Hindawi
Mobile Information Systems
Volume 2022, Article ID 1148518, 13 pages
https://doi.org/10.1155/2022/1148518

Research Article

SingleNet: A Lightweight Convolutional Neural Network for Safety Detection of an Industrial Control System

Yun Sha,[1] **Jianping Chen**,[1] **Jianwang Gan**,[2] **Yong Yan**,[1] **Xuejun Liu**,[1] **and Hao Wang**[1]

[1]*College of Information Engineering, Beijing Institute of Petrochemical Technology, 19 Qingyuan North Road, Daxing District, Beijing 102617, China*
[2]*School of Mechanical Electronic & Information Engineering, China University of Mining & Technology, 11 Xueyuan Road, Haidian District, Beijing 100083, China*

Correspondence should be addressed to Yun Sha; shayun@bipt.edu.cn

Received 20 January 2022; Revised 27 February 2022; Accepted 28 February 2022; Published 20 March 2022

Academic Editor: Hye-jin Kim

Copyright © 2022 Yun Sha et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The traditional industrial control security research mainly focuses on network intrusion detection or trapping system and lacks abnormal detection after intrusion, and the abnormal detection algorithm ability of the underlying operation data of industrial control is insufficient. The modern industrial control system is a side-cloud collaborative architecture that accesses the Internet, the edge side is usually an industrial computer with weak computing power, and the deep learning algorithm requires a lot of computing resources and is difficult to use directly on the edge side. In this paper, a lightweight convolutional neural network anomaly detection algorithm "SingleNet" suitable for the edge side of the industrial control system is proposed, which convolutes the data of each sensor for a period of time and calculates the feature correlation between points in the association calculation layer. Experimental results show that the accuracy rate on the oil depot dataset is increased from 73% to 99.4%, the training time is shortened from 2 hours to 3 minutes, and the model size is compressed from 101 MB to 1.6 MB. The accuracy rate is improved from 87% to 99.2% on the Mississippi dataset, the training time is shortened from 15 minutes to 3 minutes, and the model size is compressed from 10.6 MB to 1.63 MB. The accuracy rate is improved from 85% to 99.4% on the Batadal dataset, the training time is shortened from 18 minutes to 3 minutes, and the model size is compressed from 15.5 MB to 1.62 MB. Compared with several lightweight algorithms recently proposed, SqueezeNet, MobileNet, and ShuffleNet, the proposed algorithm has significantly improved the performance indicators of training speed, accuracy, model size, and iteration time on the industrial control datasets. Both the training and testing of the algorithm can be done on the CPU, making it possible to apply deep learning to the edge side of the industrial control system.

1. Introduction

Industrial control systems are widely used in infrastructures such as electric power, petroleum and petrochemical, transportation, and advanced manufacturing and are the pillars of national economic development, involving all aspects of people's lives. In recent years, there have been frequent intrusions into industrial infrastructure, with the earliest Iranian "Stuxnet" virus [1] destroying critical infrastructure by tampering with data and issuing false directives. In December 2015, hackers sent malware to infect the grid system on which the receiving network was located, causing more than 1.4 million people to suffer a power outage [2]. In May 2016, the world's first PLC virus was introduced, which could be transmitted directly between PLCs, infecting PLCs and sending denial of service messages that stopped working [3]. In the May 2017 "Wanna Cry" infection incident, the virus spread to the intranets of universities in Europe and China, large enterprise intranets and private networks of government agencies in just five hours, seriously affecting the normal operation of the system [4]. In August 2018, TSMC's chip processing system, which was responsible for

OEM for Apple, Qualcomm, and Huawei, was attacked by a computer virus, and three large campuses were severely damaged, causing economic losses of up to US$250 million [5]. Although the traditional security protection system can alarm the situation that the industrial control production data exceeds the limit, it is difficult to find abnormal situations in the data within the normal value range of each. Abnormal detection of business data of industrial control system is the last line of defense for industrial control security.

Driven by the demands for real-time analysis, control, security and other aspects of industrial field, the introduction of edge computing in industrial control systems has become a trend. With the deep integration of the industrial Internet and the Internet of Things and other technologies [6], a large number of IoT sensing devices have been deployed in the field of critical infrastructure, and the perception network constitutes a new model of edge computing [7], forming an edge computing network in which cyberspace information systems and physical space systems interact collaboratively. In order to explore the potential of edge computing in industrial applications, effective management of tasks based on edge computing can greatly improve the performance of industrial applications [8]. "Stuxnet", "Wanna Cry" and viruses in TSMC chip processing systems all indicate that the attack methods are tailored for the critical infrastructure in the field of industrial control systems, and the attack detection and defense are extremely difficult, and the traditional passive detection and response mechanisms can no longer be applied. Under the dual constraints of high system security and uncertainty of attack characteristics, in order to cope with new situations and challenges, and realize advanced detection and advanced defense against attacks, it is necessary to build defensive line linkage and active defense capabilities for edge computing networks [9].

Business data is the basis for the safe operation of the industrial control system, no matter which level of attack, it will eventually attack the business data to achieve the purpose of destruction, the main attack method is to hijack and tamper with the control system or operating data. Therefore, it is possible to take timely measures by detecting whether the business data is abnormal. Anomaly detection based on the operation of business data is an urgent problem to be solved and is one of the important means to protect system security.

The business logic of the industrial control system is reflected in the data collected in real time by each sensor in the oil depot. When the flow metering value increases, the level count value of a certain storage tank increases faster; when a pump is shut down, the corresponding flow sensor value decreases. When attacking the system according to business logic, the liquid level data and flow data may be tampered with, resulting in liquid spillage in the tank. The point data mentioned later in this article refers to the time series data collected by a sensor in the system, anomaly detection is mainly for the abnormal detection of the underlying data of the industrial control, and the anomaly detection algorithm of these data needs to be deployed on the field equipment of the industrial control system, that is, the edge side of the industrial control system.

At present, most of the anomaly detection used for industrial control business data uses traditional machine learning algorithms [10, 11], although a good accuracy rate has been achieved, this type of method mainly for the classification of the point data collected at a certain time. The attack of the industrial control system is usually an attack that lasts for a period of time, and the attacked point data is even within the standard range, so a single piece of data is not enough to determine whether an abnormality occurs, and it is necessary to analyze the changes in multiple point data over a period of time. The traditional machine learning method has a low accuracy rate for anomaly detection of two-dimensional industrial control data spliced in chronological order, and the security detection ability of the intrusion industrial control system is weak [12, 13]. Therefore, this paper uses the convolutional neural network in deep learning for industrial control anomaly detection, the current computing power of the field device of the industrial control system is limited, and the lightweight convolutional network is the key to solving the problem.

2. Related Research

2.1. Research Status of the Lightweight Network. Since AlexNet [14] popularized deep convolutional neural networks by winning the ImageNet Challenge: ILSVRC 2012 [15], convolutional neural networks have become ubiquitous in computer vision and have also shown great application value in the field of anomaly detection, achieving good accuracy. However, these models are more complex, require more computing resources, and are difficult to directly apply to the Industrial Personal Computer with weak computing power. In the case of ensuring accuracy, reducing the amount of model calculation is a current research hotspot. Jin et al. [16] solve the three-dimensional convolution into three one-dimensional convolutions, which greatly reduces the amount of calculation. Wang et al. [17] changed a complex one-layer network into a multilayer simple network through residual learning, which increased the depth of the network and reduced the amount of calculation. SqueezeNet [18] uses an efficient bottleneck structure to design a very small network, which significantly reduces parameters and calculations while maintaining accuracy. Quantized CNN [19] minimizes the response error of each layer by quantizing the weights in the convolutional layer and the fully connected layer to accelerate and compress the CNN model. Rastegari et al. [20] performed binary operations on the expression of the weights and intermediate layers in the convolutional neural network and found the best approximation of the convolution, which required less memory and less floating-point operations. MobileNet [21] uses deep separable convolution to obtain the latest technological achievements in lightweight models. The concept of grouped convolution first appeared in AlexNet [22], which is used to distribute models on two GPUs. It has been fully proven in ResNet [23], and then used in the Inception model [24] to reduce the amount of calculation in the first few layers.

However, the research background of these methods is in the field of images, and the effect has not been significantly improved in the abnormal detection of industrial control system of business data.

2.2. Research Status of the Lightweight Network in Attack Detection. Lightweight networks also show great application value in the field of anomaly detection. Eskandari et al. [25] proposed a lightweight intelligent intrusion detection scheme. The authors discussed the deployment of the scheme on IoT gateways. The scheme they proposed successfully detected malicious traffic, port scanning and brute force attacks. Latif et al. [26] proposed a new lightweight random neural network (RaNN) predictive model to detect different network security attacks in industrial IoT systems, such as denial of service (DoS), malicious operations, malicious control, data type detection, spying, scanning, and error settings. Kravchik et al. [27] proposed an attack detection method based on simple and lightweight neural networks, which applies a one-dimensional convolutional neural network and an autoencoder to the time and frequency domains of data to detect the physical and network attacks on industrial control systems.

However, these methods are mainly for intrusion detection against network attacks. At present, there are few literatures on business data detection of industrial control system. Once the system is invaded, the attack after gaining control cannot be detected.

2.3. Research Status of Edge Computing. Edge computing is mainly suitable for application scenarios such as the mobile Internet, the Internet of Things, and the Industrial Internet. It has the characteristics of low latency, high security, alleviating traffic pressure, improving efficiency, and data privacy. With the in-depth research and development of technology, scholars have successively proposed marginal models such as fog computing [28] and mobile edge computing (MEC) [29]. The data generated in the mobile edge network requires federated learning [30] to give full play to its value. Federated learning can automatically identify and obtain useful data generated on edge devices, such as text information, image and video information, and vehicle information. Cloudlet [31] developed by Carnegie Mellon University provides mobile computing users with "small cloud" services, extending Open Stack to edge computing platforms, so that scattered small clouds can be controlled and managed through standard Open Stack APIs. Peng et al. proposed a specific edge computing platform that provides computing and storage resources—ParaDrop [32], which uses WIFI access points (AP) and wireless gateways as the network edge to realize localized processing of sensitive data and protect user privacy. Nastic et al. proposed a middleware that supports multilevel configuration of IoT cloud, which provides comprehensive support for multilevel configuration of IoT cloud system [33].

However, the above edge computing method is not suitable for industrial control network, because the industrial control network is a one-way network gate, data can only be uploaded, the calculation of the edge side of industrial control is difficult to use resources with each other, and the way to solve the problem is to reduce the amount of computation under the condition of ensuring the correct rate.

In summary, in view of the characteristics of industrial control business data, this paper proposes an anomaly detection method based on "lightweight" convolutional neural network, and the main contribution of the paper is: 1. Explores a convolutional neural network construction method for industrial control business data; 2. This method makes it possible to apply deep learning to the edge side of the industrial control system.

3. Research Method of the SingleNet Convolutional Neural Network

3.1. Structural Design Strategy

3.1.1. Features of the Industrial Control System Dataset. The industrial control system is essentially a kind of time series data, and data at multiple points can be collected at the same time. Each point will produce a data at each time interval, after M time intervals, the point data can form a curve. Therefore, the abnormal detection of production data can be seen as a change detection problem corresponding to multiple curves of multiple points.

From t_0 to t_M, at the kth point of the industrial control system, the generated data is D_{k,t_0-t_M}, which is expressed as

$$D_{k,t_{0-t_M}} = \left(a_{k,t_0}, \cdots\cdots, a_{k,t_i}, \cdots\cdots, a_{k,t_M}\right). \tag{1}$$

The element a_{k,t_i} in D_{k,t_0-t_M} is the data of the kth point collected at time t_i, and multiple point data within a period of time can form an X matrix.

$$X_{t_0-t_M} = \left(D_{1,t_0-t_M}, \cdots, D_{k,t_0-t_M}, \cdots\cdots, D_{K,t_0-t_M}\right). \tag{2}$$

When the attack occurs during the X matrix period, X is labelled as attacked; otherwise, there is no attack. Continuously collecting data on an industrial control site and attacking it for a period of time, which can form the training and testing sets of the anomaly detection algorithm. The paper designs a lightweight convolutional network anomaly detection algorithm for security monitoring of industrial control sites.

3.1.2. Design Principles. The point data of the industrial control system has a certain correlation with each other, for example, the acceleration of the flow rate will lead to a change in the speed of the rise or fall of the liquid level. The neighborhood convolution method is suitable for extracting the changing features of adjacent pixels in the image, while when the industrial control data is collected, the adjacent point data may be unrelated to the service, and the traditional convolution cannot be used to directly calculate the features. The main purpose of convolution is to learn the changes and coupling relationships between the points in the dataset.

(1) In the industrial control system, the change of a point data in a period of time often reflects whether there is an abnormality, so it is necessary to learn the change of a single point data according to the time span of the point change, so as to extract the independent point data characteristics, enhance the

expression ability of the network, and design a feature extraction method that can characterize the changes in the data of each point

$$F_k = f_k(\alpha_k \bullet \nabla(D_k)), (0 \le k \le K). \quad (3)$$

Among them, the dataset has a total of K points, the kth point feature extraction function is f_k, α_k is the coefficient, and $\nabla(D_k)$ is the change, and the resulting feature map is F_k.

(2) The business association between the points is reflected in the different columns of the sample, so it is necessary to integrate the point features together for correlation calculation, so as to obtain the coupling relationship between the changes between the points

$$Q_L = \omega(\gamma_1 F_1, \cdots, \gamma_i F_i, \cdots, \gamma_K F_K). \quad (4)$$

Among them, Q_L is the coupling correlation of changes in each point.

(3) The edge of the industrial control system computing power is weak, the convolutional network needs to maintain the correct rate, as far as possible to reduce the amount of computation, reduce the network parameters, and need to design a lightweight convolutional neural network

3.2. Network Architecture. The SingleNet convolutional neural network frame diagram based on the single-column convolution is shown in Figure 1.

The inputs on the far left are the original training samples, which are dispersed longitudinally to obtain new samples. The new samples are input to the convolutional layer for feature extraction, the resulting feature maps input the pooling layer for feature reduction. Then the point features are integrated together by the association calculation layer for correlation calculation, and then the resulting feature maps are straightened and flattened into a one-dimensional ordered column vector feature. The fully connected layer connects the feature maps obtained after the convolution of the individual columns, and finally the output results are classified by softmax. The specific process will be described in the next section.

3.3. Design Details

(1) Reduce the amount of computation by reducing the number of convolutional input channels

Suppose a convolutional layer consisting of a convolutional of $n \times n$, in which the total number of parameters in the layer is (number of input channels) × (number of filters) × $(n \times n)$. Therefore, in order to get fewer parameters in the CNN, it is necessary to reduce not only the number of $n \times n$ filters but also the number of input channels in the $n \times n$ convolution.

The traditional LeNet convolutional neural network uses $n \times n$ templates, two layers without pooling, the number of channels to take 32 and 64, and each iteration, the number of calculations for each data in the sample reaches $n \times n \times 32 \times 64$ times, while the network structure in Figure 1 only accesses $n \times n$ times for each data, so the amount of network model computation will be very small.

(2) Obtain a larger feature map by delaying the downsampling rate

In convolutional neural network, if a larger step is taken in front of the network, most feature maps will get a smaller activation map; if the steps in front of the network are all 1 and the steps above 1 are concentrated in the second half of the network, many layers in the network will have large activation maps. In other cases, larger feature maps can achieve higher classification accuracy.

He et al. [34] applied the delayed downsampling rate to four different CNN architectures, and in each case, delayed downsampling resulted in higher accuracy of the classification. Since this article is primarily aimed at small and efficient models, the pooling layer is placed directly behind the convolutional layer when designing the network structure.

3.4. Algorithm Flow. The convolution process of the algorithm in this paper is shown in Figure 2.

Figure 2 is the flowchart of the algorithm in this paper. Firstly, the input training samples are scattered longitudinally, and then, the data of each column after scattering is convoluted in a separate row, using ReLU activation function. The whole process is executed twice into the downpooling layer. Then, the feature correlation between the points is calculated by the association calculation layer, and then the values after the sum of the weights and biases of the fully connected layer are entered into the Sigmoid for nonlinear action, and the results are finally flowed into the output layer. The neurons in the output layer are responsible for receiving the output value of the softmax function classifier, which is used as the probability value of the sample to which it belongs.

Training process:

X is a matrix with m rows and K columns, consisting of m consecutive time points and k point data values, which is called a sample; the current training sample X is divided into K samples $D_1 \boxtimes \cdots \boxtimes D_K$.

The K samples of the input are, respectively, used for the convolution operation with the standard convolution kernel of size (J_{K_w}, J_{K_h}, M, N) and generated a feature map Y of size (J_{Y_w}, J_{Y_h}, N) as output, where M is the number of input channels, J_{Y_w}, J_{Y_h} are the spatial width and height of the output feature map Y, and N is the number of output channels. Assuming that the step size is 1, the convolution calculation formula is shown in

$$d_{ij}^{(l)} = f\left(\sum_{m=0}^{k}\sum_{n=0}^{k} d_{i+m,j+n}^{(l-1)} w_{mn}^{(l)} + b^{(l)}\right). \quad (5)$$

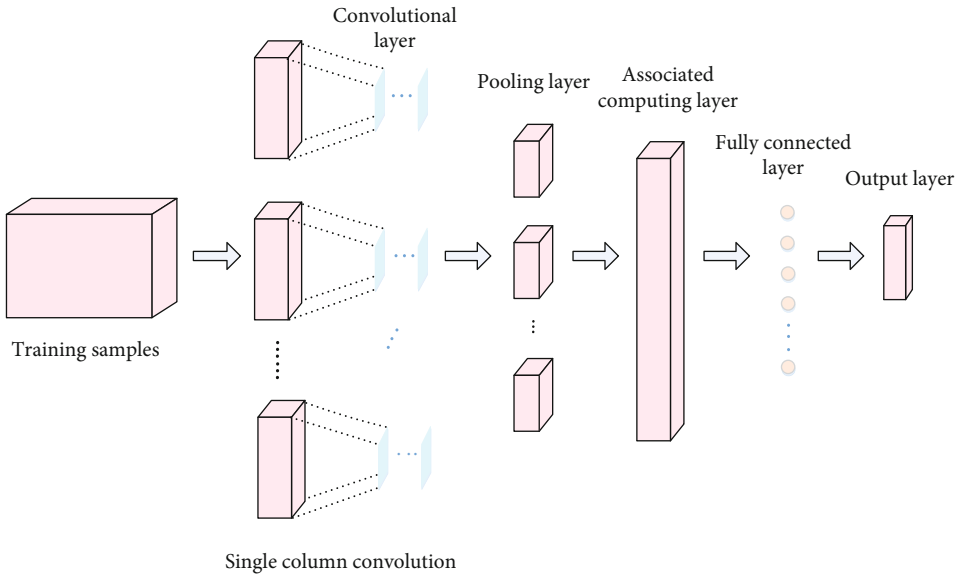

Figure 1: SingleNet's network structure frame diagram.

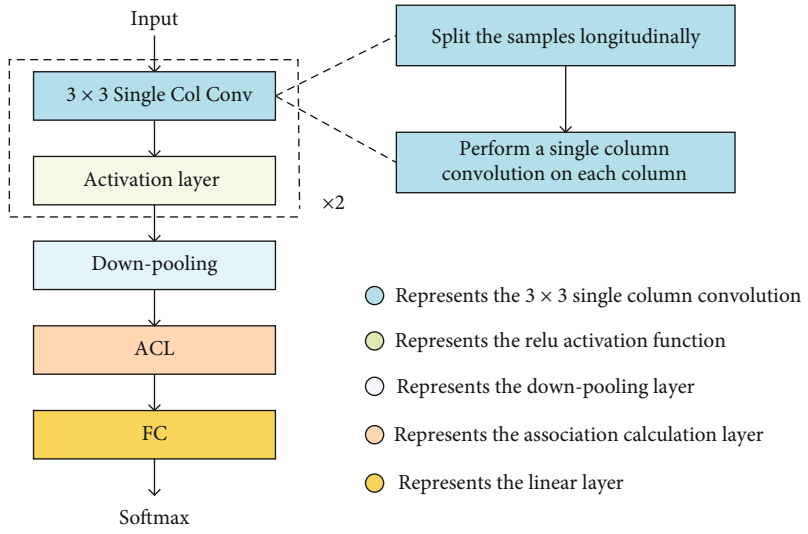

Figure 2: A detailed view of SingleNet.

Among them, l is the lth layer of convolution, $l-1$ is the $l-1$th layer of convolution, $d_{ij}^{(l)}$ is the value of the ith row and jth column of the lth layer of convolution, k represents the value of the convolution kernel length or width, $w_{mn}^{(l)}$ is the weight of the mth row and nth column of the lth convolution in the convolution kernel, $d_{i+m,j+n}^{(l-1)}$ is the pixel value of the $i+m$th row and $j+n$th column of the $l-1$th layer of convolution, $b^{(l)}$ is the bias matrix of the lth layer of convolution, and f is the activation function. According to equation (5), each column after breaking up is individually convolved to obtain the multiscale curve change of each point.

ACL is an association calculation layer, which is used to calculate the relationship between the columns, integrate the point features for association calculation, and obtain the feature association of data, in order to obtain the relevant changes between the points.

Loss function: the training of the entire network in deep learning is the training of the parameters $\theta = \{\theta_1^T, \theta_2^T, \cdots \theta_n^T\}$, where θ contains all the parameters of the model, because the use of cross entropy can improve the accuracy and training speed, so the function uses the cross-entropy loss function [35]; the cross-entropy formula is shown in

$$L(\theta, \theta^*, X) = -\sum_{i=1}^{n} \overline{y_i}(X, \theta_i^*) \log y_i(X, \theta_i). \tag{6}$$

3.5. Training Details

(1) By learning the data change of each point through the convolution operations, two layers of convolution can achieve the ideal result

(2) The activation layer uses ReLU as the activation function to increase the sparseness of the network and reduce the occurrence of overfitting

(3) There are 512 neurons in the association calculation layer, which reduces the network parameters and the amount of computation

(4) Dropout is used in the full connection layer, and the proportion is 50%, which prevents overfitting and improves the generalization ability of the model

(5) The learning rate is set to 0.1 when the model is initialized, and the learning rate is reduced linearly during training, ranging from 0.1 to 0.00001. The most reasonable learning rate is determined to be 0.001 according to the experimental results

(6) The class probability predicted by the model is computed as a cross-entropy loss function with the one hot form of the real class to improve accuracy and training speed

4. Experiment and Result Analysis

4.1. Introduction to the Datasets. Currently, most of the public datasets used for industrial control security are network attack datasets, and there are not many datasets for the business data. The experiments use two datasets containing business data of the industrial control system: the C-Town water distribution dataset constructed by the Cyber Security Research Center of Singapore University of Technology in 2018 to attack the industrial control system of a water plant [36], hereinafter referred to as the Batadal dataset; the SCADA natural gas pipeline dataset constructed by the Mississippi State University Infrastructure Protection Center in 2014 [37], hereinafter referred to as the Mississippi dataset; and the actual production dataset based on the industrial control system of an oil depot constructed by this laboratory [38], hereinafter referred to as the oil depot dataset.

The water distribution system of the Batadal dataset is based on a real medium-sized network. The network pipeline consists of 429 pipelines, 388 connection points, 7 storage tanks, 5 valves, 1 storage tank, and 11 pumps (distributed in 5 pumps). There are a total of 172804 data in the Attackdata file, with a total of 131 feature values and a label column, including 162827 normal samples and a total of 9977 attacked data.

The Mississippi dataset is the data collected by researchers using 28 attack methods to invade industrial control systems, while using a network data logger to monitor and store Modbus traffic from RS-232. Each data in the dataset is a 27-dimensional sequence record. The first 26 dimensions represent 26 different feature values, and the last dimension represents 1 classification label.

Each instance data of the oil depot dataset contains 130 characteristic attributes and a label attribute, including four categories of oil tanks, oil pumps, pipelines, and filters, including a total of 130 points including liquid level, temperature, pressure, and pressure differentials. The label of the normal sample is 0, and the label of the attacked data is 1. Each point data will show its own curve within 50 seconds.

Table 1: Experimental platform parameters.

Parameters	Basic requirements
Operating system	Linux x86_64
CPU	Intel(R) Xeon(R) CPU E3-1230 v3 @ 3.30GHz
GPU	NVIDIA Tesla K80
Memory size	263910424 kB

4.2. Experimental Parameter Settings. The experimental platform parameters of this article are shown in Table 1.

The parameter settings of the lightweight convolutional neural network architecture of this paper are shown in Table 2.

4.3. The Fixed Learning Rate Experiments of the Algorithm in This Paper (SingleNet). The experimental results of applying this algorithm (SingleNet) to the oil depot dataset, the Batadal dataset, and the Mississippi dataset with different learning rates are as follows.

In the fixed learning rate experiments, it can be seen in Figure 3 that when the learning rate is 0.1, the loss of the three datasets reaches 100% when the training iterations reaches 200, and the loss change amplitude is too large. When the learning rate is 0.01, it can be seen that the loss values of the three datasets are basically stable during the training process, and the decline is relatively slow. When the learning rate is 0.001, it can be seen that the final loss values of the oil depot dataset and the Mississippi dataset reached the lowest value compared with others, and although the Batadal dataset did not decline as fast as the learning rate of 0.01, it can be seen that the final accuracy rate is higher than the result of 0.01, and the loss change curve is relatively stable. When the learning rate is 0.0001, it can be seen that the final loss value of the three datasets training does not converge well, and the loss change curve is stable but the decline is relatively slow. When the learning rate is 0.00001, it can be seen that the final loss value of the three datasets training is relatively large, and the loss change curve decreases slowly.

Based on the above experimental results, when the learning rate is 0.001, the accuracy is higher, the loss is stable, the curve is smooth, and the convergence is stable, which is more suitable for practical applications. In the following experiments, the learning rate of the algorithm in this paper is selected as 0.001.

4.4. The Experiments of the Comparison Algorithms. The experiments selected two convolutional neural network algorithms LeNet [39] and AlexNet [14], which are less computation and relatively simple in model structure and the three recently proposed lightweight convolutional network algorithms SqueezeNet [18], MobileNet [21], ShuffleNet

Table 2: Network architecture parameters.

Parameter items	Parameter values
Convolution kernel size	3 * 3
Number of convolution kernels	32
Number of neurons in the fully connected layer	512
Training times	10000
Batch_size	1
Optimization function	Stochastic gradient descent algorithm SGD

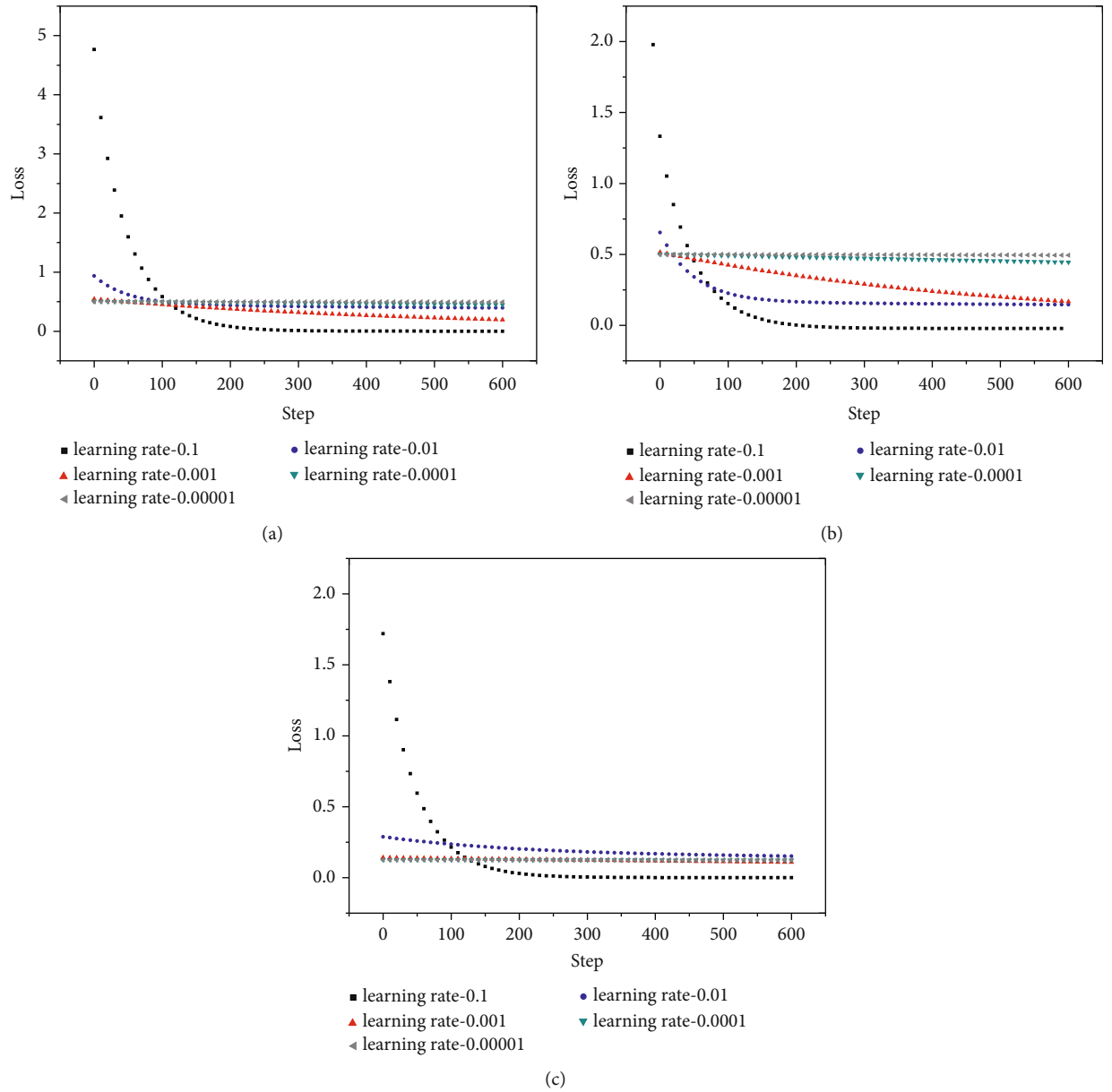

Figure 3: Comparison of experimental results of three datasets with different learning rates: (a) oil depot dataset; (b) Batadal dataset; (c) Mississippi dataset.

[40]. Comparative experiments on three datasets are performed on GPU and CPU platforms to test the effectiveness of the algorithm (SingleNet) in this paper.

4.4.1. GPU Platform. The experimental results of the algorithms on the GPU platform against the oil depot dataset are shown in Table 3.

Table 3: Experimental results of the comparison algorithms on the public datasets.

Datasets	Algorithms	Accuracy	Training time	Model size
Oil depot dataset	SingleNet	98.4%	3 min 18 s	1.6 MB
	LeNet	72.6%	2 h 14 min 26 s	101 MB
	AlexNet	27.6%	1 h 6 min 38 s	22.4 MB
	SqueezeNet	94.5%	2 h 14 min 59 s	2.65 MB
	MobileNet	91.0%	2 h 16 min 49 s	12.4 MB
	ShuffleNet	91.6%	1 h 10 min 34 s	1.25 MB
Batadal dataset	SingleNet	98.4%	3 min 31 s	1.62 MB
	LeNet	84.5%	18 min 37 s	15.5 MB
	AlexNet	46.9%	19 min 24 s	3.15 MB
	SqueezeNet	2.77556e-17	26 min 28 s	2.02 MB
	MobileNet	97.7%	36 min 27 s	12.4 MB
	ShuffleNet	Not convergence	—	—
Mississippi dataset	SingleNet	98.2%	3 min 33 s	1.63 MB
	LeNet	86.4%	15 min 29 s	10.6 MB
	AlexNet	84.4%	8 min 2 s	3.16 MB
	SqueezeNet	98.9%	23 min 10 s	2.66 MB
	MobileNet	89.6%	31 min 10 s	12.4 MB
	ShuffleNet	Not convergence	—	—

After experimental comparison, it can be seen from Table 3 that on the oil depot dataset, LeNet takes 2 hours and 14 minutes to complete the training, with an accuracy rate of only 72.6%; AlexNet takes 1 hour and 6 minutes to complete the training, with the lowest accuracy rate of only 27.6%; the accuracy rate of the three lightweight networks of SqueezeNet, MobileNet, and ShuffleNet is improved compared with the above two algorithms, which are 94.5%, 91.0%, 91.6%, respectively, but the iteration time still takes about one hour to two hours; the algorithm in this paper (SingleNet) only needs 3 minutes and 18 seconds to complete the training, the accuracy rate is as high as 98.4%, and the model size is also compressed from 101 MB to 1.6 MB. At the same time, the accuracy of this algorithm on the Batadal dataset has increased from 84.5% to 98.4%, the training time has been shortened from 18 minutes to 3 minutes, and the model size has been compressed from 15.5 MB to 1.62 MB. On the Mississippi dataset, the accuracy has increased from 86.4% increased to 98.2%, the training time is shortened from 15 minutes to 3 minutes, and the model size is compressed from 10.6 MB to 1.63 MB.

Figure 4 is a comparison diagram of the loss rate of the comparison algorithms during the training process. As can be seen from the figure, the algorithm in this paper (SingleNet) not only has the fastest convergence speed but also the most stable trend. Except for the LeNet algorithm, other algorithms show a trend of severe oscillation during training. Although LeNet can decrease steadily, its convergence speed and accuracy are far lower than the algorithm in this paper. SingleNet uses a single-column convolution method to learn the changes of the points, which effectively reduces the interference between adjacent points, effectively reduces the vibration in the iterative process, and improves the convergence speed of the model.

Figure 5 is a comparison diagram of the loss rate of the comparison algorithm during the training process on the Mississippi dataset. It can be seen from the figure that in addition to the algorithm in this paper, other algorithms also have a trend of violent oscillations during training.

In summary, compared with other algorithms—LeNet, AlexNet, SqueezeNet, MobileNet, and ShuffleNet, the algorithm (SingleNet) in this paper has the fastest convergence speed and the most stable trend, and other algorithms have continuous oscillation or slow decline. Moreover, the algorithm of this paper (SingleNet) has the shortest training time, the smallest model volume, and the highest accuracy.

4.4.2. CPU Platform. LeNet and AlexNet require more computing resources and are difficult to run on the industrial personal computer. In order to verify the effectiveness of the algorithm in this paper on the edge side of industrial control system, SingleNet is compared with three other lightweight algorithms SqueezeNet, MobileNet, and ShuffleNet on three datasets on an industrial computer with the type of Intel(R) Xeon(R) CPU E3-1230. The experimental results are shown in Table 4.

Figure 6 is a comparison chart of the experimental results of the accuracy and training time of the four algorithms on the three datasets.

It can be seen from Table 4 and Figure 6 that the algorithm (SingleNet) in this paper achieves the highest accuracy on the three datasets, the shortest training time, the fastest and most stable convergence speed, and the smallest model size, and the effect is better than that of the GPU server. It

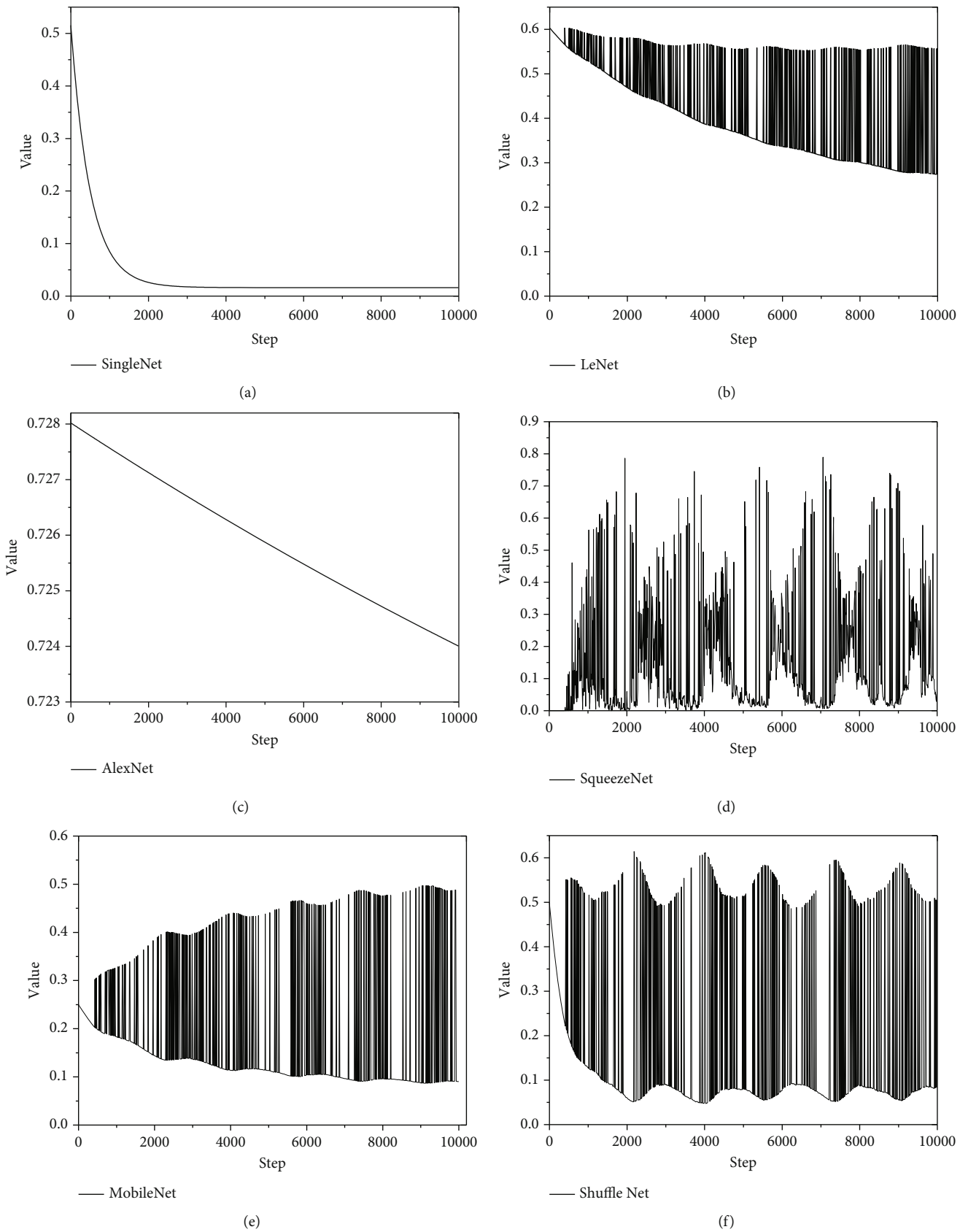

Figure 4: Comparison chart of the loss rate decline of the comparison algorithms on the oil depot dataset: (a) SingleNet; (b) LeNet; (c) AlexNet; (d) SqueezeNet; (e) MobileNet; (f) ShuffleNet.

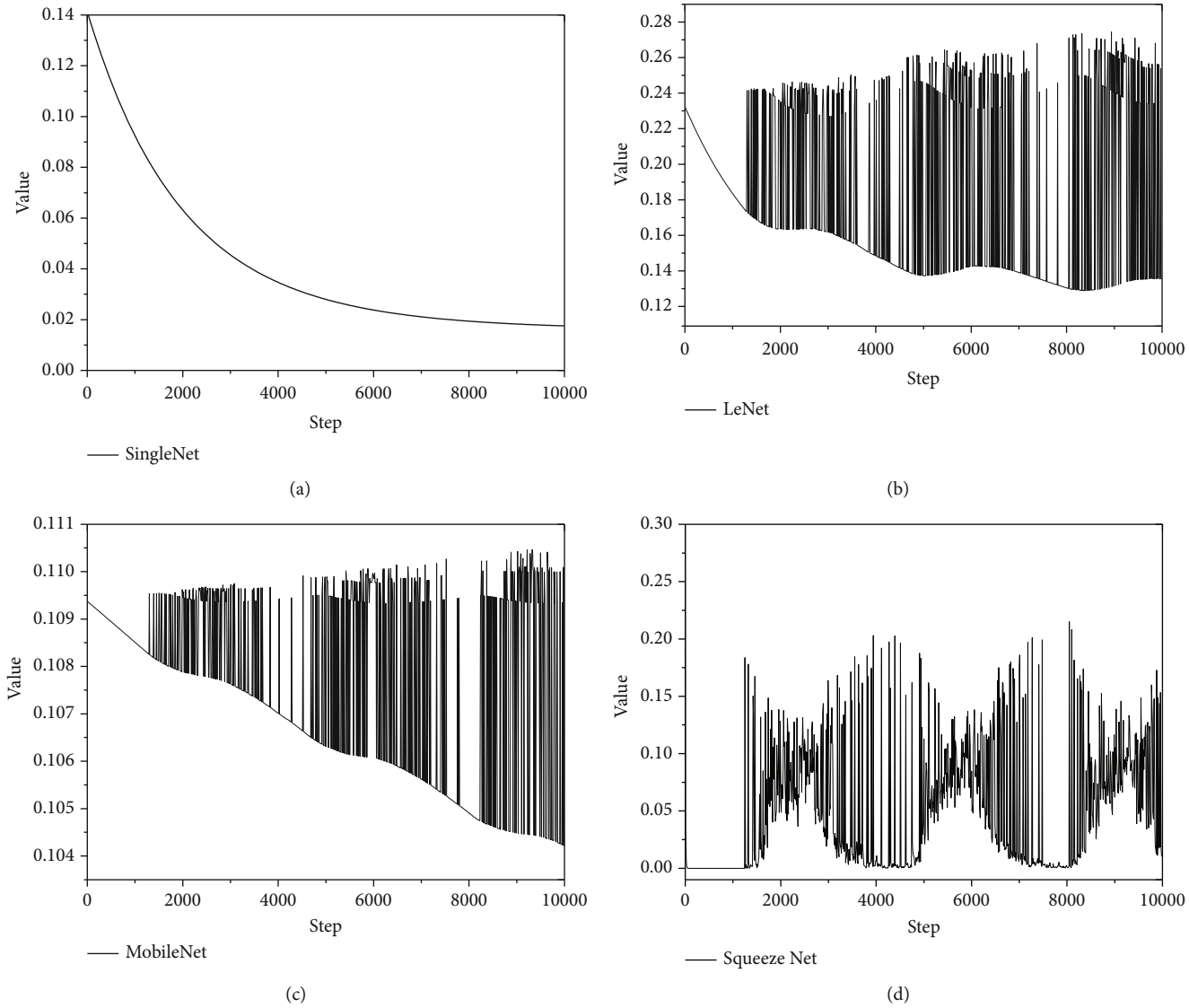

Figure 5: Comparison chart of the loss rate decline of comparison algorithms on the Mississippi dataset: (a) SingleNet; (b) LeNet; (c) MobileNet; (d) SqueezeNet.

Table 4: Comparison table of experimental results of the comparison algorithms on the three datasets.

Datasets	Algorithms	Accuracy	Training time	Model size
Oil depot dataset	SingleNet	98.4%	1 min 26 s	1.60 MB
	SqueezeNet	91.4%	50 min 22 s	2.65 MB
	MobileNet	91.0%	1 h 0 min 32 s	12.4 MB
	ShuffleNet	Not convergence	—	—
Batadal dataset	SingleNet	98.4%	1 min 28 s	1.60 MB
	SqueezeNet	0	8 min 1 s	2.02 MB
	MobileNet	97.7%	13 min 57 s	12.4 MB
	ShuffleNet	Not convergence	—	—
Mississippi dataset	SingleNet	98.2%	1 min 29 s	1.61 MB
	SqueezeNet	97.6%	6 min 48 s	2.66 MB
	MobileNet	89.0%	13 min 25 s	12.4 MB
	ShuffleNet	Not convergence	—	—

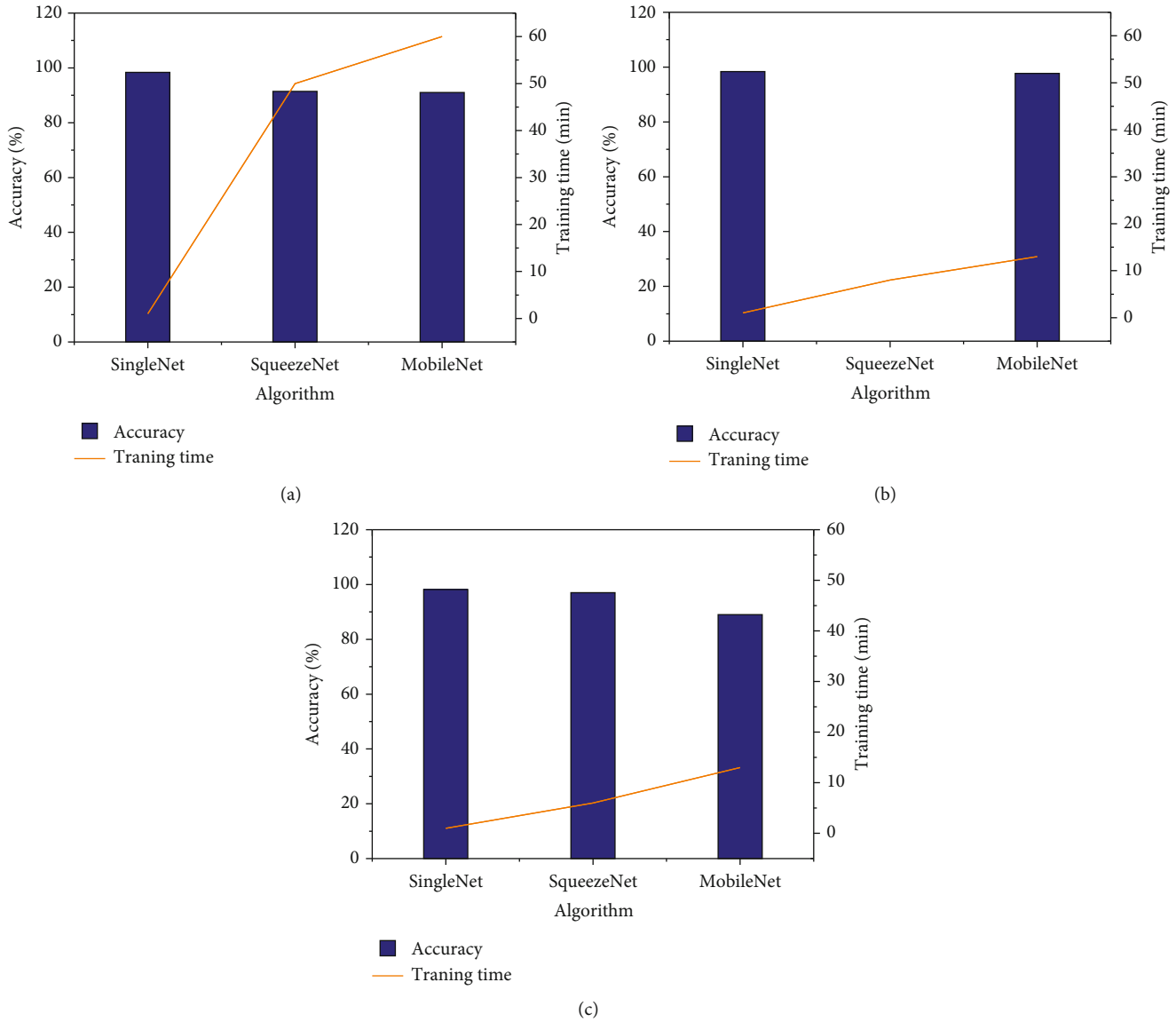

Figure 6: Comparison of experimental results of the algorithms on three datasets: (a) oil depot dataset; (b) Batadal dataset; (c) Mississippi dataset.

provides a basic method for the convolutional neural network to be applied to the edge side of the industrial control system.

5. Conclusion

To sum up, due to the problems of high leak or false alarm rate and poor active defense ability in the current industrial control system, an algorithm that can detect attacks in time and accurately while still being lightweight enough to be deployed in edge side devices with limited computing resources is needed. Therefore, a lightweight and efficient convolution neural network anomaly detection method (SingleNet) is proposed. The innovation of this method is to design the network architecture by means of single-column convolution according to the characteristics of the business dataset of the industrial control system. Compared with the traditional convolution methods, each convolution template of single-column convolution is only responsible for the learning of one point data. Extracting the features of one point not only reduces the amount of calculation but also obtains higher accuracy. Through the association calculation layer, the point features are integrated together for correlation calculation, so as to obtain the correlation changes between points, which improves the training speed, reduces the model volume of the network, saves computing resources, and reduces the requirements of the operating environment, so that the deep learning can be directly applied to the edge side of the industrial control system.

Data Availability

The data used to support the findings of this study are included within the article. Readers can access the data

supporting the conclusions of the study from the oil depot dataset, Batadal dataset, and Mississippi dataset. Among them, the oil depot dataset is confidential and the link cannot be provided. Specific information on the Batadal dataset can be accessed at https://itrust.sutd.edu.sg/itrust-labs_datasets/. Specific information on the Mississippi dataset can be accessed at https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets; the website publishes five types of datasets, and the Mississippi dataset is the fourth.

Conflicts of Interest

The authors declare that there are no conflicts of interest with any financial organizations regarding the material reported in this manuscript.

Acknowledgments

Thanks are due to the support of CNAF (KJ2019003) and other projects.

References

[1] L. A. I. Ying-xu, L. I. U. Zeng-hui, C. A. I. Xiao-tian, and Y. A. N. G. Kai-xia, "Research on intrusion detection of industrial control system," *Journal on Communications*, vol. 38, no. 2, pp. 143–156, 2017.

[2] Q. E. Network, *2015 industrial control network security situation report*, 2016, http://wenku.it168.com/d_001674462.shtml.

[3] P. Hua, "Spread like cancer: the world's first PLC virus comes out industrial control network security becomes an important battlefield for cyberspace confrontation," *Information Security and Communication Confidentiality*, vol. 6, pp. 64-65, 2016.

[4] M. Akbanov, V. G. Vassilakis, and M. D. Logothetis, "Ransomware detection and mitigation using software-defined networking: the case of WannaCry," *Computers and Electrical Engineering*, vol. 76, pp. 111–121, 2019.

[5] National and Local Joint Engineering Laboratory for Industrial Control System Security, *IT/OT Integrated Industrial Information Security Situation Report*, 2018, http://zt.360.cn/1101061855.php?dtid=1101062514&did=610131448.2019.03.

[6] S. Subin and Y. Zhen, "Concept and model analysis of industrial internet," *Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition)*, vol. 35, no. 5, pp. 1–10, 2015.

[7] M. Satyanarayanan, "Edge computing," *Computer*, vol. 50, no. 10, pp. 36–38, 2017.

[8] D. Zhennan, *Research on task offloading strategy based on edge computing in industrial Internet*, Hunan University, 2019.

[9] Z. Bo, *Research on Key Technologies of Edge Computing Network Security Line of Defense Linkage and Active Attack Defense*, Nanjing University of Science and Technology, 2019.

[10] C. Han-yi, *Research on intrusion detection method of industrial control system*, Guilin University Of Electronic Technology, 2019.

[11] Z. Liu, *Application of Anomaly Detection Algorithm Based on Support Vector Machine in ICS*, Guangdong University Of Technology, 2019.

[12] C. H. A. I. Tian-You, "Development directions of industrial artificial intelligence," *Acta Automatica Sinica*, vol. 46, no. 10, pp. 2006–2011, 2020.

[13] Q. I. A. N. Feng, D. U. Wen-Li, Z. H. O. N. G. Wei-Min, and T. A. N. G. Yang, "Problems and challenges of smart optimization manufacturing in petrochemical industries," *Acta Automatica Sinica*, vol. 6, 2017.

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Computer Science*, vol. 1, p. 6, 2014.

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, IEEE, Las Vegas, USA, 2016.

[16] J. Jin, A. Dundar, and E. Culurciello, "Flattened convolutional neural networks for feedforward acceleration," *Computer Science*, vol. 1, p. 3, 2014.

[17] M. Wang, B. Liu, and H. Foroosh, "Factorized convolutional neural networks," *IEEE Computer Society*, vol. 1, 2016.

[18] F. N. Iandola, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and 1mb model size," 2016, https://arxiv.org/abs/1602.07360.

[19] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, USA, 2015.

[20] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *European Conference on Computer Vision*, Cham, 2016Springer.

[21] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017, https://arxiv.org/abs/1704.04861.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[23] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Hawaii, Hawaii Convention Center, USA, 2017.

[24] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," *International Conference on Machine Learning*, vol. 37, 2015.

[25] M. Eskandari, Z. H. Janjua, M. Vecchio, and F. Antonelli, "Passban IDS: an intelligent anomaly based intrusion detection system for IoT edge devices," vol. 7, Tech. Rep. 8, IEEE Internet of Things Journal, 2020.

[26] S. Latif, Z. Zou, Z. Idrees, and J. Ahmad, "A novel attack detection scheme for the industrial Internet of things using a lightweight random neural network," *IEEE Access*, vol. 8, pp. 89337–89350, 2020.

[27] M. Kravchik and A. Shabtai, "Efficient cyber attack detection in industrial control systems using lightweight neural networks and PCA," in *IEEE Transactions on Dependable and Secure Computing*, 2021.

[28] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pp. 13–16, ACM, New York, 2012.

[29] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: a Survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.

[30] H. Mc Mahan, E. Moore, D. Ramage, and B. A. Arcas, "Federated learning of deep networks using model averaging," 2016, https://arxiv.org/abs/1602.05629.

[31] G. Lewis, S. Echeverria, S. Simanta, B. Bradshaw, and J. Root, "Tactical cloudlets: moving cloud computing to the edge," in *2014 IEEE Military Communications Conference*, pp. 1440–1446, Baltimore, MD, USA, 2014.

[32] L. Peng, D. Willis, and S. Banerjee, "ParaDrop: enabling lightweight multi-tenancy at the network's extreme edge," in *2016 IEEE/ACM Symposium on Edge Computing (SEC)*, Washington, DC, USA, 2016.

[33] S. Nastic, H. Truong, and S. Dustdar, "A middleware infrastructure for utility-based provisioning of IoT cloud systems," in *2016 IEEE/ACM Symposium on Edge Computing (SEC)*, pp. 28–40, Washington, DC, USA, 2016.

[34] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, USA, 2015.

[35] A. Sangari and W. Sethares, "Convergence analysis of two loss functions in soft-max regression," *IEEE Transactions on Signal Processing*, vol. 64, no. 5, pp. 1280–1288, 2016.

[36] R. Taormina, N. O. Tippenhauer, S. Galelli, and E. Salomons, "The battle of the attack detection algorithms: disclosing cyber attacks on water distribution netwoks," *Water Resources Planning and Management*, vol. 144, no. 8, 2018.

[37] P. Nader, P. Honeine, and P. Beauseroy, "One-classification for intrusion detection in scada systems," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2308–2317, 2014.

[38] W. Zhou, X. Cao, X. Li et al., "Attack sample generation algorithm based on dual discriminant model in industrial control system," *Journal of Physics: Conference Series*, vol. 1828, article 012123, 2021.

[39] A. Bouti, M. A. Mahraz, J. Riffi, and H. Tairi, "A robust system for road sign detection and classification using lenet architecture based on convolutional neural network," *Soft Computing*, vol. 24, no. 9, pp. 6721–6733, 2020.

[40] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: an extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Hawaii, Hawaii Convention Center, USA, 2017.