Locational Detection of the False Data Injection Attack in a Smart Grid: A Multilabel Classification Approach

Shuoyao Wang, Suzhi Bi, *Senior Member, IEEE*, and Ying-Jun Angela Zhang, *Fellow, IEEE*

***Abstract*—State estimation is critical to the monitoring and control of smart grids. Recently, the false data injection attack (FDIA) is emerging as a severe threat to state estimation. Conventional FDIA detection approaches are limited by their strong statistical knowledge assumptions, complexity, and hardware cost. Moreover, most of the current FDIA detection approaches focus on detecting the presence of FDIA, while the important information of the exact injection locations is not attainable. Inspired by the recent advances in deep learning, we propose a deep-learning-based locational detection architecture (DLLD) to detect the exact locations of FDIA in real time. The DLLD architecture concatenates a convolutional neural network (CNN) with a standard bad data detector (BDD). The BDD is used to remove the low-quality data. The followed CNN, as a multilabel classifier, is employed to capture the inconsistency and co-occurrence dependency in the power flow measurements due to the potential attacks. The proposed DLLD is "model-free" in the sense that it does not leverage any prior statistical assumptions. It is also "cost-friendly" in the sense that it does not alter the current BDD system and the runtime of the detection process is only hundreds of microseconds on a household computer. Through extensive experiments in the IEEE bus systems, we show that DLLD can perform locational detection precisely under various noise and attack conditions. In addition, we also demonstrate that the employed multilabel classification approach effectively enhances the presence-detection accuracy.**

***Index Terms*—Convolutional neural network (CNN), false data injection attack (FDIA), multilabel classification, power system, security, state estimation.**

Manuscript received October 8, 2019; revised February 2, 2020 and March 6, 2020; accepted March 22, 2020. Date of publication March 27, 2020; date of current version September 15, 2020. This work was supported in part by the National Key Research and Development Program under Project 2019YFB1803305, in part by the National Natural Science Foundation of China under Project 61871271, in part by the General Research Funding established by Hong Kong Research Grant Council under Project 14200315, in part by the Guangdong Province Pearl River Scholar Funding Scheme 2018, in part by the Foundation of Shenzhen City under Project JCYJ20170818101824392 and Project JCYJ20190808120415286, and in part by the Science and Technology Innovation Commission of Shenzhen under Project 827/000212. *(Corresponding author: Suzhi Bi.)*

Shuoyao Wang and Suzhi Bi are with the College of Electronic and Information Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: w.shuoy@gmail.com; bsz@szu.edu.cn).

Ying-Jun Angela Zhang is with the Department of Information Engineering, Chinese University of Hong Kong, Hong Kong (e-mail: yjzhang@i.e.cuhk.edu.hk).

Digital Object Identifier 10.1109/JIOT.2020.2983911

I. Introduction

A. Motivations and Summary of Contributions

THE POWER system is a fundamental economic-social infrastructure. The recent trends of Industrial Internet-of-Things (IIoT) technology have profoundly transformed the conventional power system in the last decades. In particular, the latest advances in smart grids extensively integrate the advanced information and communication technology (ICT) [1] with the conventional power system, which significantly increase the grid efficiency and reliability. However, the new ICT systems employed by smart grids as well as other IIoT networks are facing great security challenges especially under the mounting threats of cyberattacks. State estimation, which calculates the state of the power network system from the raw measurements gathered by the supervisory control and data acquisition (SCADA) system [2], plays a very essential role in the control center. In particular, compromised system state estimation may interfere the operation of power systems, since many power system applications (such as economic dispatch, contingency analysis, etc.) rely on the results of state estimation [3]. Liang *et al.* [4] and Deng *et al.* [5] presented comprehensive surveys on the impacts of cyberattacks on state estimation, e.g., line congestion [6], power outage [7], communication block [8], etc.

Among the existing cyberattacks, the false data injection attack (FDIA) [9] is targeted at compromising power system state estimation by injecting false data into meter measurements. A well-structured FDIA can circumvent the conventional bad data detector (BDD) in today's SCADA system, and thus is recognized as one of the most challenging threats to state estimation. For example, the work in [6] demonstrated the economical effect of false data injection by causing transmission line congestion. Various research has been developed to investigate possible ways of constructing FDIA [5]. For instance, a stealthy attack was introduced in [9], which shows how this type of false data can pass the BDD in the control center. It is shown in [10] that an undetectable attack is also possible even if the attacker has partial configuration information of the power network, and can only manipulate a small set of the power network measurements.

At the same time, much research effort has been devoted to defending against FDIA, which is broadly classified into two categories, namely, physical-based defending strategies [10]–[12] and data-dependent detection

algorithms [13]–[23]. For example, Yang *et al.* [11] computed the minimum number of sensors that must be compromised to manipulate and developed an effective algorithm for optimal PMU placement to defend against FDIA. Gai *et al.* [12] introduced dynamic programming to produce an optimal solution of maximizing privacy protection levels for resource-constrained devices. Alternatively, various data-dependent algorithms have been proposed to investigate the FDIA detection problem, e.g., mixture Gaussian distribution methods [13], maximum-likelihood estimation [14], Kalman filters [15], sparse optimization [16], network theory [17], and similarity matching [18]. For instance, Ashok *et al.* [18] leveraged load forecast information, generation schedules, and synchrophasor data to obtain a statistical characterization of the variation between SCADA-based state estimates and forecast-based predictions to detect anomalies. However, the effectiveness of most of the existing work highly depends on the knowledge of the attack model and power system information (e.g., the assumption of the near chordal sparsity of the power grids). Recently, data-driven detection methods based on deep learning have been proposed. Instead of deriving an algorithm from a predefined attack and power system model, deep learning methods allow the system to learn the models directly from training data. A summary of recent work is provided in Section I-B. However, to the best of our knowledge, all the existing methods only focus on detecting the presence of attacks, i.e., whether there exists a malicious attack. In practice, other than presence detection, it is essential to identify the location of the attack for the fast deployment of effective countermeasures. In addition, the inconsistency and co-occurrence dependency captured by location identification provides additional room for enhancing the presence-detection performance.

To bridge the gap, in this article, we consider a deep-learning-based mechanism to achieve the locational detection of FDIA. In particular, we formulate the FDIA locational detection problem as a multilabel classification problem. To solve this problem, we propose an architecture that concatenates a convolutional neural network (CNN) with a standard BDD detector. The standard BDD detector is used to remove the low-quality data. The followed CNN, as a multilabel classifier, is employed to capture the inconsistency and co-occurrence dependency introduced by FDIA. The architecture is practical in the sense that it is model-free and requires no alternation of the current BDD system. Moreover, the runtime of the detection process is only hundreds of microseconds on a household computer. In summary, our main contributions are detailed as follows.

1) To the best of our knowledge, this article is among the first to develop a deep-learning-based locational detection mechanism of FDIA in the power system. In particular, the proposed algorithm, referred to as DLLD, concatenates a deep neural network with a standard BDD detector. With updated network parameters, the DLLD architecture can adapt to the variation of the underlying attack and topology models.
2) To extract power flow correlation features and improve the locational detection performance, we formulate the FDIA locational detection problem as a multilabel classification problem and employ CNN as the classifier. We carefully design the network structure (e.g., render up pooling layers) and loss function according to the unique structures of the FDIA locational detection problem.
3) We carry out extensive evaluations to verify and analyze the proposed framework with open-source data and code. The parameter sensitivity test is also performed to evaluate the performance and generalization ability of the proposed mechanism. By the way of illustration, results in the IEEE 118-bus system show that the proposed DLLD on average achieves 93.18% locational detection accuracy and 99.1% presence-detection accuracy. Overall, we conclude that DLLD is a scalable robust mechanism with high accuracy.

B. Related Work

There have been some existing FDIA detection methods using machine learning technologies. For instance, Ganjkhani *et al.* [19] used a nonlinear autoregressive exogenous network to identify the injected bad data in state estimation and generate FDIAs, where less than 10% of the generated FDIAs are detected by conventional processors. Ozay *et al.* [20] proposed semisupervised and online learning algorithms to detect FDIA. The algorithms can be employed in hierarchical and topological networks for different attack scenarios. Both supervised and unsupervised learning methods were proposed in [21] to distinguish the stealthy FDIA and safe operation modes. He *et al.* [22] used the conditional deep belief network (CDBN) to reveal the high-dimensional temporal behavior features of the stealthy FDIAs, and successfully detected 90% of the FDIAs. Most recently, Yu *et al.* [23] employed a discrete wavelet transform to reveal the spatial data characteristics and deep neural network to capture the temporal data correlations for FDIA detection. Therein, the detection accuracy is improved from around 70% by Kalman filters [15] to more than 90%. Overall, all the existing works focused only on detecting the presence of the attack. In contrast, the FDIA locational detection considered in this article shares some similarities with the multilabel classification problem in image processing and speech recognition [24], [25].

Krizhevsky *et al.* [26] proposed a deep CNN for image classification, consisting of convolutional layers, pooling layers, and fully connected layers. A variant of this model won the ILSVRC-2012 competition, and CNNs have received skyrocketed interest in both academia and industry since then. More recently, CNNs have played a crucial role in AlphaGo [27] which beats the best human player in Go games. One main advantage of utilizing a CNN for image multilabel classification is that the semantic structure of multilabel images can be better preserved [24]. In this article, we intend to capture the inconsistency and co-occurrence dependency on multiple measurements introduced by FDIA, which has a similar flavor of semantic structures in multilabel images classifications. We, therefore, apply CNN as the core module for locational detection of FDIA in this article.

The remainder of this article is organized as follows. We briefly introduce the conventional state estimation method and

its vulnerability against FDIA in Section II. In Section III, we illustrate the architecture and implementation issues on the proposed FDIA locational detection mechanism. The simulation results with parameter sensitivity are presented in Section IV. Finally, this article is concluded in Section V.

II. Preliminary

A. Power System State Estimation

State estimation is to infer the operational state of a power system from the available meter measurements in an SCADA system. In this article, we focus on the state estimation in dc microgrids, which are widely deployed due to the advantages over their ac counterparts, including higher reliability, simpler control, and more efficient interfacing with renewable energy sources and energy storage units [28]. In a dc model, the relationship between the n-dimensional measurement $\boldsymbol{z} = (z_1, z_2, \ldots, z_n)^T$ and the m-dimensional system state $\boldsymbol{x} = (x_1, x_2, \ldots, x_m)^T$ can be expressed as

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e} \tag{1}$$

where $\boldsymbol{e} = (e_1, e_2, \ldots, e_n)^T$ and $\boldsymbol{H}$ denote the measurement noise and Jacobian matrix, respectively.

The conventional BDD approaches compare the ℓ_2-norm of the measurement residual with a threshold τ to check whether there exists a low quality, i.e., bad or compromised, measurements [9]. In this case, the detector announces the presence of an attack as long as

$$R = \|\boldsymbol{z} - \boldsymbol{H}\boldsymbol{x}\|_2^2 \geq \tau. \tag{2}$$

B. False Data Injection Attack

The objective of FDIA is to mislead the system operator into considering a compromised state estimate $\hat{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{c}$ as a valid estimation, where $\boldsymbol{c} \neq \boldsymbol{0}$ is the deviation of the power system state. To achieve this, an attacker changes the received measurements at the control center to $\hat{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{a}$, where $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)^T$ is the compromised attack vector. Then, the observation model can be described as

$$\hat{\boldsymbol{z}} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e} + \boldsymbol{a}. \tag{3}$$

In general, an unstructured $\boldsymbol{a}$ is likely to be identified by the traditional BDD in (2). To circumvent the BDD mechanism, the attack vector should be structured, such as $\boldsymbol{a} = \boldsymbol{H}\boldsymbol{c}$. In such cases, the ℓ_2-norm of the residual is unchanged

$$\left\|\hat{\boldsymbol{z}} - \boldsymbol{H}\hat{\boldsymbol{x}}\right\| = \|\boldsymbol{z} + \boldsymbol{a} - \boldsymbol{H}(\boldsymbol{x} + \boldsymbol{c})\| = \|\boldsymbol{z} - \boldsymbol{H}\boldsymbol{x}\| \tag{4}$$

and thus the attack can bypass the BDD. Accordingly, the power system operator would mistake $\boldsymbol{x} + \boldsymbol{c}$ for a valid estimate, and thus an error vector $\boldsymbol{c}$ is introduced.

In reality, getting access to all information of $\boldsymbol{H}$ comes at an unbearable cost and effort for attackers because the information is kept confidential and highly secured. In practice, constructing a successful stealthy FDIA may only require compromising a small number of meters [10], [29], even with partial knowledge of system parameters. For instance, Bi and Zhang [10] proved that an optimal stealthy attack that minimizes the attackers' resource cost can be constructed efficiently through solving a min-cut problem, when the attacker only has limited knowledge of $\boldsymbol{H}$.

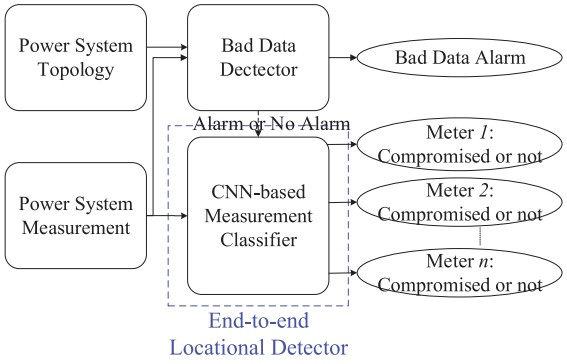

Fig. 1. Proposed FDIA locational detection mechanism.

In this article, we develop a new data-driven mechanism that can detect the location of FDIA in a SCADA system. It is formulated as a multilabel classification problem that determines whether each meter measurement is compromised. The problem is formulated and solved in Section III.

III. Locational Detection: Multilabel Classification Approach

In this article, we propose a multilabel classification mechanism using the recent advances in deep learning technologies to capture the inconsistency and co-occurrence dependency caused by FDIA. In this section, we first elaborate on the structure of the proposed mechanism. Then, we present the detailed implementation of the mechanism.

A. Locational Detection

Mathematically, to detect the existence of FDIA is equivalent to classifying the whole measurement vector, i.e., $\boldsymbol{x}$, into two categories: 1) exist or 2) not. This is a single-label classification problem from the perspective of machine learning, whereas to identify the location of the attack is equivalent to classifying each element of the measurement vector, i.e., x_i, into two categories. That is, the locational detection problem is a multilabel classification problem from the perspective of machine learning. Although deep learning technologies have achieved great success in single-label classification over the past decade, multilabel classification is still attracting much research interest due to its complexity and wide applicability. Unlike single-label classification, multilabel classification problems can be evaluated with a multitude of quality measures, often conflicting in nature. Besides, the labels of multilabel classification problems are usually extremely unbalanced, and thus the single-label balance methods (e.g., downsampling) do not work. To address the problem, we carefully design the CNN structure in Fig. 2 to extract and describe the associated data information to produce satisfactory performances in multilabel classification. In addition, we will also evaluate the improvements over the original single-label methods in our numerical experiments.

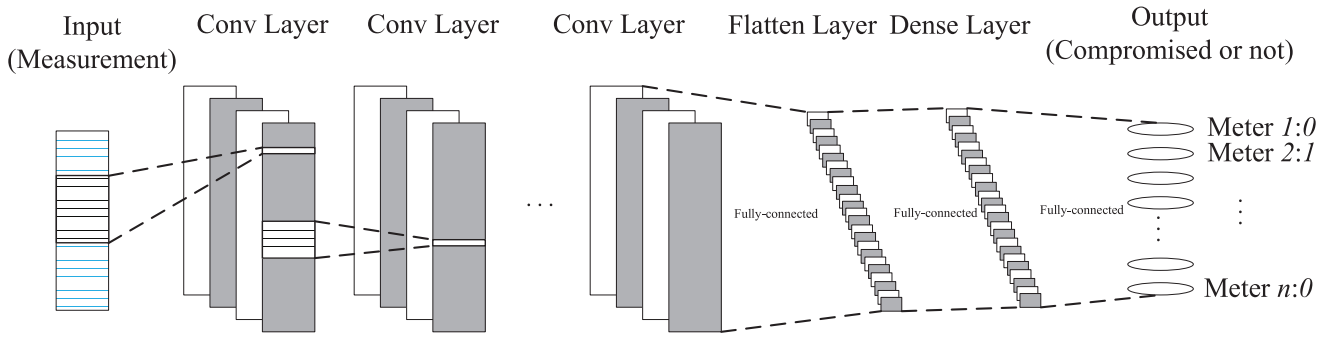

Fig. 2. Architecture of 1-D deep CNN for DLLD. 1: compromised meter; and 0: uncompromised meter.

B. Proposed Mechanism

The proposed FDIA locational detection mechanism is depicted in Fig. 1. The proposed framework receives measurements from consecutive discrete sampling time instances, i.e., the time instances when the conventional state estimation takes place.[1] This, together with the fact that the training process of the CNN classifier only requires measurements and ground-truth labels and confirms that the proposed mechanism does not leverage any prior statistical assumptions (e.g., $\boldsymbol{H}$). At a sampling time instance t, the input data (the real-time measurement) first goes through the BDD detector. As described in (2), BDD evaluates the quality of the measurement data by calculating the ℓ_2-norm of the measurement residual and comparing with a predetermined threshold τ. BDD reports the current meter as compromised or noisy if $R \geq \tau$.[2] By doing so, the sampling and communication errors as well as potential unstructured FDIA can be effectively detected, because of their high residual values [5]. If the measurement data pass the BDD, a CNN-based multilabel classifier will detect the presence and location of structured FDIAs by analyzing the inconsistency and co-occurrence dependency of the data.

The proposed DLLD scheme employs a CNN to extract and analyze the high-dimensional temporal features of FDIA.

1) Data: We denote the input (i.e., the measurements), the ground-truth labels (i.e., the meter classes), and the output (i.e., the classification of the CNN at time t) as $\boldsymbol{z}^t = (z_1^t, \ldots, z_n^t)$, $\boldsymbol{y}^t = (y_1^t, \ldots, y_n^t)$, and $\hat{\boldsymbol{y}}^t = (\hat{y}_1^t, \ldots, \hat{y}_n^t)$, respectively. For example, in our numerical experiments in Section IV, the dimensions of input and output data are both 19 for the IEEE 14-bus system, because there are 19 measurements inside the 14-bus system in our simulation settings. The ground-truth label of meter i at time t is determined according to the following rule:

$$y_i^t = \begin{cases} 1, & \text{the meter } i \text{ at time } t \text{ is compromised} \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

The output of CNN $\hat{y}_n^t$'s is continuous numbers between 0 and 1. Correspondingly, the classifier defines a discrimination threshold to quantify the outputs to 0 or 1. The discrimination threshold can be adjusted to increase or decrease the sensitivity to application factors. Unless specified otherwise, the discrimination threshold is set to 0.5 in this article following the common practice.

2) Architecture: The architecture of the deep CNN for FDIA locational detection is shown in Fig. 2. It contains an input layer, several convolutional layers, one flattening layer, one fully connected hidden layer, and an output layer. The input layer has n input numbers representing the n measurements at each time instance. Each filter in the first convolutional layer is applied to the windows in the input layer to generate features through the convolution operation, batch normalization, and nonlinear transformation with the rectified-linear unit (ReLU) activation function [31]. The feature maps $c_{1,j}$ of the first convolutional layer generated from the input data $\boldsymbol{z}$, which can be expressed as

$$c_{1,j} = \text{ReLU}\big(\boldsymbol{z} * \boldsymbol{h}_{1,j} + b_{1,j}\big). \tag{6}$$

Here, $\boldsymbol{h}_{1,j}$ is the jth convolution kernel, which is essentially a 1-D filter,[3] and $b_{1,j}$ is the corresponding scalar bias. In (6), a scalar bias $b_{1,j}$ is added to all the convolution output, which is a commonly used representation in deep learning [31]. The convolution operation is denoted by $*$ in (6) and the output at position i is defined as

$$\sum_{k=1}^{l_{1,j}} \big(\boldsymbol{h}_{1,j}\big)[i] \times (\boldsymbol{z})\left[i - k + \frac{l_{1,j}}{2}\right]. \tag{7}$$

Here, $l_{1,j}$ and $\times$ denote the length of the filter $\boldsymbol{h}_{1,j}$ and the inner product operation, respectively.

The hidden features generated by filters in the $(q-1)$th convolutional layer are then used as the input to the qth convolutional layer and processed in a similar way. The output can be written as

$$c_{q,j} = \text{ReLU}\big(\boldsymbol{c}_{q-1} * \boldsymbol{h}_{q,j} + b_{q,j}\big) \tag{8}$$

where $c_{q,j}$ is the jth feature map at the qth convolutional layer. The number of filters in each layer and depth of convolutional layers are hyperparameters, which will be further discussed in the simulation section. The extracted features learned by the last convolutional layer, i.e., the $q^{\max}$th convolutional layer, are merged into one single vector in the flatten layer and fed into

[1]The sample rates range from 100 Hz (burst mode recording) to hourly readings for SCADA systems [30].

[2]Following the common practice [22], the selection of the value of the threshold τ is numerically studied and selected as 10 in this article.

[3]1/2/3-D: one/two/three-dimensional.

TABLE I
DEEP-LEARNING-BASED LOCATIONAL DETECTION (DLLD) NETWORK FOR THE IEEE 14-BUS SYSTEM. TOTAL PARAMETERS: 247 827; TRAINABLE PARAMETERS: 246 675; AND NONTRAINABLE PARAMETERS: 1152

Stage	Type	Kernel	Output Size	Parameters
0	Input	-	19×1	0
1	Conv	5×1	19×128	768
2	BatchNorm	-	19×128	512
3	LeakyRELU	-	19×128	0
4	Conv	3×1	19×256	98560
5	BatchNorm	-	19×256	1024
6	LeakyRELU	-	19×256	0
7	Conv	3×1	19×128	98432
8	BatchNorm	-	19×128	512
9	LeakyRELU	-	19×128	0
10	Conv	3×1	19×64	24640
11	BatchNorm	-	19×64	256
12	LeakyRELU	-	19×64	0
13	Flatten	-	1216×1	0
14	FullyConn	-	19×1	23123
15	Sigmoid	-	19×1	0

a fully connected hidden layer (also known as dense layer) with the activation function ReLU. That is

$$c_{F,j} = \text{ReLU}\left(\boldsymbol{w}_F \times \boldsymbol{c}_{q^{\max}} + b_F\right) \tag{9}$$

where $c_{F,j}$, $\boldsymbol{w}_F$, and $\boldsymbol{b}_F$ denote the feature maps, weights, and biases of the flatten layer, respectively. The nodes in the dense layer are also fully connected to n nodes in the output layer. The sigmoid function is applied to the nodes in the output layer to classify the type of each measurement. For meter j at time t, the final multilabel classification result $\hat{y}_j^t$ is

$$\hat{y}_j^t = \text{sigmoid}(\boldsymbol{w}_D \times \boldsymbol{c}_F + b_D) \tag{10}$$

where $\boldsymbol{w}_D$ and b_D denote the weights and biases of the dense layer, respectively.

Overall, we plot Table I to show an example of the DLLD network for the IEEE 14-bus system.

Remark 1: Besides the convolutional layers, pooling and dropout layers are also important components in ordinary CNN architectures. However, they do not appear in our design for the following reasons. First, pooling layers are normally used for the downsampling of high-dimensional computation, e.g., 2-D-convolution and 3-D-convolution computations. In our problem, all the convolutional layers are 1-D-convolutional layers whose computation under GPU programming is quite efficient. Second, traditionally, pooling layers are one of the main factors to achieve nonlinear mapping in deep CNN. However, the popular ReLU active function also introduces nonlinearity in deep models. Therefore, rendering up pooling layers sometimes achieves even a better performance, because useful details may be discarded by pooling layers [32]. Third, dropout is a widely used technique to control overfitting. Meanwhile, the proposed DLLD already has employed the mini-batch methodology for overfitting control, which intentionally introduces sufficient noise to each gradient update. Indeed, we have tested the performance of pooling and dropout, and found that they do not provide any performance gain.

C. Training

Before using the proposed FDIA locational detection scheme to classify the measurements, we need to first optimize the learning parameters, i.e., the filters $\boldsymbol{h}$, weights $\boldsymbol{w}$, and biases $\boldsymbol{b}$, in each layer. This parameter tunning process is called training, which aims to find the optimal parameters that match the input and output in the training data.

1) Mini-Batch and Cross-Validation: To enhance the convergence rate and avoid overfitting, we adopt the mini-batch gradient descent method to train the network. In our simulations, each mini-batch contains 200 instances of data. In each iteration, a fixed number of training samples, i.e., a mini-batch, are randomly selected from the training set to calculate the gradient. Following the common practice in machine learning, we separate 7/10 data into the training set and 3/10 data into the validation set for each batch. Then, fitting is done using the Adam optimizer with an initial learning rate of 0.001 and a patience of 5.

2) Loss Function: To find the optimal learning parameter set, we introduce a loss function to measure the difference between the actual output and the ground-truth output among each mini-batch. To extend our framework to multilabel classification, the loss function of the proposed CNN is chosen as the cross-entropy function. In particular, the cross-entropy loss function over a mini-batch $\theta = \{t_1, \ldots, t_{200}\}$ is expressed as

$$\text{cross-entropy}(\theta) = \sum_{t\in\theta} -\frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i^t \log\left(y_i^t\right) + \left(1-\hat{y}_i^t \log\left(1-y_i^t\right)\right)\right). \tag{11}$$

With clearly defined loss function, we can adopt the Adam [31] optimizer to find the optimal parameters given a mini-batch θ.

IV. EXPERIMENTS

In this section, we first present the training and testing data generation step by step in Section IV-A. In Section IV-B, we introduce the implementation details and benchmark algorithms. Then, we show detailed examples to demonstrate how the proposed method captures the inconsistency and co-occurrence dependency introduced by FDIA on nearby measurements in Section IV-C. The detection accuracy and robustness of locational and presence detection are investigated in Sections IV-C and IV-D, respectively.

A. Data Set

In this section, we assess the performance of the proposed FDIA locational detection mechanism in the IEEE 14- and 118-bus power systems. The topologies of the grid can be obtained from MATPOWER [33] and summarized in Table II. The measurements of meters located at adjacent lines or buses are highly correlated. Besides, CNN obtains the features by analyzing the meter measurements of adjacent indices. Therefore, we index the meter measurements based on the network topology. In this article, we first index the line flow meters from $q = 1$ as follows: 1) we index the unindexed meters connecting bus q and set $q = q + 1$ and 2) if

TABLE II
STATISTICS OF IEEE 14- AND 118-BUS POWER TEST SYSTEMS

No. of buses	14-bus	118-bus
No. of lines	20	186
No. of measurements	19	180
No. of inject measurements	8	70
No. of flow measurements	11	110
No. of unmeasured lines	2	7

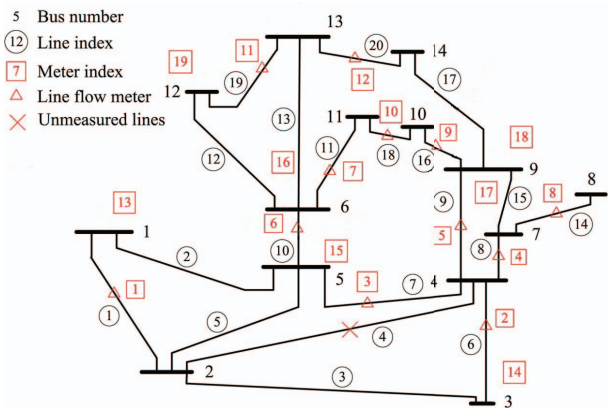

Fig. 3. Indexed IEEE 14-bus system.

$q > 14(118)$, we terminate the index process; otherwise, the policy turns back to 1). Then, we continue the index from line meters and label the injection meters based on the ascending order of the bus index. For illustration purpose, an indexed measurement placement of the IEEE 14-bus system is plotted in Fig. 3. The measurement placement and indices for the 118-bus system are omitted for the simplicity of expositions. The complete source code implementing DLLD and data sets is available at https://github.com/wsyCUHK/WSYCUHK_FDIA.

1) Base Load: We first generate uncompromised data by extending the real-world data through artificially generating the loads on each bus. The generated loads follow a normal distribution whose mean is equal to the baseload and standard variance is equal to 1/6 of the value of base load [14], [34]. Besides, we also generate compromised data. As mentioned in Section I, there are two categories of FDIAs, namely, well-structured FDIA and unstructured FDIA. Unstructured FDIAs can be precluded by the conventional BDD process inside our DLLD framework. The system will consider them as faulty measurements and discard them directly. Hence, we only generate well-structured FDIA.

2) Attack Implementation: Due to the limited budgets of attackers, we generate the compromised data based on the min-cut FDIA model with partial network knowledge in [10]. More specifically, the optimal partial knowledge attack is the one who requires the minimum cost of obtaining the knowledge of a particular transmission line impedance. Without loss of generality, the system parameters are generated as follows.

1) The number of target state variables follows a discrete uniform $(2, 5)$ distribution in the 14-bus system and a discrete uniform $(2, 10)$ distribution in the 118-bus system, respectively.
2) The cost of obtaining the knowledge of a particular transmission line impedance is set in the same way as in [10].

The ℓ_2-norm of the injection data varies from 1 to 5 in Fig. 5 and is set as 1 in all other experiments.

3) Measurement Noise: Last but not least, noticing that there exists an inevitable dynamic noise in measurement and communication processes, we also append random Gaussian noises to the measurement values. In particular, the noise standard derivation varies from 0.1 to 0.5 in Fig. 5 and is set as 0.2 in all other experiments.

4) Training and Testing Data Set: Under each level of attack and noise, we generate a training set that contains 10 000 compromised instances and 100 000 instances without any injection. We generate another ten independent testing data sets that contain 500 compromised time instances and 500 instances without any injection for performance evaluation. Each value presented in this section is averaged over ten independent testing sets. The detailed generation process and data sets are available at https://github.com/wsyCUHK/WSYCUHK_FDIA.

B. Implementation Details

All simulations are conducted on a machine with an Intel Xeon E5-2630 CPU, two nVidia GTX 1080 GPUs, and 64-GB RAM. The multilayer perceptron network (MLP) and CNN are constructed using Keras [35] for a computational speed boost. For benchmark methods, we compare the proposed scheme with the state-of-the-art methods, including support vector machine (SVM), light gradient boosting machine (LightGBM), and deep-learning-based identification (DLBI) [22].[4]

1) *DLBI:* He *et al.* [22] proposed a CDBN architecture to extract high-dimensional temporal features. The CDBN detects the FDIA by analyzing the temporal attack patterns that are presented by the real-time measurement data from the geographically distributed meters.
2) *SVM:* SVM [36] is a maximum margin classifier that constructs a hyperplane(s) in a high-dimensional space. It is widely used since it achieved top performance in some classification problems (e.g., text spam and images) in the 1990s.
3) *LightGBM:* LightGBM [37] is a gradient boosting framework that uses tree-based learning algorithms published by Microsoft. LightGBM is being widely used in many winning solutions of machine learning competitions.

Hyperparameter (e.g., the number of convolutional layers and the number of filters) tuning is done using a random search strategy, where we select the model that assigns the highest F_1-Score (defined in Section IV-C) to the validation data.

C. Locational Detection Performance

1) Performance Evaluation Metrics: In our experiments, we employ the precision and recall of the generated outputs

[4]The open-source implementation of SVN and LightGBM can be find in https://github.com/soloice/SVM-python and https://github.com/Microsoft/LightGBM, respectively.

as performance evaluation metrics. The precision and recall are defined as

$$\text{precision} = \frac{\text{True Positive Rate}}{\text{True Positive Rate} + \text{False-Positive Rate}} \quad (12)$$

and

$$\text{recall} = \frac{\text{True Positive Rate}}{\text{True Positive Rate} + \text{False-Negative Rate}} \quad (13)$$

respectively. In this article, true positive rate (TPR), false-positive rate (FPR), and false-negative rate (FNR) are defined as the probability that a compromised location is labeled as compromised, an uncompromised location is labeled as compromised, and an uncompromised location is labeled as uncompromised, respectively. To strike a balance between the precision and recall, we also strike the F_1-score. In particular, F_1-score is the geometrical average of the precision and recall, and is expressed as

$$F_1\text{-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (14)$$

Moreover, we introduce row accuracy (RACC) as an evaluation metric. RACC is defined as the probability that all the uncompromised locations in the grid are labeled as uncompromised and all the compromised locations are labeled as compromised.

We first evaluate the proposed method when the ℓ_2-norm of the injection data is 2 and the standard deviation of the measurement noise is 0.2. We compare the proposed mechanism not only with the state-of-the-art methods: SVM and LightGBM but also with an alternative of the proposed mechanism, where the CNNs in our location and detection mechanism is replaced by the MLPs. Accordingly, the proposed mechanism and the MLP alternative are named as DLLD and MLP-DLLD, respectively. In particular, the number of hidden layers in the MLP varies from 2 to 6 and the number of units is selected with the highest F_1-Score. To guarantee a fair comparison, we use the same data sets for the training and testing procedure of all four methods.

2) IEEE 14-Bus System: First, we compare the four metrics among SVM, LightGBM, and MLP-DLLD with a different number of hidden layers, and DLLD with a different number of hidden layers in the IEEE 14-bus system in Table III. Overall, DLLD outperforms the three benchmark algorithms in both F_1-Score and RACC, which justifies the effectiveness of the proposed mechanism.

From Table III, we observe the metrics increase when the number of hidden layers of the MLP increases from 2 to 4. Meanwhile, the metrics decrease slightly when the number of hidden layers of the MLP increases from 4 to 6. This is known as the degradation problem: with the network depth increasing, accuracy gets saturated and then degrades rapidly [32]. On the other hand, we also observe that the metrics increase when the number of hidden layers of CNN increases from 2 to 5 and keeps almost the same when the number of hidden layers of CNN increases from 5 to 6. Overall, the fine-tuned DLLD architecture achieves very high F_1-Score and RACC. This, together with the fact that the computational complexity also increases with the number of hidden layers and drives us to design our DLLD architecture with five hidden layers, in order to achieve a good balance between location accuracy and computational complexity.

TABLE III
PERFORMANCE COMPARISON IN THE IEEE 14-BUS SYSTEM

Structure		Precise (%)	Recall (%)	F_1	RACC (%)
MLP	2 layers	96.28	98.00	97.13	51.0
	3 layers	96.55	98.12	97.33	54.3
	4 layers	96.93	98.33	97.62	54.7
	5 layers	96.24	97.83	97.03	52.3
	6 layers	96.21	97.91	97.05	50.9
CNN	2 layers	97.75	98.87	98.31	94.1
	3 layers	99.46	99.64	99.55	95.6
	4 layers	99.50	99.75	99.62	96.7
	5 layers	99.66	99.84	99.75	97.3
	6 layers	99.53	99.75	99.64	97.0
SVM		81.21	86.96	83.99	71.1
LightGBM		81.26	87.00	84.03	71.9

TABLE IV
CLASSIFICATION RESULTS ON THE 3RD, 4TH, AND 11TH MEASUREMENTS

Compromised Location	3rd	4th	3rd&4th	Neither
# of cases	1432	1388	1893	287
Accuracy	0.997	0.997	1.000	0.996
Compromised Location	3rd	11th	3rd&11th	Neither
# of cases	2623	748	702	927
Accuracy	0.997	0.996	0.996	0.998

A point worth noting is that SVM and LightGBM are conventionally proposed for single-label multiclass classification problems. In order to solve the multilabel classification problems using SVM and LightGBM, we convert the multilabel data set to the single-label data set. For example, in the IEEE 14-bus system, the 19 binary labels $(y_t^1, y_t^2, \ldots, y_t^{19})$ are converted into one label with a class size of 2^{19}. Due to the high co-coherence dependency, after we remove the classes that never happen in the whole data set, such that the class size shrinks from 2^{19} to 80. Then, we can employ SVM and LightGBM to classify the converted multiclass problem. As a result, the RACCs achieved by SVM and LightGBM are higher than the ones achieved by MLP-DLLD while the precises and recalls are lower.

We would like to emphasize that the high accuracy achieved by the proposed CNN structure is due to the fact that our proposed CNN structure can capture the inconsistency and co-occurrence dependency on nearby measurements introduced by FDIA. For example, CNN can capture the co-occurrence of FDIA on the 3rd and 4th measurements as they are directly connected. In Table IV, we show the locational classification results on the 3rd, 4th, and 11th measurements. For example, in the first three rows, we demonstrate the classification accuracy for four attack cases: 1) the 3rd measurement is compromised and the 4th is not; 2) the 4th measurements are compromised and the 3rd is not; 3) both the 3rd and 4th measurements are compromised; and 4) neither the 3rd and 4th measurements are compromised. We observe that the co-occurrence of FDIA on the 3rd and 4th measurements is much larger than the one on the 3rd and 11th measurements and the classification accuracy is also higher than the one on the 3rd and 11th measurements. It is because the 3rd

TABLE V
PERFORMANCE COMPARISON IN THE IEEE 118-BUS SYSTEM

Structure		Precise (%)	Recall (%)	F_1	RACC (%)
MLP	2 layers	1.21	0.99	1.09	50.9
	3 layers	1.72	1.07	1.32	50.9
	4 layers	1.67	1.14	1.35	50.1
	5 layers	1.06	1.04	1.05	50.1
	6 layers	1.15	1.02	1.08	50.0
CNN	2 layers	98.40	99.23	98.81	87.5
	3 layers	98.61	99.39	99.00	89.6
	4 layers	99.28	99.50	99.39	93.3
	5 layers	99.08	99.71	99.40	93.3
	6 layers	99.14	99.54	99.34	91.9

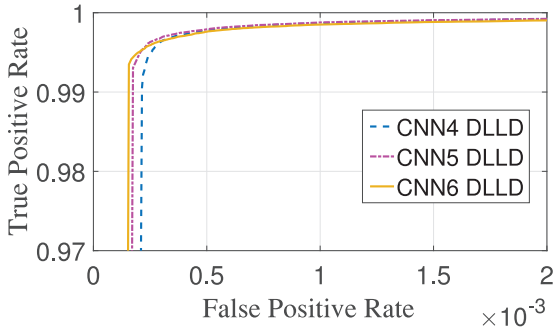

Fig. 4. ROC curve for the proposed mechanism. TPR rises to 0.99 extremely fast when FPR increases from 0 to 0.0002 and thus we only plot TPR versus FPR from 0 to 0.002.

and 4th measurements are directly connected and thus the measurements are highly coupled.

3) IEEE 118-Bus System: The performance comparison in the IEEE 118-bus system is given in Table V. We can observe that the precises and recalls are only about 1%, and the RACCs are always near 50% under MLP-DLLD. This is because when the bus system is large, MLP-DLLD can only detect the presence of the attack but not its location. Moreover, as expected, SVM and LightGBM do not converge in the IEEE 118-bus system after we go through 200 epochs. This is due to the fact that the conversion introduces a huge number of classes, e.g., more than 10 000 in our simulation. In contrast, DLLD still achieves 99.37 F_1-Score and 93.2% RACC. Hence, we conclude that the proposed DLLD is scalable when the system size becomes large.

As discussed in Section III-B, the outputs of the CNN $\hat{y}_n^t$'s are continuous within [0, 1], and are quantized to 0 or 1 by a discrimination threshold. In the above figures, we have fixed the discrimination threshold at 0.5. In general, the value of the threshold determines the tradeoff between TPR and FPR. Specifically, a lower threshold results in a higher TPR and a lower FPR. We investigate the tradeoff in Fig. 4, which plots FPR versus TPR when the threshold varies from 0 to 1. To depict relative tradeoffs between TPR and FPR, the area under the ROC (AUC) is commonly considered as a performance metric of the discriminatory capacity [38]. Here, AUC is defined as the area between the FPR, TPR, x-axis, and y-axis. An excellent model has AUC near to 1, which means that it has a good measure of separability. The model predicates 1s as 1s and 0s as 0s. When a model has AUC near to 0, the model predicates 0s as 1s and 1s as 0s [38]. From the figure, we can see the proposed mechanism has AUC near to 1, which represents the excellent discriminatory capacity of the proposed mechanism.

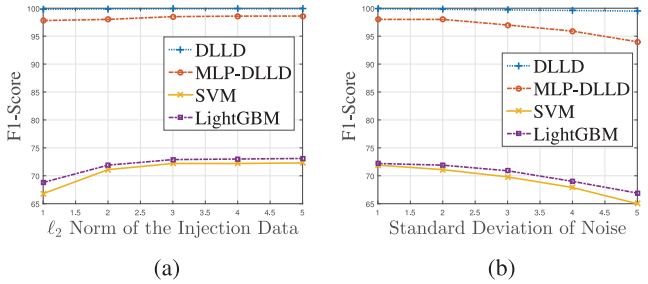

Fig. 5. F_1-Score comparison in the IEEE 14-bus system. (a) Comparison versus the standard deviation of noise. (b) Comparison versus ℓ_2-norm of the injection data.

4) Robustness: In Fig. 5, we evaluate the robustness of the proposed mechanism against the aggressiveness of the attacker and the noise in the data acquisition environment. In particular, we evaluate the proposed mechanism as follows.

1) *Aggressiveness:* We fixed the standard deviation σ to be 0.2, and varied the ℓ_2-norm of the injection from 1 to 5.
2) *Noise:* We fix the ℓ_2-norm of the injection to be 2, and vary the standard deviation σ from 0.1 to 0.5.

Fig. 5 shows that compared with MLP-DLLD, SVM, and LightGBM, DLLD achieves the highest F_1-Score. From Fig. 5(a), the F_1-Score of all of the four schemes increases with the ℓ_2-norm of the injection data. This is because that the patterns of the normal data and the compromised data become more distinguishable when the attack is more aggressive. Likewise, from Fig. 5(b), we can see that when the noise level increases, the F_1-Score of all of the four schemes decreases. This is because that the patterns of the normal data and the compromised data are less distinguishable when the noise power increases. In both subfigures, the proposed DLLD can always achieve F_1-Score near 100, when the standard deviation varies from 0.1 to 0.5 and the ℓ_2-norm of the injection data varies from 1 to 5. This implies that the proposed high accuracy DLLD mechanism is robust to the environmental noise and the size of attack injection.

We would like to emphasize that the runtime of the detection process is only about 100 μs, where the sampling rate for SCADA systems is larger than 100 Hz. This, together with the fact that our simulations were run on a machine with an Intel Xeon E5-2630 CPU, two nVidia GTX 1080 GPUs, and 64-GB RAM, indicates that the proposed mechanism is practical and cost-friendly.

D. Presence-Detection Performance

We take one step back and investigate how well the proposed mechanism works in terms of detecting the presence of attacks. In particular, we regard the power system as uncompromised or the attacks are absent if $\hat{y}_t^i = 0$, for all $i = 1, \ldots, n$. Otherwise, the power system is regarded as compromised or the attacks are present. In Fig. 6, we investigate the FDIA presence-detection performance of the proposed mechanism. In particular, we compare the detection accuracy

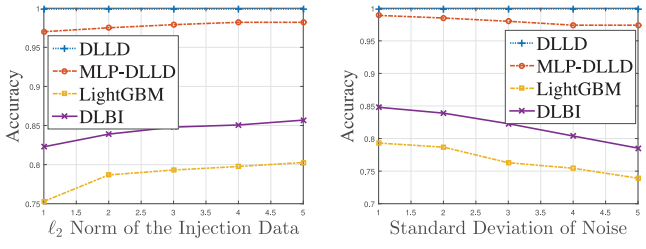

Fig. 6. Accuracy of stealthy FDIA detection in the IEEE 118-bus system. (a) Accuracy versus ℓ_2-norm of the injection data. (b) Accuracy versus the standard deviation of measurement noise.

with the two benchmarks: 1) SVM and 2) DLBI. Moreover, since the performance gaps in the IEEE 118-bus system are more distinct than those in the IEEE 14-bus system. We only plot the accuracy in the IEEE 118-bus system for simplicity.

In Fig. 6(a), we compare the detection accuracy achieved by DLBI, SVM, MLP-DLLD, and DLLD. Overall, compared with DLBI and SVM methods, the proposed detection scheme achieves the highest detection accuracy. Moreover, we can also see that as the noise level increases, the detection accuracy of DLBI and SVM methods decreases, which is similar to the conclusion drawn in Fig. 5(a).

Finally, we investigate the presence-detection accuracy versus the standard deviation of measurement noise in Fig. 6(b). As expected, the proposed scheme achieves the highest detection accuracy. Similar to the conclusion in Fig. 5(b), the detection accuracy of all the four methods increases with the standard deviation of the noise. Before we leave this section, we would like to emphasize that the proposed multilabel classification method, although is targeted at detecting FDIA locations, also improves the presence-detection accuracy. This is because multilabel classification captures the inconsistency and co-occurrence dependency of meter measurements.

V. Conclusion

In this article, we have formulated the locational detection problem of FDIA as a multilabel classification problem and designed a BDD-CNN architecture as a multilabel classifier. The standard BDD detector is to estimate the quality of the real-time measurement data and used to remove the low-quality data. The CNN is to capture the inconsistency and co-occurrence dependency introduced by FDIA. The mechanism is model-free in the sense that the architecture does not depend on any assumed attack model, and is cost-friendly in the sense that the architecture is built on the existing BDD that requires no alternation of the current BDD system and the runtime of the detection process is only hundreds of microseconds on a household computer. Moreover, we have carried out extensive simulations in the IEEE 14- and 118-bus power systems to demonstrate the practicability. In particular, we have shown that DLLD can perform locational detection for the whole bus system under various noise and attack conditions. In addition, we also have demonstrated that the presence-detection accuracy can be further improved through multilabel classification formulation, and thus the achieved presence-detection accuracy is better than that of the state-of-the-art benchmarks.

References

[1] K. Gai, K. Xu, Z. Lu, M. Qiu, and L. Zhu, "Fusion of cognitive wireless networks and edge computing," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 69–75, Jun. 2019.

[2] M. S. Thomas and J. D. McDonald, *Power System SCADA and Smart Grids*. Hoboken, NJ, USA: CRC Press, 2015.

[3] B. M. Horowitz and K. M. Pierce, "The integration of diversely redundant designs, dynamic system models, and state estimation technology to the cyber security of physical systems," *Syst. Eng.*, vol. 16, no. 4, pp. 401–412, 2013.

[4] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1630–1638, Jul. 2017.

[5] R. Deng, G. Xiao, R. Lu, H. Liang, and A. V. Vasilakos, "False data injection on state estimation in power systems—Attacks, impacts, and defense: A survey," *IEEE Trans. Ind. Informat.*, vol. 13, no. 2, pp. 411–423, Apr. 2017.

[6] L. Xie, Y. Mo, and B. Sinopoli, "Integrity data attacks in power market operations," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 659–666, Dec. 2011.

[7] X. Liu, Z. Li, X. Liu, and Z. Li, "Masking transmission line outages via false data injection attacks," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 7, pp. 1592–1602, Jul. 2016.

[8] K. Gai, M. Qiu, Z. Ming, H. Zhao, and L. Qiu, "Spoofing-jamming attack strategy using optimal power distributions in wireless smart grid networks," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2431–2439, Sep. 2017.

[9] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Trans. Inf. Syst. Security*, vol. 14, no. 1, pp. 1–33, Jun. 2011.

[10] S. Bi and Y. J. Zhang, "Using covert topological information for defense against malicious attacks on DC state estimation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 7, pp. 1471–1485, Jul. 2014.

[11] Q. Yang, D. An, R. Min, W. Yu, X. Yang, and W. Zhao, "On optimal PMU placement-based defense against data integrity attacks in smart grid," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 7, pp. 1735–1750, Jul. 2017.

[12] K. Gai, K.-K. R. Choo, M. Qiu, and L. Zhu, "Privacy-preserving content-oriented wireless communication in Internet-of-Things," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 3059–3067, Aug. 2018.

[13] S. A. Foroutan and F. R. Salmasi, "Detection of false data injection attacks against state estimation in smart grids based on a mixture Gaussian distribution learning method," *IET Cyber Phys. Syst. Theory Appl.*, vol. 2, no. 4, pp. 161–171, 2017.

[14] R. Moslemi, A. Mesbahi, and J. M. Velni, "A fast, decentralized covariance selection-based approach to detect cyber attacks in smart grids," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4930–4941, Sep. 2018.

[15] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using Kalman filter," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 4, pp. 370–379, Dec. 2014.

[16] L. Liu, M. Esmalifalak, Q. Ding, V. A. Emesih, and Z. Han, "Detecting false data injection attacks on power grid by sparse optimization," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 612–621, Mar. 2014.

[17] Y. Guan and X. Ge, "Distributed attack detection and secure estimation of networked cyber-physical systems against false data injection attacks and jamming attacks," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 1, pp. 48–59, Mar. 2018.

[18] A. Ashok, M. Govindarasu, and V. Ajjarapu, "Online detection of stealthy false data injection attacks in power system state estimation," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1636–1646, May 2018.

[19] M. Ganjkhani, S. Fallah, S. Badakhshan, S. Shamshirband, and K.-W. Chau, "A novel detection algorithm to identify false data injection attacks on power system state estimation," *Energies*, vol. 12, p. 2209, Jun. 2019.

[20] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Machine learning methods for attack detection in the smart grid," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 8, pp. 1773–1786, Aug. 2016.

[21] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1644–1652, Sep. 2017.

[22] Y. He, G. J. Mendis, and J. Wei, "Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2505–2516, Sep. 2017.
[23] J. J. Q. Yu, Y. Hou, and V. O. K. Li, "Online false data injection attack detection with wavelet transform and deep neural networks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3271–3280, Jul. 2018.
[24] F. Zhao, Y. Huang, L. Wang, and T. Tan, "Deep semantic ranking based hashing for multi-label image retrieval," in *Proc. CVPR Conf.*, Jun. 2015, pp. 1556–1564.
[25] T. N. Sainath, O. Vinyals, A. W. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. ICASSP Conf.*, Apr. 2015, pp. 4580–4584.
[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS Conf.*, 2012, pp. 1097–1105.
[27] D. Silver *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–503, Jan. 2016.
[28] T. Dragičević, X. Lu, J. C. Vasquez, and J. M. Guerrero, "DC microgrids—Part I: A review of control strategies and stabilization techniques," *IEEE Trans. Power Electron.*, vol. 31, no. 7, pp. 4876–4891, Jul. 2016.
[29] A. Anwar, A. N. Mahmood, and Z. Tari, "Identification of vulnerable node clusters against false data injection attack in an AMI based smart grid," *Inf. Syst.*, vol. 53, pp. 201–212, Oct./Nov. 2015.
[30] S. A. Boyer, *SCADA: Supervisory Control and Data Acquisition*. New York, NY, USA: Int. Soc. Autom., 2009.
[31] I. J. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2016.
[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR Conf.*, 2016, pp. 770–778.
[33] R. D. Zimmerma *et al.* (2018). *MATPOWER*. [Online]. Available: http://www.pserc.cornell.edu/matpower
[34] H. Sedghi and E. Jonckheere, "Statistical structure learning to ensure data integrity in smart grid," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1924–1933, Jul. 2015.
[35] F. Chollet *et al.* (2015). *Keras*. [Online]. Available: https://keras.io
[36] C. Cortes and V. Vapnik, "Support vector machine," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[37] G. Ke *et al.*, "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3146–3154.
[38] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.

Shuoyao Wang received the B.Eng. (First Class Hons.) and Ph.D. degrees in information engineering from the Chinese University of Hong Kong, Hong Kong, in 2013 and 2018, respectively.

From 2018 to 2020, he was a Senior Researcher with the Department of Risk Management, Tencent, Shenzhen, China. Since 2020, he has been with the College of Electronic and Information Engineering, Shenzhen University, Shenzhen, where he is currently an Assistant Professor. His research interests include optimization theory, operational research, and machine learning in multimedia processing, smart grid, and communications.

Suzhi Bi (Senior Member, IEEE) received the B.Eng. degree in communications engineering from Zhejiang University, Hangzhou, China, in 2009, and the Ph.D. degree in information engineering from the Chinese University of Hong Kong, Hong Kong, in 2013.

From 2013 to 2015, he was a Postdoctoral Research Fellow with the ECE Department, National University of Singapore, Singapore. Since 2015, he has been with the College of Electronic and Information Engineering, Shenzhen University, Shenzhen, China, where he is currently an Associate Professor. His research interests mainly involve in the optimizations in wireless information and power transfer, mobile computing, and smart power grid communications.

Dr. Bi received the IEEE ComSoc Asia–Pacific Outstanding Young Researcher Award in 2019 and was a co-recipient of the IEEE SmartGridComm 2013 Best Paper Award. He is currently an Editor of IEEE WIRELESS COMMUNICATIONS LETTERS.

Ying-Jun Angela Zhang (Fellow, IEEE) received the Ph.D. degree from the Department of Electrical and Electronic Engineering, Hong Kong University of Science and Technology, Hong Kong, in 2004.

She is an Associate Professor with the Department of Information Engineering, Chinese University of Hong Kong, Hong Kong. Her research interests focus on optimization in wireless communication systems and smart power grids.

Dr. Zhang was a co-recipient of the 2014 IEEE Comsoc Asia–Pacific Outstanding Paper Award, the 2013 IEEE SmartGridComm Best Paper Award, and the 2011 IEEE Marconi Prize Paper Award on Wireless Communications; and a recipient of the 2011 Young Researcher Award of the Chinese University of Hong Kong. As the only winner from Engineering Science, she has won the Hong Kong Young Scientist Award 2006, conferred by the Hong Kong Institution of Science. She had served as the Chair of the Executive Editor Committee of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the Chair of the IEEE Technical Committee on Smart Grid Communications. She is an Editor-at-Large of the IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY, and a Steering Committee Member of IEEE WIRELESS COMMUNICATIONS LETTERS and IEEE SmartGridComm conference. She had also served many years on the Editorial Boards of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON COMMUNICATIONS, and the *Journal of Security and Communication Networks* (Wiley). She was a Guest Editor of the IEEE INTERNET OF THINGS JOURNAL, *IEEE Communications Magazine*, and the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS. She has been on the Organizing Committees of many top conferences, including IEEE GLOBECOM, ICC, VTC, SmartGridComm ICCC, and MASS. She is a Fellow of the Institution of Engineering and Technology, and a Distinguished Lecturer of IEEE ComSoc.