based data-driven approach for classifying power system events considering cyberattacks," *IEEE Systems Journal*, vol. 14, no. 3, pp. 3558–3569, 2020.

[8] T. Wu, Y.-J. Angela Zhang, and X. Tang, "Online detection of events with low-quality synchrophasor measurements based on $i$forest," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 168–178, 2021.

[9] L. Xie, Y. Chen, and P. R. Kumar, "Dimensionality reduction of synchrophasor data for early event detection: Linearized analysis," *IEEE Transactions on Power Systems*, vol. 29, no. 6, pp. 2784–2794, 2014.

[10] H. Li, Y. Weng, E. Farantatos, and M. Patel, "An unsupervised learning framework for event detection, type identification and localization using pmus without any historical labels," in *2019 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2019, pp. 1–5.

[11] D. Senaratne, J. Kim, and E. Cotilla-Sanchez, "Spatio-temporal frequency domain analysis of pmu data for unsupervised event detection," in *2021 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*. IEEE, 2021, pp. 1–5.

[12] S. Pandey, A. K. Srivastava, and B. G. Amidan, "A real time event detection, classification and localization using synchrophasor data," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4421–4431, 2020.

[13] T. Lan, Y. Lin, J. Wang, B. Leao, and D. Fradkin, "Unsupervised power system event detection and classification using unlabeled pmu data," in *2021 IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*. IEEE, 2021, pp. 01–05.

[14] A. Ahmed, S. K. Sadanandan, S. Pandey, S. Basumallik, A. K. Srivastava, and Y. Wu, "Event analysis in transmission systems using spatial temporal graph encoder decoder (stged)," *IEEE Transactions on Power Systems*, 2022.

[15] Y. Cheng, N. Yu, B. Foggo, and K. Yamashita, "Online power system event detection via bidirectional generative adversarial networks," *IEEE Transactions on Power Systems*, vol. 37, no. 6, pp. 4807–4818, 2022.

[16] A. Ahmed, V. V. G. Krishnan, S. A. Foroutan, M. Touhiduzzaman, C. Rublein, A. Srivastava, Y. Wu, A. Hahn, and S. Suresh, "Cyber physical security analytics for anomalies in transmission protection systems," *IEEE Transactions on Industry Applications*, vol. 55, no. 6, pp. 6313–6323, 2019.

[17] J. Wu, Z. Zhao, C. Sun, R. Yan, and X. Chen, "Fault-attention generative probabilistic adversarial autoencoder for machine anomaly detection," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7479–7488, 2020.

[18] Z. Huang, P. Du, D. Kosterev, and S. Yang, "Generator dynamic model validation and parameter calibration using phasor measurements at the point of connection," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1939–1949, 2013.

[19] P. Wang, Z. Zhang, Q. Huang, W. Zhang, and W.-J. Lee, "Pmu based problematic parameter identification approach for calibrating generating unit models," *IEEE Transactions on Industry Applications*, vol. 57, no. 5, pp. 4520–4527, 2021.

[20] C.-C. Tsai, L.-R. Chang-Chien, I.-J. Chen, C.-J. Lin, W.-J. Lee, C.-C. Wu, and H.-W. Lan, "Practical considerations to calibrate generator model parameters using phasor measurements," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2228–2238, 2017.

[21] S. Jenkins, J. Eto, and J. Dagle, "Transmission Innovation Symposium: Modernizing the U.S. Electrical Grid," Tech. Rep. None, 1825004, ark:/13030/qt3z521027, Jun. 2021. [Online]. Available: https://www.osti.gov/servlets/purl/1825004/

[22] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[23] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1127647

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. [Online]. Available: https://arxiv.org/abs/1412.6980