Using Deep Learning for Detecting Mirroring Attacks on Smart Grid PMU Networks

Yusuf Korkmaz*, Alvin Huseinovic†, Halil Bisgin*, Saša Mrdović†, Suleyman Uludag*
*Dept. of Computer Science, The University of Michigan - Flint, Email: {ykorkmaz, bisgin, uludag}@umich.edu
†Fac. of Elec. Eng., U. of Sarajevo, Bosnia & Herzegovina, Email: {ahuseinovic, ssasa.mrdovic}@etf.unsa.ba

***Abstract*—Similar to any spoof detection systems, power grid monitoring systems and devices are subject to various cyberattacks by determined and well-funded adversaries. Many well-publicized real-world cyberattacks on power grid systems have been publicly reported. Phasor Measurement Units (PMUs) networks with Phasor Data Concentrators (PDCs) are the main building blocks of the overall wide area monitoring and situational awareness systems in the power grid. The data between PMUs and PDC(s) are sent through the legacy networks, which are subject to many attack scenarios under with no, or inadequate, countermeasures in protocols, such as IEEE 37.118-2. In this paper, we consider a stealthier data spoofing attack against PMU networks, called a mirroring attack, where an adversary basically injects a copy of a set of packets in reverse order immediately following their original positions, wiping out the correct values. To the best of our knowledge, for the first time in the literature, we consider a more challenging attack both in terms of the strategy and the lower percentage of spoofed attacks. As part of our countermeasure detection scheme, we make use of novel framing approach to make application of a 2D Convolutional Neural Network (CNN)-based approach which avoids the computational overhead of the classical sample-based classification algorithms. Our experimental evaluation results show promising results in terms of both high accuracy and true positive rates even under the aforementioned stealthy adversarial attack scenarios.**

***Index Terms*—CNN, Deep Learning, PMU, PMU Spoofing, PMU forged data, Mirroring Attack.**

I. Introduction

The volume and intensity of cyberattacks are increasing against all computational systems, including especially the critical infrastructure by the highly determined, focused, and well funded adversaries. The smart grid, enhanced traditional power grid with computational and communications improvements, is not an exception [1] to the ever growing attack vectors, as it is an attractive target [2] with lethal and vital economic and social consequences by means of disruption to electricity delivery [3]. World Economic Forum's 2018 report [4] emphasizes the increasing cyberattacks on the critical and strategic infrastructure that may result in disrupting the society. Addressing these critical security and privacy assurances in the smart grid is emerging as an urgency as a result [5]–[12].

An important enabler of the Smart Grid initiatives is the enhanced use of sensing and measurement capabilities. Phasor Measurement Units (PMUs) are the advanced, accurate, and synchronized measurement devices to take the situational awareness to a new level. While the traditional Supervisory Control And Data Acquisition (SCADA) measurements are taken every 2-4 seconds, PMU reports them 30-120 times per second with GPS time stamps. The PMU-enabled conceptual model of wide-area monitoring, protection, and control subsystem is illustrated [12] in Figure 1.

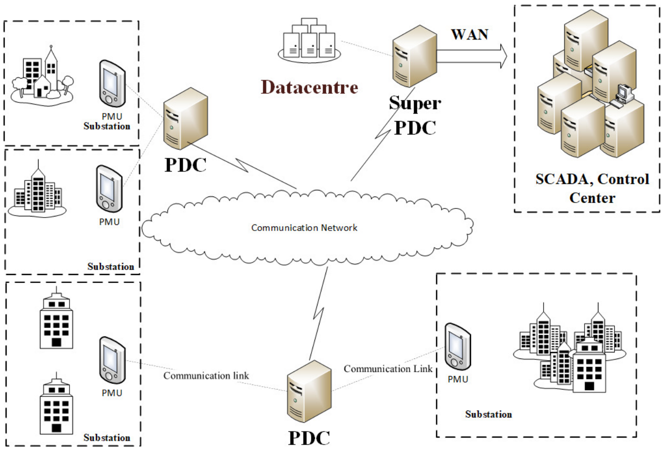

Fig. 1. A conceptual framework for a wide-area monitoring, protection, and control (WAMPAC) system for the Smart Grid made possible by PMUs [13].

PMUs transmit data to the Phasor Data Concentrator (PDC) by using IEEE 37.118-2 synchrophasor protocol. Data received at PDC is then used for state estimation or historical analysis. It is relatively more recently recognized that the PMU data, especially over the IEEE 37.118-2 protocol, which has no security mechanisms [14], has many vulnerabilities [15], [16], such as transport layer attacks [17], data tampering attacks [18], etc.

In this paper, we consider a PMU data collection network where the threat environment assumes a compromised PMU injecting spoofed data into the network to corrupt, or disrupt or confuse the state estimation and the situational awareness of the overall power grid. We focus on a specific attack, mirroring attack, with both plain from the literature and novel stealthier strategies, whose detailed definition are given in Section III. As a countermeasure, we develop a novel frame-based transformation of the data to invoke efficient classification through 2D Convolutional Neural Network (CNN) with augmentation. Our performance evaluation results provide promising results in terms of accuracy and true positive rates even under the enhanced attack scenarios in terms of the strategy and the lower intensity of forged packets.

The rest of the paper is organized as follows: Section II presents the related work towards spoofing cyber attacks in the Smart Grid. Section III focuses on the PMU data spoofing with the threat model, dataset, and attack vector details. The Section IV introduces the machine learning methodologies for detecting the PMU spoofing in terms of the mirroring attack together with the countermeasures using frame-based 2D Convolutional Neural Network (CNN) approach under a novel enhanced attack scenario. Section V includes the experimentation setup together with the simulations and the discussion of these results. Concluding remarks and future work ideas are given in Section VI.

II. Related Work

Table I shows the most common spoofing attacks on PMU data: repeat last value attack [19], [20], time dilation attack [19]–[24], mirroring attack [19], [20], [23], play back attack [21], [22], data drop attack [21], [22] polynomial fit attack [23], and general false data injections attack [21], [25], [26].

There defensive approaches both for intentional attacks by adversaries and unintentional faults in the system can be summarized as follows: [19], [20] use SVM and Artificial Neural Network (ANN) to detect anomalies relying on the highly-correlated inter-PMU and intra-PMU parameters. [23], [24] focus on SVM.

In [21], authors artificially create their datasets and use Recurrent Neural Networks (RNN) and Long short-term memory neural network (LSTM) to detect False Data Injection Attacks (FDI) against PMU based state estimators. Same authors in [22] employ Symbolic Aggregation Approximation in data preprocessing phase. For detection of these attacks, text mining using Bag of Pattern and Multivariate Bag of Pattern are used and compared. The feature extraction is obtained through Principal Component Analysis (PCA).In [25] authors detect FDI attacks using Rule Based Autoregressive Moving Average (ARMA) and Autoregressive Integrated Moving Average (ARIMA) applied on calculations based on Kirchoffs laws.

	Repeat Last Value	Time Dilation	Mirroring	Playback	Data Drop	Polynomial Fit	False Data Injection
[19]	✓	✓	✓				
[20]	✓	✓	✓				
[21]		✓		✓	✓		✓
[22]		✓		✓	✓		
[23]		✓	✓			✓	
[24]		✓					
[25]							✓
[26]							✓
[27]	✓						

TABLE I
The most common PMU attacks from the literature.

In [28] authors use different types of clustering approaches to identify fault events in the power grid. The selected fault events are divided into: single-line-to-ground faults, line-to-line faults, three-phase faults and no-fault data. They use two different clustering approaches. The first is time series clustering that uses hierarchical clustering for which they claim to be the most appropriate in case of time series data. The other clustering method is instantaneous clustering that uses is based on k-means and Density Based Spatial Clustering of Applications with Noise (DBSCAN). In [29] authors use k-nearest neighbor (KNN), binary SVM, multi-SVM and Decision Trees (DT) to detect events based either on event zone or event type. This approach requires field knowledge in order to correctly apply event labeling on the PMU data.

In [30] authors are completely agnostic to the data being transmitted between PMUs and PDC. Instead of checking the validity of the data they use k-means clustering to separate the network traffic not typical for PMU to PDC communication.

In [31] authors consider single-phase, two-phase and three phase types of faults. They also consider short circuits and ground for each individual phase. They applied Linear Discriminant Analysis (LDA), kNN, SVM and ANN machine learning approaches on simulated IEEE 123-bus distribution system.

In [26] phasor measurement unit data attacks (PMUDA) by using different machine learning algorithms that can be used for supervised and semi-supervised learning. The supervised machine learning algorithms are multi-layer perceptron (MLP), SVM, KNN, AdaBoost+, C4.5 DT and XGBoost. The semi-supervised learning techniques are deep autoencoders (DA) and one-class SVM (OC-SVM).

In [32] authors use Generative Adversarial Networks (GAN) and Neural ordinary differential equations (NODE) to artificially generate PMU data events. They observe three event types: Bus Fault, Line Tripping and Load shedding. The simulation environment includes a 10-machine IEEE 39 bus system. To classify events they use PCA and Discrete Wavelet Transformation(DWT) SVM kernels. In [33] authors use SVM with online learning in order to predict short-term voltage instability on IEEE 39 bus based network.

III. PMU Data Spoofing

A. Threat Model

We consider a PMU network subsection from Figure 1 where multiple PMUs send sensed data to a PDC. Assuming a compromised PMU, an attacker aims at disrupting the monitoring subsystem by means of falsifying data stealthily. Our focus in this paper is on a specific spoofing technique, called mirroring attack (MA), as introduced in [23], [24] and used also used in [19], [20]. MA is a type of attack where the adversary selects the last valid data in a sequence of n measurements and replays them in reversed order, as shown in Figure 2. MA is designed to be local, i.e. on a specific PMU, based on historical PMU data and to cause no discontinuity of the measurements. These attack characteristics makes detection harder by design.

B. Dataset

We make use of the École Polytechnique Fédérale de Lausanne (EPFL) dataset [34], [35], collected from their own campus transmission network with 7 PMUs collecting the following information: time stamp (epoch time format), fraction of the seconds in msec, latency in sec, rate of change of frequency in Hz/s, and magnitudes of three phase voltages

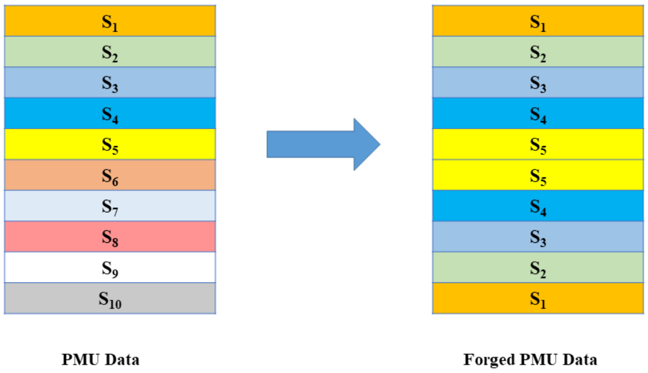

Fig. 2. Mirroring Attack: The original sequence of the PMU data on the left is forged on the right by reverse replaying S_5 through S_1 in place of the original S_6 through S_{10}, respectively.

and currents with phase angles in radians. In later sections, we will be comparing our approach to that of [20] and [19], whose data (Bonneville BPA) is not publicly available. However, from the descriptions in [20] and [19], we have concluded that both BPA and EPFL data are similar except for frequency, 50 versus 60 Hz which does not make a change in the analysis.

C. Attack Vector and Scenarios

In this work, we use a 24-hour block of EPFL from March 15, 2019 and assume that only one PMU out of seven is under the mirroring attack. We generate the simulated MA by means of two different scenarios by changing the number of attacks (NOA) or spoofed readings for each hour NOA_h: (i) 50+ attack, (ii) perfect attack.

In the first attack scenario, we randomly select time points (t_a) in each hour where we start spoofing with a minimum number of attacks (min_{NOA}) set to 50. The maximum number of attacks (max_{NOA}) after this time point is also randomly generated. Once we determine these boundaries, we start mirroring r_{NOA} readings starting from t_a by overwriting those many samples, where $min_{NOA} = 50 \leq r_{NOA} \leq max_{NOA}$. Therefore, we coin this scenario as 50+ to indicate that at least 50 samples are mirrored for every t_a once the attack is initiated. We continue to spoof at least 50 samples at different times until we hit the hourly attack count, NOA_h. Once we spread all mirroring attacks for all hours with the same principle, one data set becomes ready to be used in a prediction model. It is worth noting that we keep track of our earlier attacks so that later attacks do not cause distortion on the previously designated mirrored signals due to randomization.

While the 50+ scenario incorporates several random features to create a stealthier attack, our second scenario, termed the *perfect attack*, relaxes the randomization by removing min_{NOA} and max_{NOA} from the parameter list. Instead, selects time points,$\frac{t_a}{2}$, with the same NOA_h in mind, but guarantees that $t_a \in \mathbb{Z}$. In other words, at every half second it mirrors first 25 readings so that remaining 25 readings in the same second time frame become mirror "image" of the first half. Since there are 50 samples in a second and first half is exactly mirrored within the same time frame, we call this setup as "perfect scenario".

In both attack models, NOA_h is changed from 500 to 3,000 with the increments of 500, corresponding to 0.3% to 1.7%, respectively with highly unbalanced spoofed packets to simulate realistic stealthy attacks. Furthermore, we note that our scenarios have significantly less number of attacks compared to earlier studies [19], [20] which not only have a higher percentage of spoofed packets ($\sim$ 50%), but also follow a regular and simpler pattern, such as spoofing the second half of every second without any randomness. Hence, we tackle a much more challenging attack vector in this study.

IV. Mirroring Attack Detection Methodology

A. Frame Approach

By the very definition of the mirroring attack, it duplicates readings and inserts them at different points in the time series data. As a result, conventional classification methods face additional challenge of differentiating these identical/duplicated values. In order to avoid this complicated dichotomy, we aggregate measurements into one second frames of data and apply the classification at that coarser granularity level, i.e. at frame-level versus individual data points in the conventional methods. While we may have to be discarding some good observations in a frame as a result of a few bad observations, the overall impact of these deletions of good observations is not expected to be significant due to the statistical nature of the time series data; it is very well-known fact that omissions of individual values will not have a significant impact on the statistical parameters of the overall system.

Our frame-based approach turns the problem into an image classification task through a transformation from samples to image-like frames as illustrated in Figure 3. In particular, it takes P PMUs for N variables with a sampling rate, S, and results in T frames that are $S \times \underbrace{P \times N}$ in size. In our case, out of 24-hour measurements (4,320,000 samples) we generate 86,400 frames or images that come with a size of 50×63 for $S = 50$, $P = 7$, and $N = 9$.

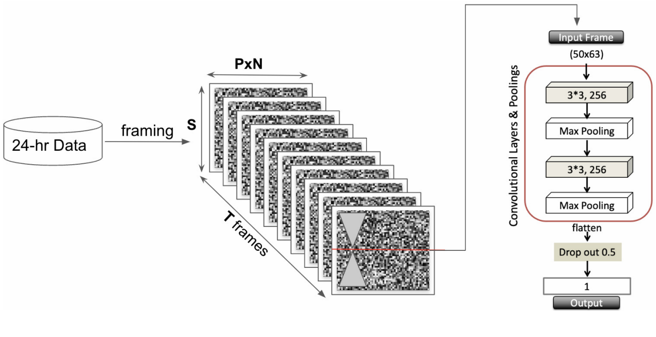

Fig. 3. Model development pipeline using CNN architecture on transformed data. Each pixel represent a cell in our data set. Some frames have mirrored images.

Even though we do not have real images in our frames, we treat every single reading as a pixel value and our goal is to determine whether a frame hosts mirroring pixels (attacks) in the current context. In other words, we assume mirrored pixels (a time slice in one PMU) form an object in a given image, and the reflection of the object creates a pattern to help us identify attacked frame.

One major challenge of this approach for 50+ scenario is that reflections of our imaginary objects may overflow to the adjacent frame. Since we spread the attacks in a random fashion, some frames may end up with a incomplete portions of mirrored attacks. This not only poses a challenge to a classifier with the absence of expected pattern, but also calls frames with very few mirrored samples spoofed. In the perfect scenario, however, we still incorporate randomization for the timing of the attack, but we guarantee that spoofed frames have perfect reflections of imaginary objects without any overflows.

B. *Machine Learning Algorithms*

Considering our frame-based transformation of the classification problem, we exploit a deep neural network (DNN) architecture specialized for images; 2-dimensional convolutional neural network (CNN). enhanced through data augmentation to handle frame overflow mentioned earlier.

1) 2D CNN: CNN is a special type of Artificial Neural Network (ANN) or DNN because of its multiple layers including convolutional and pooling layers [36], [37]. CNN's advantage comes from the fact that it learns and extracts image features through its convolutional and pooling layers which are then utilized for classification [38]. Therefore, CNNs are very effective deep learning (DL) models for image classification and object detection [39].

Our network architecture has seven layers, as shown in Figure 3. The first layer is a 2D convolutional layer with a kernel size of 3x3 and 256 filters with the activation function of rectified linear unit (RLU). The second layer is a pooling layer with a 2x2 pool size. The third and forth layers are 2D convolutional and pooling layers that follow the same configuration. The sixth is the flattening and the seventh is the dropout layer using a drop rate of 0.5. Finally, the output is a dense layer with sigmoid activation function.

2) Data Augmentation: Data Augmentation (DA) is a technique to create artificial variations of existing images to enrich a data set and increase its size by using several transformation methods such as crop, shift, rotation, and flip [40]. It also helps reduce overfitting [41]. To handle shifting duplicates due to randomization, we use the shift option of data augmentation in Keras [42].

We train our models on 6 hours of data and test on the remaining 18 hours on the same day. Further, we incorporate data augmentation into perfect scenario for a stealthier attack. Namely, we expose perfect scenario to data augmentation during the training stage to prepare it for any future variations of mirroring attacks which might be found in 50+ attacks. Then, we test our data-augmented perfect attack model on 24 hours, 50+ attack data.

V. Experimentation

A. *Simulation Setup*

The simulations are run on a server with 128GB RAM and two Intel Xeon Silver 4208 processors. Each processor has 8 cores and 16 threads which gives a total of 32 simultaneous threads running on the training and testing processes.

We explore four different train-test combinations which show not only the impact of the attack ratio, but also the improvement in predictions as we incorporate different techniques, as shown in Figure 4 and detailed below: (1) Train on 6 hours of 50+ attack, test on the remaining 18 hours, (2) Train on 6 hours under the perfect attack, test on the remaining 18 hours, (3) Train on 6 hours under the perfect attack, test on the full 50+ attack, and (4) Train on 6 hours under the perfect attack with augmentation, test on the full 50+ attack.

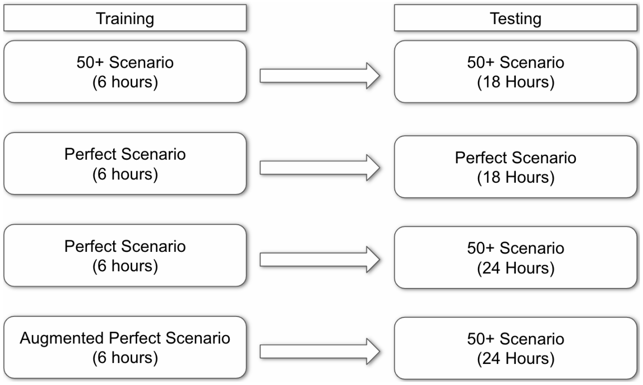
Training
Testing
50+ Scenario (6 hours)
50+ Scenario (18 Hours)
Perfect Scenario (6 hours)
Perfect Scenario (18 Hours)
Perfect Scenario (6 hours)
50+ Scenario (24 Hours)
Augmented Perfect Scenario (6 hours)
50+ Scenario (24 Hours)

Fig. 4. Four different test-train combinations for the simulations.

B. *Impact of Attack Ratio*

We start our simulations with 50+ scenario with very small number of hourly attacks (500) corresponding to 0.3% of 24-hour measurements. Then, we continue increasing hourly attack counts with 500 increments until we spoof 40% of the data so that our attack ratio can be comparable with earlier studies [19], [20] in terms of the spoofed packet percentage. To highlight the impact of the attack ratio, we develop a CNN model for each data set varying with different ratio based on 6 hours of data, and test it for the remaining hours. Since accuracy calculations are dominated by high true negative rates (TNRs) due to the highly unbalanced data sets, we shed light on true positive rates (TPRs) to flag spoofed time segments. In Figure 5, we observe that our CNN models start to perform better after an attack ratio of 5% and hit 98% TPR for 40% attack ratio for the 50+ attack that is still less than that of aforementioned studies. Figure 5 clearly illustrates that when the attack ratio is high, as used in [19], [20], detection is rather trivial. Our focus in this paper is on the less dense and stealthier attack cases with lower percentages.

When we focus on stealthier attacks for 50+ simulations we calculate the average TPR and its standard deviations to observe the effect of random initialization on the performance and measure the robustness of DL models in detecting the spoofed data. Figure 6 shows an increasing pattern for 50+ attack (bottom-most green curve, corresponding to Fig. 4's first scenario), but not good enough to serve as a reliable predictor with very low TPR rates.

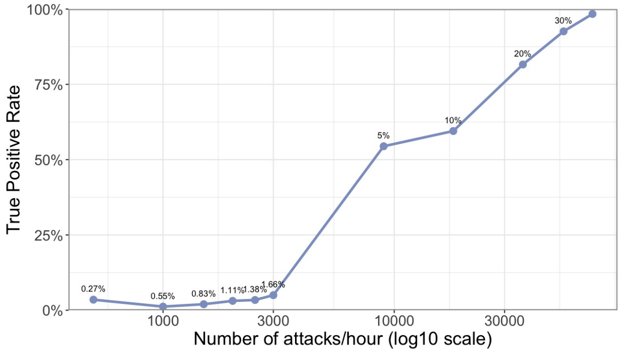

Fig. 5. True Positive Rate (TPR) as a function of increasing attack percentage for the 50+ attack scenario, the first combination from Fig. 4.

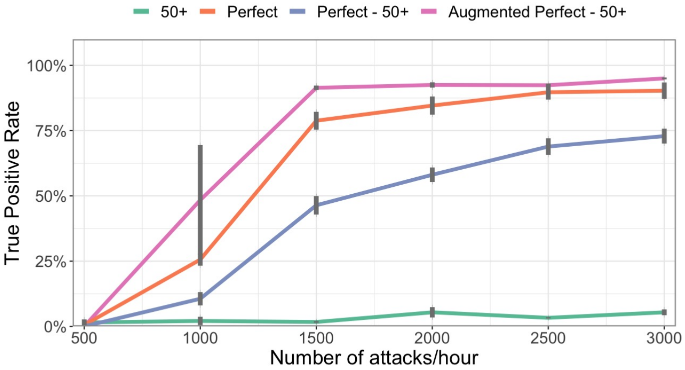

Fig. 6. TPR as a function of of increasing attack percentage for the four different combinations from Fig. 4, respectively.

C. Perfect Scenario

As we noted earlier perfect scenario relies on an ideal mirroring for a given frame and we aim to determine those frames through a CNN model again. Thus, in this second simulation from Figure 4, we follow the 6 vs. 18-hour split for training and testing which have perfect hourly attacks on a random basis. In Fig. 6 we observe that models developed on perfect scenario (second curve from the bottom) show an improvement over 50+ scenario with increasing TPRs as attack ratio goes up.

D. Utilization of Perfect Scenario for 50+

In the first two simulations, we observe that 50+ attack scenario cannot be handled effectively and the model developed on perfect scenario has relatively higher TPR. In this step (the third from Fig. 4), therefore, we attempt a hybrid model where a model is trained on perfect scenario and utilized on a different attack scenario, i.e., 50+. In doing so, we aim to teach a model how the overall pattern looks like and test its capability on a stealthier attack. We find out that this approach can detect spoofed frames with higher TPRs when compared to the first pipeline where a model on 50+ scenario attempt to predict other 50+ attacks. It should be noted that for extremely stealthy attacks this approach still suffers, but when considering a test case over the course 24 hours it performs 40-60% better than earlier case which can be inferred from Figure 6.

We further exploit DL features for our CNN models to incorporate data augmentation. We augment the model that relies on perfect scenario. Thus, our model learns possible (vertical) shifts of the perfect mirroring attacks in a given second and can extrapolate shifted readings which emulates overflowing attacks to consecutive frames. When we test this approach (the last one from Fig. 4) on the 50+ scenario data set, we observe further improvements in TPRs that are presented in Figure 6. While this approach cannot achieve more than other models for the hourly attack count of 500, which is very stealthy, it reaches 80% TPR in the next hourly attack count, 1000, which is still very low in percentage. As expected, it performs better as we increase the hourly attack count. This phenomenon can also be observed in Figure 6 where the TPR reaches 91% with standard deviation less than 0.01 which also shows the robustness of the model. We observe one exception for 1000 attack count where we have higher standard deviation which might be due to the random nature of the attacks.

VI. Conclusion

As part of the smart grid enhancement to fully take advantage of the communications and computational advancements, the attack surface is expanding, especially by the highly motivated adversaries and the attractiveness of the severe consequences of such attacks. As a critical monitoring and sensing of the smart grid, the protection of PMU networks becomes more important than ever. In this paper, we have introduced a stealthier mirroring attack, for the first time in the literature to the best of our knowledge. To counter such an attack, we proposed a novel computationally efficient frame-based 2D CNN approach. Our preliminary experimental evaluation results show promising results in terms of accuracy and true positive rates. One potential future research direction is to explore other sample-based machine learning classification techniques to compare against our 2D CNN approach. Finally, a detailed study of the computational and spatial overhead of the our approach against the others would be an interesting extension of the current work.

References

[1] "Surviving a Catastrophic Power Outage," The President's National Infrastructure Advisory Council (NIAC), Tech. Rep., Dec.

[2] M. P. Barrett, "Framework for Improving Critical Infrastructure Cybersecurity Version 1.1," National Institute of Standards and Technology (NIST), Tech. Rep., apr.

[3] N. Kshetri and J. Voas, "Hacking Power Grids: A Current Problem," *Computer*, no. 12, pp. 91–95, dec.

[4] "The Global Risks Report 2018 13th Edition," World Economic Forum, Tech. Rep.

[5] S. Systems, P. McDaniel, and S. McLaughlin, "Security and Privacy Challenges in the Smart Grid," *Security & Privacy, IEEE*, vol. 7, no. 3, pp. 75–77, 2009.

[6] H. Khurana, M. Hadley, N. Lu, and D. A. Frincke, "Smart-grid security issues," *IEEE Security and Privacy*, vol. 8, pp. 81–85, 2010.

[7] A. R. Metke and R. L. Ekl, "Security Technology for Smart Grid Networks," *IEEE Trans. on Smart Grid*, no. 1, pp. 99–107, jun.

[8] W. Wang and Z. Lu, "Cyber security in the Smart Grid: Survey and challenges," *Computer Networks*, no. 5, pp. 1344–1371.

[9] J. Liu, Y. Xiao, S. Li, W. Liang, and C. L. P. Chen, "Cyber Security and Privacy Issues in Smart Grids," *Communications Surveys Tutorials, IEEE*, vol. 14, no. 4, pp. 981–997, 2012.

[10] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu, "Securing smart grid: cyber attacks, countermeasures, and challenges," *Communications Magazine, IEEE*, vol. 50, no. 8, pp. 38–45, 2012.

[11] Y. Xiao, *Security and Privacy in Smart Grids*. Taylor & Francis.

[12] A. Huseinović, S. Mrdović, K. Bicakci, and S. Uludag, "A survey of denial-of-service attacks and solutions in the smart grid," *IEEE Access*, vol. 8, pp. 177 447–177 470, 2020.

[13] V. Terzija, G. Valverde, P. Regulski, V. Madani, J. Fitch, S. Skok, M. M. Begovic, and A. Phadke, "Wide-Area Monitoring, Protection, and Control of Future Electric Power Networks," *Proc. of the IEEE*, no. 1, pp. 80–93, jan.

[14] S. M. S. Hussain, S. M. Farooq, and T. S. Ustun, "A security mechanism for ieee c37.118.2 pmu communication," *IEEE Tran. on Industrial Electronics*, vol. 69, no. 1, pp. 1053–1061, 2022.

[15] C. Tu, X. He, X. Liu, and P. Li, "Cyber-attacks in pmu-based power network and countermeasures," *IEEE Access*, vol. 6, pp. 65 594–65 603, 2018.

[16] R. Khan, K. Mclaughlin, D. Laverty, and S. Sezer, "Design and implementation of security gateway for synchrophasor based real-time control and monitoring in smart grid," *IEEE Access*, vol. 5, pp. 11 626–11 644, 2017.

[17] Y. Wang, T. T. Gamage, and C. H. Hauser, "Security implications of transport layer protocols in power grid synchrophasor data communication," *IEEE Tran. on Smart Grid*, vol. 7, no. 2, pp. 807–816, 2016.

[18] M. N. Aman, K. Javed, B. Sikdar, and K. C. Chua, "Detecting data tampering attacks in synchrophasor networks using time hopping," in *IEEE ISGT-Europe*. IEEE, 2016, pp. 1–6.

[19] J. Jiang, X. Liu, S. Wallace, E. Cotilla-Sanchez, R. Bass, and X. Zhao, "Defending against adversarial attacks in transmission- and distribution-level pmu data," 2020.

[20] J. Jiang, "Defending against adversarial attacks in electric power systems: A machine learning approach," *Washington State University*, no. 1, 2019.

[21] 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.

[22] 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.

[23] J. Landford, R. Meier, R. Barella, S. Wallace, X. Zhao, E. Cotilla-Sanchez, and R. B. Bass, "Fast sequence component analysis for attack detection in smart grid," in *5th SMARTGREENS*, 2016, pp. 1–8.

[24] X. Liu, S. Wallace, X. Zhao, E. Cotilla-Sanchez, and R. B. Bass, "Episodic detection of spoofed data in synchrophasor measurement streams," in *IEEE IGSC*, 2019, pp. 1–8.

[25] B. Chen, S. i. Yim, H. Kim, A. Kondabathini, and R. Nuqui, "Cybersecurity of wide area monitoring, protection, and control systems for hvdc applications," *IEEE Tran. on Power Systems*, vol. 36, no. 1, pp. 592–602, 2021.

[26] J. Wang, D. Shi, Y. Li, J. Chen, H. Ding, and X. Duan, "Distributed framework for detecting pmu data manipulation attacks with deep autoencoders," *IEEE Tran. on Smart Grid*, vol. 10, no. 4, pp. 4401–4410, 2019.

[27] A. Huseinovic, Y. Korkmaz, H. Bisgin, S. Mrdovic, and S. Uludag, "PMU Spoof Detection via Image Classification Methodology against Repeated Value Attacks by using Deep Learning," in *28th Int'l Conf. on Inf., Comm. and Automation Tech (ICAT)*, Sarajevo, Bosnia and Herzegovina, 2022.

[28] E. Klinginsmith, R. Barella, X. Zhao, and S. Wallace, "Unsupervised clustering on pmu data for event characterization on smart grid," in *5th SMARTGREENS*, 2016, pp. 1–8.

[29] A. Shahsavari, M. Farajollahi, E. M. Stewart, E. Cortez, and H. Mohsenian-Rad, "Situational awareness in distribution grid using micro-pmu data: A machine learning approach," *IEEE Tran. on Smart Grid*, vol. 10, no. 6, pp. 6167–6177, 2019.

[30] P. Donner, A. S. Leger, and R. Blaine, "Unsupervised machine learning for anomaly detection in synchrophasor network traffic," in *2019 North American Power Symposium (NAPS)*, 2019, pp. 1–6.

[31] F. L. Grando, A. E. Lazzaretti, M. Moreto, and H. S. Lopes, "Fault classification in power distribution systems using pmu data and machine learning," in *2019 20th Int Conf on Int Sys App to PS (ISAP)*, 2019, pp. 1–6.

[32] X. Zheng, B. Wang, D. Kalathil, and L. Xie, "Generative adversarial networks-based synthetic pmu data creation for improved event classification," *IEEE Open Access Journal of Power and Energy*, vol. 8, pp. 68–76, 2021.

[33] "Pmu-based voltage stability prediction using least square support vector machine with online learning," *Electric Power Systems Research*, vol. 160, pp. 234–242, 2018.

[34] EPFL. Epfl campus pmu dataset.

[35] M. Pignati, M. Popovic, S. Barreto, R. Cherkaoui, G. Dario Flores, J.-Y. Le Boudec, M. Mohiuddin, M. Paolone, P. Romano, S. Sarri, T. Tesfay, D.-C. Tomozei, and L. Zanni, "Real-time state estimation of the epfl-campus medium-voltage grid by using pmus," in *IEEE ISGT*, 2015, pp. 1–5.

[36] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1–6.

[37] U. Michelucci, *Advanced Applied Deep Learning*. Apress, 2019.

[38] C. Zheng, D.-W. Sun, and L. Zheng, "Recent developments and applications of image features for food quality evaluation and inspection–a review," *Trends in Food Science & Technology*, vol. 17, no. 12, pp. 642–655, 2006.

[39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[40] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[41] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

[42] F. Chollet *et al.*, "Keras," https://keras.io, 2015.