IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 November 2024, accepted 20 December 2024. Date of publication 00 xxxx 0000, date of current version 00 xxxx 0000.

Digital Object Identifier 10.1109/ACCESS.2025.3525520

Cyber-Physical System Security for Manufacturing Industry 4.0 Using LSTM-CNN Parallel Orchestration

SALMAN SAEIDLOU[1], NIKDOKHT GHADIMINIA[2], AND KWADWO OTI-SARPONG[3]
[1]School of Engineering, Technology and Design, Canterbury Christ Church University, Canterbury, CT1 1QU Kent, U.K.
[2]Old Royal Naval College, University of Greenwich, SE10 9LS London, U.K.
[3]Centre for Smart Infrastructure and Construction, Department of Engineering, University of Cambridge, CB3 0FA Cambridge, U.K.

Corresponding author: Salman Saeidlou (salman.saeidlou@canterbury.ac.uk)

ABSTRACT Interoperability among different machines, systems, and humans connected via the Internet of Things (IoT) has blessed Industry 4.0 with numerous advantages over the years. However, these benefits have unleashed risks of cyber attacks on internet-connected manufacturing units such as autonomous intelligent computer-controlled cutting (ICNC) machines. These are used in different manufacturing industries to ensure high precision and faster production. Over the Internet these machines receive product designs and instructions of how to produce them. Intrusions through malicious code embedded in the design can hamper precision and cause production delays, resulting in significant revenue loss. This paper presents an innovative cyber-physical system (CPS) security mechanism, using a long short-term memory (LSTM) network and a convolutional neural network (CNN) coordinated by a parallel orchestration (PLO) algorithm. It detects intrusions from both image and text data with 90.85% and 91.66% accuracy, respectively. Applying the proposed methodology in a simulated manufacturing industry shows an average yearly successful intrusion reduction from 184 to 15, saving an average of $30,474 in revenue. Its innovative concept, the distinctive mechanism of the PLO algorithm, and applying it in a simulated manufacturing industry make the proposed security system superior to comparable approaches.

INDEX TERMS Cyber-physical systems, Internet of Things, Industry 4.0, LSTM, CNN, intrusion detection.

I. INTRODUCTION

The advancement of the IoT, artificial intelligence (AI), cloud computing, robotics, and automation has fueled the growth of Industry 4.0 [1]. Different sectors have rapidly embraced Industry 4.0 due to its increased efficiency, productivity, flexibility, and data-driven decision-making [2]. However, these benefits come with security concerns. Early adopters like Epsilon, Equifax, Marriott International, Colonial Pipeline, and T-Mobile faced business losses of $4.3 billion, $700 million, $341 million, $4 million, and $150 million, respectively, due to cyber attacks [3]. The autonomous manufacturing industry is one of the most vulnerable sectors at risk of cyber attacks [4]. The statistics show that 25% of all cyber attacks in 2023 were against the manufacturing industry [5]. This paper presents an innovative CPS security method for the manufacturing industry that automatically detects intrusions from text and images, takes action to maintain manufacturing procedural sequences, and reduces production delays.

The CPS security method presented in this paper has been developed for a simulated ready-made garment (RMG) manufacturing facility that uses Industry 4.0 specifications [6]. The industry standard Arena simulation software has been used to create the simulated production environment [7]. Its production units are interconnected and autonomous, and they have been developed using the real-world response from the physical Prestige 60 machines. The administrators of the production facility can control the units over the Internet, access logs, and monitor status. The autonomous units cut

The associate editor coordinating the review of this manuscript and approving it for publication was Zhan-Li Sun.

clothes based on predefined images. Other forms of communication are carried out through text data. A security layer has been developed in this paper by the parallel orchestration of an LSTM network and CNN to secure communication between manufacturing units and the operators. The LSTM network detects intrusions from text data, and the CNN does the same from image data. The security system has been trained with CICIoT2023 [8] and Malimg [9] datasets. This novel CPS security layer significantly reduces the production delay caused by intrusions and saves revenue. The core contributions of this research are listed below:

- **Parallel Orchestration Algorithm (PLO):** the PLO algorithm developed in this paper maintains a perfect harmony between the LSTM network and CNN, ensuring protection against intrusion from both image and non-image data.
- **Classification Accuracy:** the CPS security layer detects intrusion from image and non-image data with 90.85% and 91.66% accuracy, respectively. The average precision, recall, and F1-score are above 90% as well.
- **Reducing Production Delay:** the most significant contribution of the proposed security system is reducing the average production delay caused by intrusions from 116 hours to 5 hours per year. It saves 169 units of product yearly.
- **Revenue Savings:** the computational analysis shows that the proposed security system helps save $30,474 worth of revenue per year on average, when six manufacturing units run at their full capacity.

In the evolving landscape of Industry 4.0, the integration of intelligent systems becomes paramount. In response to these evolving challenges, this study introduces a novel integration of LSTM networks and CNN through our uniquely designed PLO algorithm. This orchestration is not merely a technical enhancement but a strategic innovation that significantly reinforces CPS security. The PLO algorithm enables a dynamic, real-time response mechanism, effectively enhancing detection capabilities and mitigating threats more efficiently than traditional models. This integration demonstrates a substantial advancement in the application of machine learning technologies to secure Industry 4.0 infrastructures, highlighting the original contribution of our work to the field.

The remaining part of the paper has been organized into six distinct sections. The second section presents the literature review, with discussion related to the background of this paper in the third section. The methodology used to develop the proposed security system is in the fourth section. The computational results and performance evaluation have been presented and analyzed in the fifth section. There are several limitations of this paper, which are highlighted in the sixth section. Finally, the paper concludes in the seventh section.

II. LITERATURE REVIEW

The literature reviews on CPSs conducted by Pivoto et al. [10], Oks et al. [11], and Lampropoulos et al. [12] suggest that cyber security is one of the most significant concerns of Industry 4.0. The IoT is at the heart of autonomous manufacturing units, facilitating communication with the units over the Internet. However, because of their resource-constrained nature, IoT devices are unsuitable for embedded sophisticated security features [13]. Research and development to integrate additional security layers is necessary, as has been carried out in this paper.

Li et al. [14] applied federated deep learning for intrusion detection in CPS. Their approach combines a CNN and a gated recurrent unit (GRU), which is similar to the proposed methodology. However, their study is limited to exploring the classification performance, whereas the proposed system explores the real-world effects of the security system. O'Donovan et al. [15] proposed a fog computing-based security system using machine learning (ML) for a CPS. This innovative approach leaves a significant weakness, whereby the system can be affected by network intrusion because the security features run on the cloud. The proposed system is a security layer just above the hardware level, which ensures maximum security of the CPS production units. A support vector machines (SVM)-based approach proposed by Sharma et al. [16] detects vulnerabilities in a CPS. However, it doesn't provide any mechanism to resume the production process; this has been developed in the proposed paper.

The quantum deep learning (QDL) approach by Rajawat et al. [17] is unique to CPS security. However, the manufacturing industries are still not ready to adopt quantum computing-based solutions [18]. Abdullahi et al. [19] trained an LSTM network to detect cyber attacks on a CPS without applying it in real-world scenarios. The hybrid approach of Alguliyev et al. [20] combines CNN, GRU, and LSTM to develop a cyber-attack detection system for a CPS. However, the approach does not include developing a practical application model. The blockchain-based approach developed by Alabadi et al. [21] demonstrates a promising performance. However, processing delays are a significant barrier to adopting it as a security feature for CPS manufacturing units. While, Wu et al. [22] explored a digital twin (DT)-based approach, which is very computational resource consuming, making it impractical to minimize the production cost. Compared to these approaches, the proposed method is a practical approach that is not confined within the boundary of theoretical analysis. It has a significant impact on reducing production loss due to cyber-attacks, and on increasing revenue.

III. BACKGROUND

The previous section has provided a detailed literature review outlining how current research, addresses the vulnerabilities inherent to Industry 4.0 manufacturing machines. Moving forward, we will delve into some of these vulnerabilities, setting the stage for the introduction of our innovative solution. The results presented in this paper involve a particular industrial manufacturing machine with an onboard computer

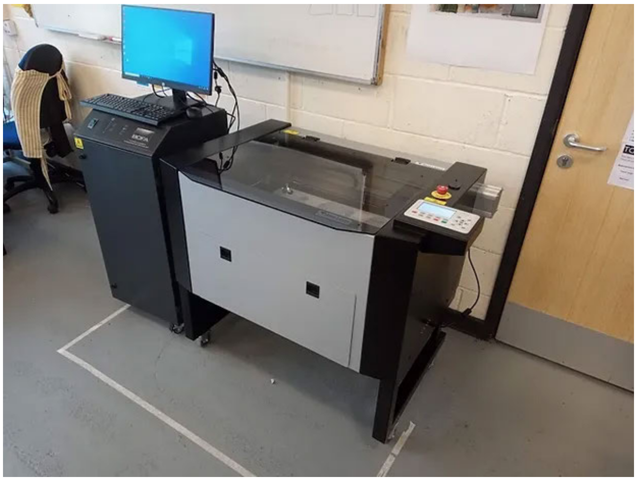


FIGURE 1. The intelligent computer-controlled cutting (ICNC) machine used to study the proposed cyber-physical system security for the manufacturing industry 4.0.

with intelligent modules. This section discusses background knowledge about this machine and its vulnerabilities, which are the subject of this research.

A. INTELLIGENT COMPUTER-CONTROLLED CUTTING (ICNC) MACHINE

This experiment used the intelligent computer-controlled cutting (ICNC) machine illustrated in Figure 1. It is called Prestige 60, designed and developed by Kent Lasers. This ICNC machine has been designed for both educational and commercial production environments and has the capability of high-precision laser cutting and engraving. It uses linear guideways to ensure proper alignment of all optical components. As a result, it achieves exceptional precision. A 50W high-power density laser cartridge cutting (LCC) unit follows the designs provided by the computer-aided design (CAD) layout and cuts the cloth with rapid speed. The operation is controlled by user-friendly advanced software. High-speed AC servo motors move the LCC unit according to the CAD layout with very high precision and accuracy. The software interface is connected to the motor controller through a universal serial bus (USB) port.

B. ARENA SIMULATION SOFTWARE

The Arena simulation software, developed by Rockwell Automation, is widely used to simulate various industries, including manufacturing, healthcare, and logistics. It enables the creation of complex, real-life-like virtual environments for experimenting without an actual physical environment. Physical experimental phases involve risks of damaging physical equipment, which can be costly. The proper application of this software helps prevent such unwanted incidents. It is possible to create precise digital replicas of operational processes to identify bottlenecks, evaluate the impact of different scenarios, and optimize resource allocation and workflows. It can generate real-world statistical reports that aid in decision-making, improving operational efficiency, and enhancing performance across an organization's value chain [23].

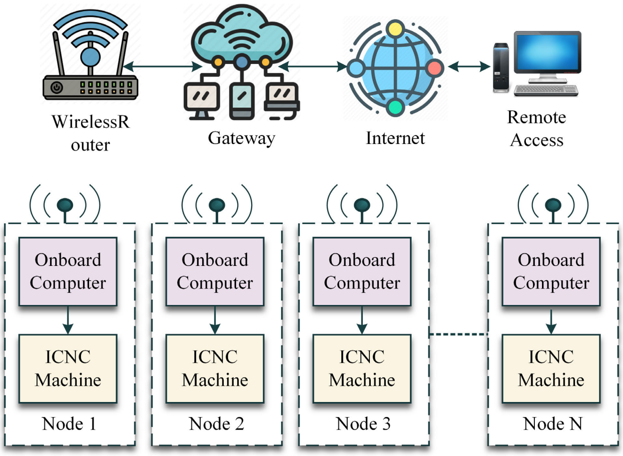


FIGURE 2. The simplified communication model.

C. VULNERABILITY ANALYSIS

The ICNC machine offers remote access and monitoring facilities. Besides, it collects the data and stores it on remote storage. These communications are done over the Internet. In this communication model, the ICNC machines are IoT nodes, as illustrated in Figure 2. The onboard computers, physical Wi-Fi routers, and devices with remote access are the most vulnerable entities of the communication system. Anyone with access credentials to any of these devices can intrude on the IoT devices. The ICNC machines are connected to the onboard computers through network ports; this means they are locally accessible, which is another point of vulnerability for the experimental system.

The concepts and frameworks discussed in this section lay the foundation for the subsequent exploration of the proposed LSTM-CNN orchestration model. Understanding these foundational elements is essential for appreciating the complexities and nuances of the security enhancements detailed in the following sections. The transition from theoretical underpinning to practical application is crucial for grasping the full scope of the research contributions to CPS security.

IV. ADVANCED INTRUSION DETECTION METHODOLOGY

An overview of the proposed methodology is illustrated in Figure 3. The process starts with selecting the appropriate deep neural network (DNN). After that, the CNN and LSTM architectures were developed. Finally, the PLO algorithm was designed to coordinate between these two networks. The methodological details have been presented in this section.

A. FEATURE-BASED NETWORK SELECTION

The experimental manufacturing machine uses both image and textual data to operate. There are numerous deep learning (DL) networks for these two data types [24]. This section presents a comprehensive feature analysis with

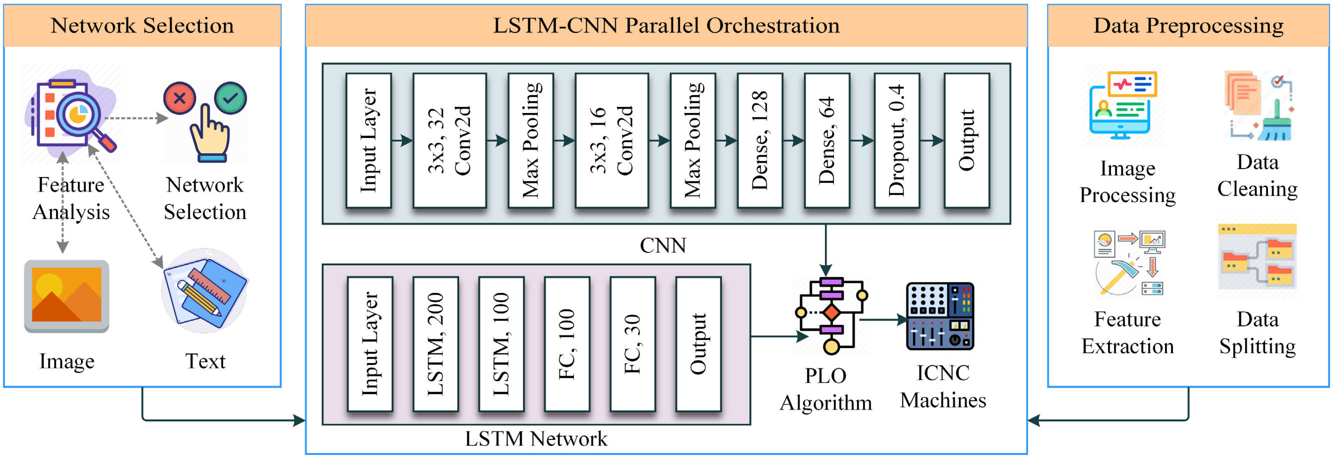


FIGURE 3. Overview of the proposed methodology.

respect to network capability, to choose the most appropriate networks to develop the proposed security mechanism for the cyber-physical system (CPS).

1) IMAGE FEATURES

According to Burri et al. [25], convolutional neural networks (CNNs), defined in equation 1, are most appropriate for classifying images based on their features. The experimental cyber-physical system commands the LCC according to the design presented as an image. Malicious code embedded in the image as hidden image features is a security threat to the system. As CNNs efficiently learn from image features and classify them accurately [26], it is an appropriate DL network for the proposed security system.

$$\hat{Z}_{mn} = (\hat{W} * \hat{K})_{mn} = \sum_{\hat{x}} \sum_{\hat{y}} \hat{W}_{(m+\hat{x})(n+\hat{y})} \hat{K}_{\hat{x}\hat{y}} \tag{1}$$

In Equation 1, $\hat{W}$ denotes the hidden layers that extract image features, followed by a pooling layer. $\hat{K}$ represents the kernel with dimensions $\hat{x} \times \hat{y}$, and $\hat{Z}_{mn}$ is the output at coordinates (m, n).

2) TEXT FEATURES

The experimental CPS accepts commands from remote devices through the Internet. It also has a local access point. The commands and control signals from both access points come in as packets representing textual features [27]. These packets form sequential data at the execution end. According to Halbouni et al. [28], recurrent neural networks (RNN) and long short-term memory networks (LSTM) are appropriate for sequential data. The RNN is defined by equation 2, and it produces output by following the principle of equation 3 where 2 defines how the hidden state $\bar{z}_t$ is updated, while Equation 3 defines the calculation of the output.

$$\bar{z}_t = \bar{\omega}(\bar{Q}_{zz}\bar{z}_{t-1} + \bar{Q}_{xz}\bar{x}_t' + \bar{c}_z) \tag{2}$$
$$\bar{y}_t' = \bar{Q}_{zy}\bar{z}_t + \bar{c}_y \tag{3}$$

However, RNN suffers from vanishing and exploding gradients [29], long-term dependency issues [30], and hierarchical representation problems [31]. According to Sherstinsky et al., [32], LSTM is a type of RNN that overcomes the limitations. it operates using input, forget, and output gates, represented by Equations 5, 4, and 6, respectively.

$$F_t = \psi(\mathcal{M}_f \langle \mathcal{H}_{t-1}, \mathcal{X}_t \rangle + \mathcal{B}_f) \tag{4}$$
$$I_t = \psi(\mathcal{M}_i \langle \mathcal{H}_{t-1}, \mathcal{X}_t \rangle + \mathcal{B}_i) \tag{5}$$
$$O_t = \psi(\mathcal{M}_o \langle \mathcal{H}_{t-1}, \mathcal{X}_t \rangle + \mathcal{B}_o) \tag{6}$$

Based on the characteristics of the DL networks discussed in this section, CNN and LSTM are the most appropriate models to develop a CPS security system that involves both image and textual data; thus, these two networks have been used in this paper.

B. DATASET ANALYSIS & PREPROCESSING

The CICIoT2023 dataset has been used to develop the proposed CPS security system. The Canadian Institute for Cybersecurity curates it and serves as a comprehensive resource for research in IoT security. It includes distributed denial of service (DDoS), denial of service (DoS), brute force (BF), Mirai botnet (MB), spoofing, and many other intrusions. The dataset expands up to 80 different traffic features. A sample of the dataset is listed in Table 1.

1) CLEANING AND SPLITTING DATASET

The maximum number of features on the dataset is 80. However, numerous rows have fewer features because of missing values. The dataset was cleaned before further processing, which started with addressing the missing, duplicate, and outlier values. In the beginning, the dataset was denoted as D, which is defined by equation 7.

$$D = \{d_1, d_2, \ldots, d_N\} \tag{7}$$

TABLE 1. A simplified sample of the CICIoT2023 dataset where four features have been mentioned.

Intrusion Type	Unique Feature 1	Unique Feature 2	Unique Feature 3	Unique Feature 4
DDoS	Packet Burst Frequency	Unique Source Port Count	Payload Entropy	Response Time
DoS	ICMP Echo Request Rate	Failed Connection Count	Flow Byte Distribution	Average Packet Size
Brute-Force	Login Success Ratio	Account Lockout Rate	Sequential Login Attempts	Timestamp Drift
Spoofing	IP Address Randomization	Protocol Violation Flags	ARP Cache Poisoning	Network Device Anomalies

The attributes or features of the dataset are denoted by A_j, among which some of the data are missing. The missing features are $\mathcal{M}_j$, which have been replaced by a substitute calculated using equation 8 where μ_j is the replaced value. After calculating the missing values, it was injected back into the dataset by following the mathematical principle of equation 9.

$$\mu_j = \frac{1}{N - |\mathcal{M}_j|} \sum_{d_n \in D \setminus \mathcal{M}_j} d_{n,j} \tag{8}$$

Subsequently, the missing values were replaced with μ_j according to Equation (9).

$$d_{n,j} = \begin{cases} \mu_j, & \text{if } d_n \in \mathcal{M}_j \\ d_{n,j}, & \text{otherwise} \end{cases} \tag{9}$$

It has been observed that there are numerous instances of duplicate values. Having too many duplicate values in the dataset increases the probability of overfitting [33]. These duplicate values have been removed using equation 10 where D^* represents the refined dataset containing unique entries. After that, it was discovered that there were many outliers in the dataset, which have been removed. Figure 4 shows the data distribution before and after removing the outliers.

$$D^* = \{d \in D \mid \nexists d' \in D \setminus \{d\}, d = d'\} \tag{10}$$

After cleaning, 61,922 instances remained in the dataset. It is suggested in the state-of-the-art approaches that splitting the dataset into training, testing, and validation sets with a ratio of 70:15:15 generates optimal results [34]. After splitting the dataset, 43,345 instances for training were found. Both of the testing and validation sets had 9288 instances each.

2) FEATURE EXTRACTION FROM TEXT DATA

The LSTM network of the proposed CPS security layer is responsible for detecting intrusions from the network packets containing text data. The clean CICIoT2023 dataset is a labeled dataset that needs to be converted into sequences to train the LSTM network. Besides, the data needs to be normalized based on having different ranges of numerical values. Selecting appropriate features to train the LSTM network is another step involved with the feature extraction process. All of these steps have been presented in this section.

a: DATA NORMALIZATION

The numeric ranges of the instances of the dataset widely vary. As a result, it is essential to normalize them to fit within the same scale to avoid training the LSTM network improperly. In this paper, the Z-score normalization has been used to normalize the data; it is defined by equation 11 where x_{nj} is the feature value to be normalized.

$$z_{nj} = \frac{x_{nj} - \mu_j}{\sigma_j} \tag{11}$$

The $Z = \{z_1, z_2, \ldots, z_N\}$ is the normalized dataset that has been used to train the LSTM network. Here $z_n = (z_{n1}, z_{n2}, \ldots, z_{nm})$ represents the feature vector of the n^{th} sample. At the beginning of the process, the cleaned dataset is expressed $D = \{d_1, d_2, \ldots, d_N\}$ where N is the index of the last instance of the dataset. The feature vectors of n^{th} sample are $d_n = (x_{n1}, x_{n2}, \ldots, x_{nm})$. The Z-score calculation requires the mean μ_j and standard deviation σ_j, which are calculated using equations 12 and 13.

$$\mu_j = \frac{1}{N} \sum_{n=1}^{N} x_{nj} \tag{12}$$

$$\sigma_j = \sqrt{\frac{1}{N} \sum_{n=1}^{N} (x_{nj} - \mu_j)^2} \tag{13}$$

b: FEATURE VECTOR REDUCTION

Feature vector reduction is an essential step in DL-based approaches for better performance [35]. The dataset used to develop the proposed CPS security layer is a massive dataset with multiple features weakly associated with the target variable. A mutual information (MI) method [36] was employed to reduce the feature vector by using the most relevant features only. The mathematical principle that governs this process is expressed in equation 14.

$$MI(F, T) = \sum_{f \in F} \sum t \in Tp(f, t) \log \frac{p(f, t)}{p(f)p(t)} \tag{14}$$

In equation 14, the relevant features are expressed by F, and the target variable is T. The method depends on join and marginal probability, which have been denoted by $p(f, t)$, $p(f)$, and $p(t)$, respectively. The MI method ranks the features based on their relevancy.

c: SEQUENCE GENERATION

The last step of the feature extraction is sequence generation for the LSTM network. The input layer of an LSTM network requires sequential data [37]. Dong et al. [38] used the sliding window method to generate a sequence for the LSTM network from the normalized dataset. A similar method was

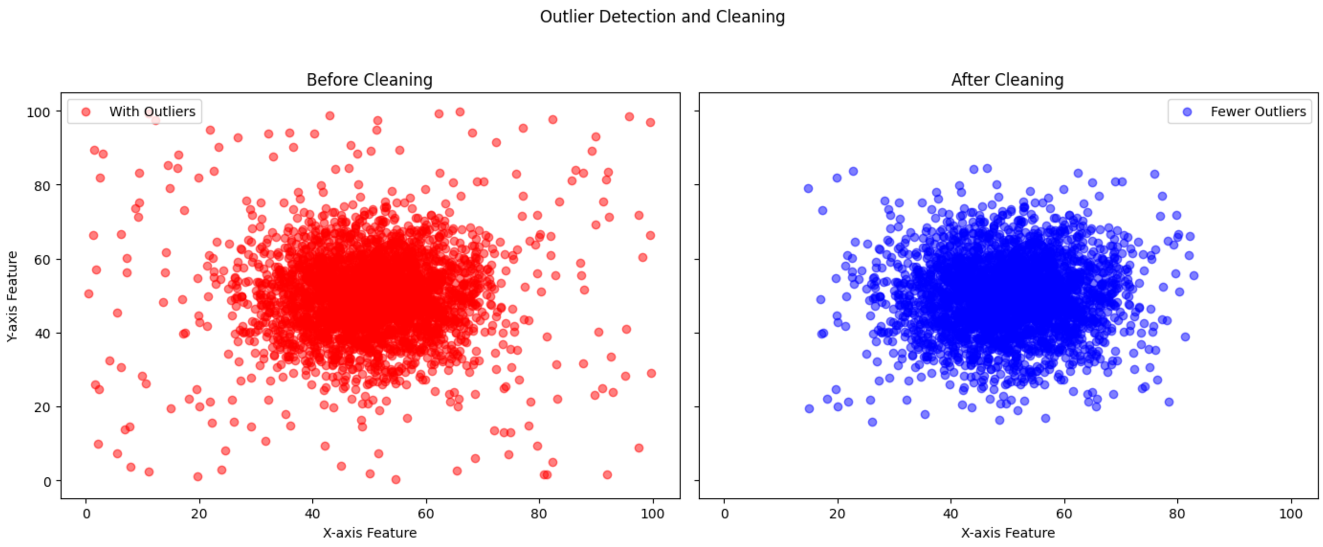


FIGURE 4. A random sample of 3000 features before and after removing outliers.

TABLE 2. The dataset description of the malimg dataset.

Feature	Description
Pixel and Colorspace	Malware samples are depicted as grayscale images. Pixel values range from 0 (black) to 255 (white).
Target Variable	Seven unique malware families or classes.
Number of Instances	A total of 9,339 images representing various malware samples.
Image Resolution	Varies, with common sizes being 64x64 or 256x256.
Instance Balance	Certain malware families are over-represented relative to others.
File Format	Common image formats such as PNG and JPEG.
Data Labeling	Each image is categorized according to the corresponding malware family.

followed in this paper, which is defined by equation 15.

$$(X_{t-w+1}, X_{t-w+2}, \ldots, X_t) \mapsto Y_{t+1} \tag{15}$$

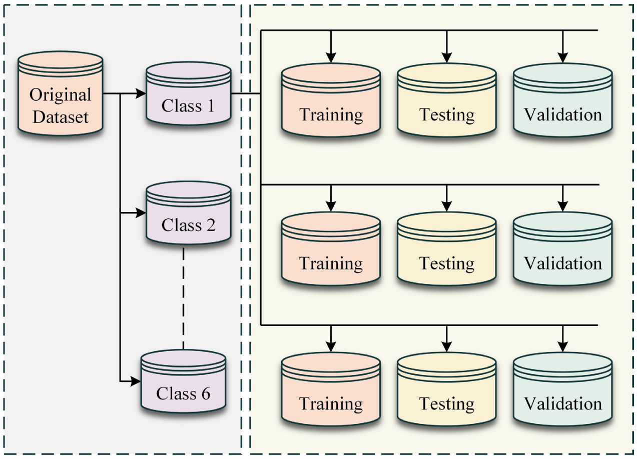
In equation 15, w is the window size. It slides over each dataset row and generates a sequence maintaining a specific timestep. The input to the process is X_t at t timestep, and the output is Y_{t+1}.

3) IMAGE DATASET ANALYSIS AND PREPROCESSING

Multiple effective methodologies suggest that the Malimg dataset is a widely used dataset to study intrusion through image [39]. It was developed and first studied by Nataraj et al. [40]. The original dataset has 9339 images embedded with malicious code at the pixel level. The dataset description is presented in Table 2

a: MALIMG DATASET SPLITTING

The original Malimg dataset is divided into seven directories, each representing a target class. Each target class directory has been further divided into training, testing, and validation directories, as depicted in Figure 5. The image data available in each target class directory has been divided into training, testing, and validation datasets with a ratio of 70:15:15, respectively, and allocated in the designated directories.



(a) Original Dataset Structure (b) Restructured Directories

FIGURE 5. The dataset structure for training.

b: MALICIOUS IMAGE PROCESSING

The images of the Malimg dataset come in different sizes. As a result, they are not ready to be used to train a CNN. It is essential to resize them into a uniform resolution. However, the malicious codes are embedded into the image pixels, and resizing the images alters the code patterns. The nearest neighbor interpolation (NNI) was used in this experiment to resize the image into a 64 × 64 pixel size while keeping the malicious code features intact. The size of the original image in the Malimg dataset is defined as $M \times N$, and the task is to resize it to a target size of $H \times W$, where $H = 64$ and $W = 64$. The original image is expressed as $I(x, y)$, where $x \in [0, M-1]$ and $y \in [0, N-1]$. The original images will be resized as $I'(x', y')$, where $x' \in [0, H-1]$ and $y' \in [0, W-1]$. The relationship between the coordinates of the original and resized images is defined by equations 16 and 17.

$$x = \left\lfloor \frac{x' \times (M-1)}{H-1} \right\rfloor \tag{16}$$

$$y = \left\lfloor \frac{y' \times (N-1)}{W-1} \right\rfloor \tag{17}$$

Using equations (16) and (17), the pixel value of the resized image at coordinates (x', y') is obtained from the nearest pixel in the original image by following the mathematical structure explained in equation 18. This is how the original malicious pixel values are retained in the resized image with exact feature representation.

$$I'(x', y') = I\left(\left\lfloor \frac{x' \times (M-1)}{H-1} \right\rfloor, \left\lfloor \frac{y' \times (N-1)}{W-1} \right\rfloor\right) \tag{18}$$

C. LSTM-CNN PARALLEL ORCHESTRATION

The CNN and LSTM networks were developed separately. Then, the PLO algorithm was designed to maintain harmony between these two networks and defend against intrusion through both image and non-image data.

1) CNN ARCHITECTURE

A CNN has been designed to learn malicious image features effectively. The original Malimg dataset has seven classes and six of these classes are most appropriate for the proposed CPS security layer. Therefore, the CNN has been designed to classify malicious input images into seven classes, six of which are malignant and one is benign.

a: INPUT AND CONVOLUTIONAL LAYER

Although the Malimg images are in grayscale with a single channel, the CNN designed to develop the CPS security layer has a three-channel input layer of the size 64 × 64 × 3. It has been prepared to offer real-time protection in real-world scenarios where most of the images are three-channel images. The input layer passes the signals to the convolutional layer, which has 32 filters with 3 × 3 convolutional matrix dimension. The input layer doesn't perform any mathematical operations. It simply transmits the input to the convolutional layer. The convolutional layer uses the mathematical operation defined in equation 19 where Q_i represents the output feature map. The Rectified Linear Unit (ReLU) is utilized as the activation function for the Conv2D layer, as defined in equation 20.

$$Q_i(u, v) = \sum_{p=-l}^{l} \sum_{q=-l}^{q} J(u+p, v+q) K_i(p, q) \tag{19}$$

$$\mathrm{ReLU}(t) = \max(0, t) \tag{20}$$

b: PEAK AGGREGATION LAYER

The max-pooling method defined in equation 21 in the peak aggregation layer (PAK) was applied to down-sample the images with the most prevalent features. Here, the output image $V(u, v)$ is downsampled from the original image J.

$$V(u, v) = \max_{p=0}^{l-1} \max_{q=0}^{l-1} J(u \times l + p, v \times l + q) \tag{21}$$

c: ADDITIONAL EXTENSION LAYERS

In the initial phase, the proposed CNN faced some performance issues. However, this was resolved by adding an extended convolutional layer with 16 filters. After that, another max-pooling layer was added with 2 × 2 pool size. After this modification, the network demonstrated an overfitting nature. It was resolved by a dropout layer that randomly drops out 20% of the total number of hidden nodes. The dropout layer is defined by equation 22.

$$R(w) = \begin{cases} w, & \text{if chosen with probability } 1-q \\ 0, & \text{if chosen with probability } q \end{cases} \tag{22}$$

d: DENSE LAYER

The convolutional layer, in association with the max-pooling, extracts the image features F, which include both regular features and malicious features. These features are transmitted to the dense layer for learning and classification. The dense layer of the proposed CNN for the CPS security layer has two layers. The first layer has 128 nodes, and the second layer has 64 nodes. There is a dropout layer after the first dense layer with a 40% dropout rate. The operations of this layer are defined by equation 23.

$$z_i = \text{Activation}\left(\sum_{k=1}^{m} v_{ik} t_k + c_i\right) \tag{23}$$

e: OUTPUT LAYER

The CNN's output layer has seven nodes. Six nodes represent six types of intrusions, and one layer represents the benign class. The output layer maps the decision made by the dense layer onto a probability scale ranging from 0 to 1 using the Softmax activation function. The working principle of this layer is defined in equation 24.

$$\mathrm{softmax}(z_i) = \frac{\exp(z_i)}{\sum_{k=1}^{m} \exp(z_k)} \tag{24}$$

f: LOSS FUNCTION

The proposed CNN is a multiclass classifier with seven classes. A categorical cross-entropy loss function is recommended for such CNNs [41]. This loss function has been used in a CNN to quantify the difference between the ground truth and the predicted class. Equation 25 shows how it has been used in the experimental CNN where y represents the class label, $\hat{y}$ is the prediction, and m is the number of classes.

$$L(y, \hat{y}) = -\sum_{j=1}^{m} y_j \log(\hat{y}_j) \tag{25}$$

2) LSTM NETWORK ARCHITECTURE

The LSTM network designed in this paper aimed to classify four types of intrusions from textual data retrieved from network packets. It was trained with 43,345 instances optimized for generating accurate predictions.

a: INPUT LAYER

The input layer of the proposed LSTM network accepts sequential data segmented at a specific timestep with a sequence length of $M = 112$. The characteristics of the input layer are expressed in equation 26. The input layer transmits the incoming sequences to the next LSTM layer.

$$\text{Input Shape} = (M, 80) \tag{26}$$

b: LSTM LAYERS

The network's LSTM layer has been designed to capture information from long sequences and retain it as long as necessary. The proposed LSTM network has two LSTM layers. The first layer contains 200 LSTM nodes, and the second layer has 100 nodes. These two layers are defined by equations 27 and 28, respectively. In these equations, s_t is the input at time t, and $g_t^{(1)}$ and $m_t^{(1)}$ are the hidden state and cell state, respectively.

$$g_t^{(1)}, m_t^{(1)} = \text{LSTM}^{(1)}(s_t, g_{t-1}^{(1)}, m_{t-1}^{(1)}) \tag{27}$$

$$g_t^{(2)}, m_t^{(2)} = \text{LSTM}^{(2)}(g_t^{(1)}, g_{t-1}^{(2)}, m_{t-1}^{(2)}) \tag{28}$$

c: FULLY CONNECTED LAYERS

The information retrieved and retained by the LSTM nodes is transmitted to subsequent fully connected layers. There are two layers. The first layer has 100 nodes, and the second layer has 30 nodes. These two layers follow the working principles defined in the equations 29 and 30, respectively, where $V_d^{(1)}$ and $a_d^{(1)}$ represent the weights and biases.

$$z_t^{(1)} = \text{ReLU}(V_d^{(1)} g_t^{(2)} + a_d^{(1)}) \tag{29}$$

$$z_t^{(2)} = \text{ReLU}(V_d^{(2)} z_t^{(1)} + a_d^{(2)}) \tag{30}$$

d: OUTPUT LAYER

The final output layer contains five units representing the attack types (DDoS, DoS, brute force, and spoofing) and one benign class. The Softmax activation function is applied to convert the scores into class probabilities. This layer is defined in equation 31, where V_o and a_o represent the weights and biases. The working principle of the Softmax function is defined in equation 32.

$$\hat{z}_t = \text{softmax}(V_o z_t^{(2)} + a_o) \tag{31}$$

$$\text{softmax}(o_i) = \frac{\exp(o_i)}{\sum_{j=1}^{5} \exp(o_j)} \tag{32}$$

3) PARALLEL ORCHESTRATION (PLO) ALGORITHM

The LSTM and CNN function in parallel, ensuring CPS security against intrusion via textual and image data simultaneously. In this experiment, a novel algorithm, the PLO, has been developed to coordinate both the LSTM and CNN and make them function properly to defend against ten different types of intrusions. It has been presented as Algorithm 1. This algorithm runs in the onboard computers of the experimental CPS. It is initiated when the ICNC machines are started. It works as a security layer between ICNC software instruction and the hardware controlling signals of the LCC tools.

Initially, the PLO algorithm classifies image and non-image data into two categories. If the data is related to an image, it transmits the signals to the CNN, and if the data is non-image, the signals are passed to the LSTM network. Sometimes, the ICNC software sends both image and non-image data simultaneously. When that happens, the PLO algorithm activates both the CNN and LSTM at the same time. If any malicious code is detected in the image data, the PLO algorithm removes the image extension and generates an alarm at the control panel to notify the administrator. At the same time, it preserves the current state of the LCC tools in the log to resume progress from there. When intrusions are detected by the LSTM network, the PLO algorithm immediately blocks the network interface that transmits the malicious codes and generates an alarm to notify the administrator. The PLO algorithm also preserves the progress log for the LSTM network so that the tasks can be resumed exactly from where they were interrupted.

D. PHYSICAL-VIRTUAL ENVIRONMENT

The proposed methodology has been tested in a physical-virtual environment. The Prestige-60 used in this study is the physical device. A primary experiment was conducted on it. However, later, the real-world environment was simulated in Arena Simulation Software by replicating six Prestige-60 machines with the exact physical response data. The physical-virtual environment developed for this study has been illustrated in Figure 6. The physical machine performs with and without intrusions, and the performance data are stored in two different databases. Later, these production data are used to replicate the physical machine in a virtual environment. Finally, the data obtained from the simulated environment are stored in a separate database.

V. PERFORMANCE EVALUATION

The performance of the proposed CPS security method has been evaluated from three distinct perspectives. It detects and blocks intrusions from both image and text data. The first two perspectives involve the classification performance of intrusions from these two data types. The third perspective directly relates to the production rate and the corresponding annual revenue. This section provides a comprehensive analysis of the proposed security system's performance.

In this study, the focus has been primarily on the development and validation of the LSTM-CNN parallel orchestration algorithm aimed at enhancing CPS security. The results presented thus far are intended to demonstrate the algorithm's efficiency in a controlled environment, pertinent

Algorithm 1 Parallel Orchestration (PLO) Algorithm for Intrusion Detection

```
Input: $D_{in}$ (incoming data from ICNC machines)
Output: Intrusion detection and system security
Initialization:
    Initialize $PLO()$ when ICNC machines start
    Set $LCC\_log \leftarrow 0$
    Set status $\{CNN, LSTM\} \leftarrow$ {idle, idle}
Main Process:
while $ICNC_{running} =$ True do
    Data Classification:
        $\{D_{img}, D_{txt}\} \leftarrow Classify(D_{in})$
    if $D_{img} \neq \emptyset$ then
        $Output_{CNN} \leftarrow CNN(D_{img})$
    end if
    if $D_{txt} \neq \emptyset$ then
        $Output_{LSTM} \leftarrow LSTM(D_{txt})$
    end if
    if $D_{img} \neq \emptyset \wedge D_{txt} \neq \emptyset$ then
        $\{Output_{CNN}, Output_{LSTM}\} \leftarrow$
        $\{CNN(D_{img}), LSTM(D_{txt})\}$
    end if
    Intrusion Detection and Response:
    if $DetectMalicious(Output_{CNN}) =$ True then
        $RemoveExtension(D_{img})$
        $Alarm(ControlPanel)$
        $LCC\_log \leftarrow SaveState(LCC\_tools)$
    end if
    if $DetectIntrusion(Output_{LSTM}) =$ True then
        $BlockNetworkInterface(D_{txt})$
        $Alarm(ControlPanel)$
        $LSTM\_log \leftarrow SaveState(LSTM)$
    end if
end while
Conclusion:
    $ResumeState(LCC\_log, LSTM\_log)$
=0
```

to its ability to enhance security protocols within Industry 4.0 frameworks. Recognizing the importance of broader simulation results, the authors plan to explore and present extensive operational management aspects of the model. This follow-up study will include comprehensive discrete event simulations that detail the information flow and system-wide impacts, thereby providing a holistic view of the operational efficiencies the model can deliver.

A. EVALUATION METRICS

The accuracy, precision, recall (sensitivity), and F1 Score, which are mathematically expressed as 33–36, respectively, have been used to evaluate the performance of the LSTM network and the CNN. These metrics depend on true positive (TP), true negative (TN), false positive (FP), and false negative (FN). These values have been retrieved from confusion matrices illustrated in Figure 8 and Figure 9. The real-world performance of the proposed CPS security layer has been evaluated based on production, delay, unit lost, and revenue savings, which are later listed in Table 7.

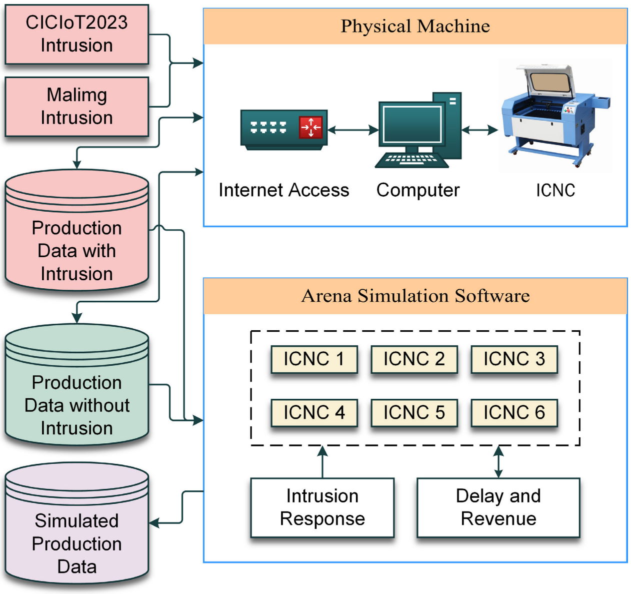


FIGURE 6. The experimental physical-virtual environment.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{33}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{34}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{35}$$

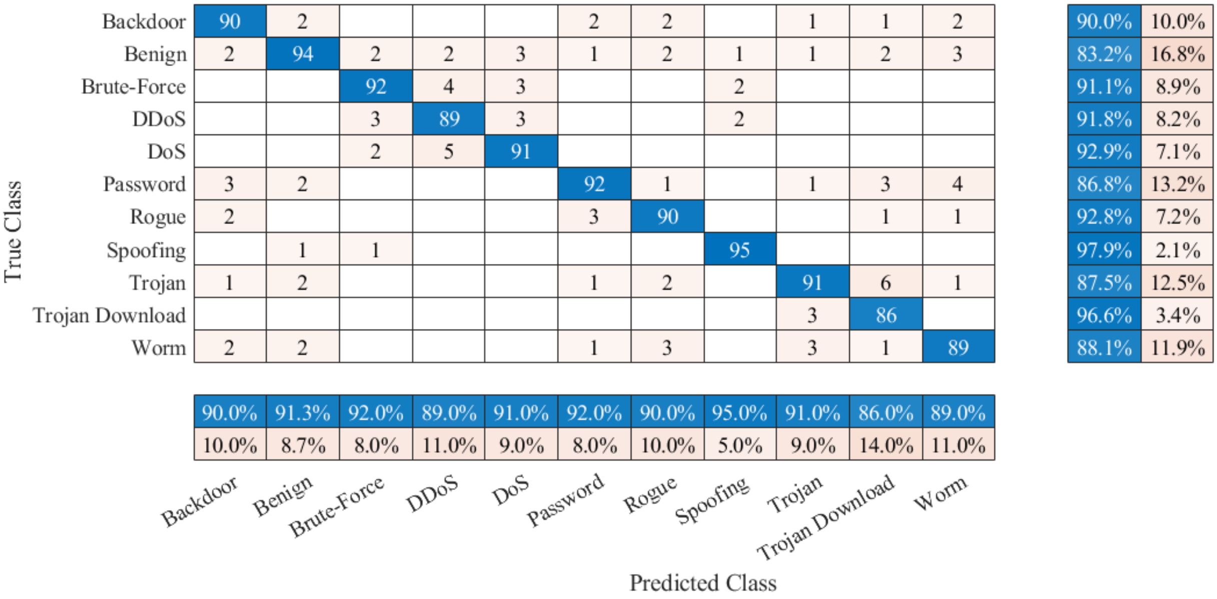
$$\text{F1 Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \tag{36}$$

B. OVERALL PERFORMANCE

The validation accuracies of the LSTM network and CNN are 91.66% and 90.85%, respectively. Both networks demonstrate stable performances that are good enough to be applied in real-world settings. The average precision, recall, and F1 score of the LSTM network are 92.02%, 92.04%, and 92.0%, respectively. For the CNN, these values are 90.03%, 90.17%, and 90.04%, respectively. The proposed system saves $30,474 yearly in revenue, by reducing the delay caused by intrusions from 116 hours to 5 hours only. This significant improvement reduces the yearly average unit loss from 184 to 15.

C. LSTM CONFUSION MATRIX ANALYSIS

The confusion matrix illustrated in Figure 8 demonstrates the effectiveness of the LSTM network in detecting various network intrusions. With an overall accuracy of 92.6%, the system provides comprehensive CPS security against brute force, DDoS, DoS, and spoofing attacks. The balance between precision and recall across all classes showcases the reliable detection capabilities of the proposed PLO algorithm.



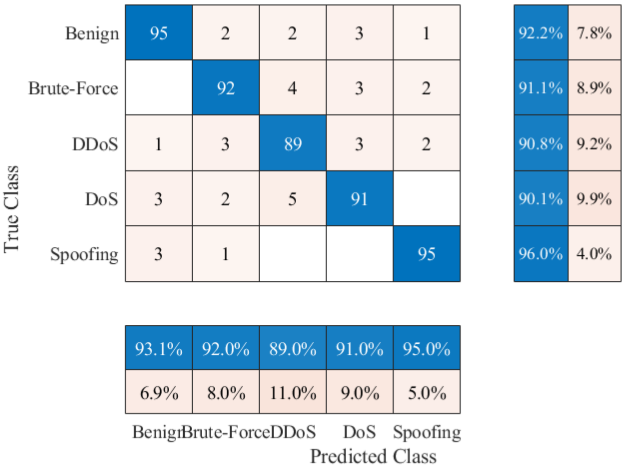
	Backdoor	Benign	Brute-Force	DDoS	DoS	Password	Rogue	Spoofing	Trojan	Trojan Download	Worm		
Backdoor	90	2				2	2		1	1	2	90.0%	10.0%
Benign	2	94	2	2	3	1	2	1	1	2	3	83.2%	16.8%
Brute-Force			92	4	3			2				91.1%	8.9%
DDoS			3	89	3			2				91.8%	8.2%
DoS			2	5	91							92.9%	7.1%
Password	3	2				92	1		1	3	4	86.8%	13.2%
Rogue	2					3	90			1	1	92.8%	7.2%
Spoofing		1	1					95				97.9%	2.1%
Trojan	1	2				1	2		91	6	1	87.5%	12.5%
Trojan Download									3	86		96.6%	3.4%
Worm	2	2				1	3		3	1	89	88.1%	11.9%
	90.0%	91.3%	92.0%	89.0%	91.0%	92.0%	90.0%	95.0%	91.0%	86.0%	89.0%		
	10.0%	8.7%	8.0%	11.0%	9.0%	8.0%	10.0%	5.0%	9.0%	14.0%	11.0%		

FIGURE 7. The overall performance of the proposed intrusion detection syste.

TABLE 3. Performance metrics for each intrusion class.

Intrusion Class	Precision (%)	Recall (%)	F1-Score (%)
Backdoor	90.0	90.0	90.0
Benign	91.3	83.2	87.0
Brute Force	92.0	91.1	91.5
DDoS	89.0	91.8	90.4
DoS	91.0	92.9	91.9
Password	92.0	86.8	89.3
Rogue	90.0	92.8	91.4
Spoofing	95.0	97.9	96.4
Trojan	91.0	87.5	89.2
Trojan Download	86.0	96.6	91.0
Worm	89.0	88.1	88.6

Table 4 presents the performance of the LSTM network numerically.



	Benign	Brute-Force	DDoS	DoS	Spoofing		
Benign	95	2	2	3	1	92.2%	7.8%
Brute-Force		92	4	3	2	91.1%	8.9%
DDoS	1	3	89	3	2	90.8%	9.2%
DoS	3	2	5	91		90.1%	9.9%
Spoofing	3	1			95	96.0%	4.0%
	93.1%	92.0%	89.0%	91.0%	95.0%		
	6.9%	8.0%	11.0%	9.0%	5.0%		

FIGURE 8. The confusion matrix generated from the LSTM network on a test dataset.

TABLE 4. Performance metrics for each intrusion class.

Intrusion Class	Precision (%)	Recall (%)	F1-Score (%)
Benign	93.1	92.2	92.6
Brute Force	92.0	91.1	91.5
DDoS	89.0	90.8	89.9
DoS	91.0	90.1	90.5
Spoofing	95.0	96.0	95.5

D. CNN CONFUSION MATRIX ANALYSIS

The confusion matrix depicted in Figure 9 demonstrates the effectiveness of the CNN network in detecting various network intrusions. With an overall accuracy of 89.6%, the system provides comprehensive CPS security against Backdoor, Benign, Password, Rogue, Trojan, Trojan Download, and Worm attacks. The balance between precision and recall across all classes showcases the reliable detection capabilities of the proposed PLO algorithm. The numerical values of different performance evaluation metrics are presented in Table 5.

E. K-FOLD CROSS VALIDATION

The performance inspection table using k-fold cross-validation illustrates the overall effectiveness of the proposed CPS security system. The results indicate a consistent performance across all folds, reflecting the robustness of the classification model. The accuracy across all folds ranges from 89.77% to 91.55%, demonstrating stable classification performance. The highest accuracy, 91.55%, was achieved in fold 5; whereas the lowest, 89.77%, was observed in fold 4. Precision values lie between 91.13% (fold 5) and 92.78% (fold 1), with fold 1 achieving the highest precision at 92.78%, highlighting its ability to correctly identify positive

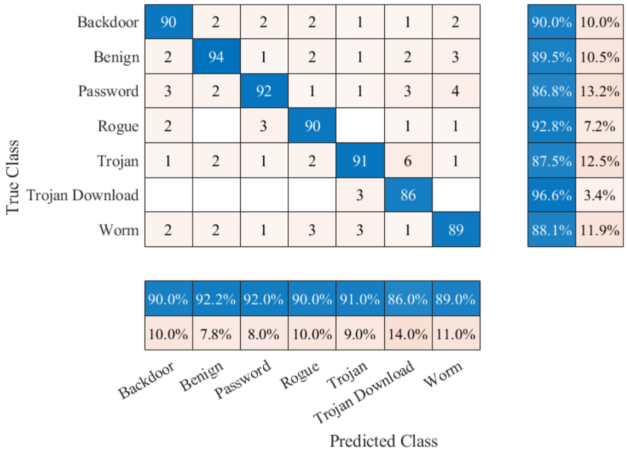


FIGURE 9. Confusion matrix analysis for CNN.

TABLE 5. Performance metrics for each intrusion class.

Intrusion Class	Precision (%)	Recall (%)	F1-Score (%)
Backdoor	90.0	90.0	90.0
Benign	92.2	89.5	90.8
Password	92.0	86.8	89.3
Rogue	90.0	92.8	91.4
Trojan	91.0	87.5	89.2
Trojan Download	86.0	96.6	91.0
Worm	89.0	88.1	88.6

instances. Recall values vary from 90.82% (fold 5) to 91.35% (fold 1), and the system consistently maintained a high recall across all folds, indicating its capability for detecting most positive instances. The F1-score ranges between 90.97% (fold 5) and 92.06% (fold 1), with fold 1 having the highest F1-score of 92.06%, balancing precision and recall effectively. The consistency of F1-scores across all folds demonstrates the reliability of the system. In summary, the proposed CPS security system shows a stable and robust performance with an average accuracy of 90.99%, precision of 91.72%, recall of 91.09%, and F1-score of 91.4%. This consistent performance across different folds underscores the reliability and effectiveness of the model.

F. PRODUCTION DELAY AND REVENUE

The experiment was conducted on six ICNC machines in a simulated industry which produces primarily tailored suits. The cost and delay were modeled using Arena simulation software. The performance analysis data are presented in Table 7. The maximum capacity of each machine is 5000 units per year. However, the actual amount produced varies, and is listed as Average Production in the table. Without the proposed CPS security layer, the simulated industry suffers from 58 successful intrusions on average per year. It causes an average of 116 hours of production delay, resulting in 184 units less of production. It causes around $33,191 revenue loss per year. However, after applying the proposed security system, the yearly successful intrusion rate is lowered to around 5, with only about 9 hours of production delay. As a result, it saves approximately $30,474 in revenue.

TABLE 6. Performance inspection using K-fold cross validation.

k	Accuracy	Precision	Recall	F1-Score
1	90.14	92.78	91.35	92.06
2	90.46	91.7	90.96	91.33
3	91.03	91.65	91.19	91.42
4	89.77	91.33	91.11	91.22
5	91.55	91.13	90.82	90.97

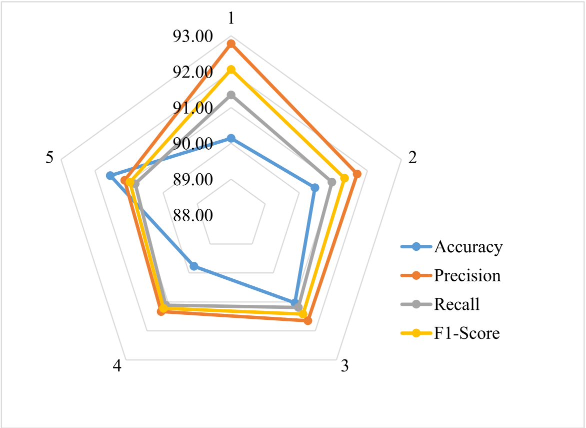


FIGURE 10. The variations among the precision, recall, and F1-score.

Figure 11 presents a comprehensive performance analysis of the effect of the proposed CPS security method on production and revenue generation from the manufacturing industry. Figure 11(a) compares the average production and revenue across six ICNC machines. Despite the maximum capacity being consistent at 5000 units per year, the actual amount of production varies, with ICNC 6 achieving the highest production of 4792 units and generating the highest revenue of $872,144. The ICNC 2 has the lowest production of 4453 units and the corresponding revenue of $810,446. Figure 11(b) illustrates the reduction in successful intrusions before and after implementing the PLO algorithm. On average, the number of successful intrusions drops from 58 to 5 per year per machine, highlighting the effectiveness of the proposed security system. Figure 11(c) showcases the production delays before and after the PLO algorithm implementation. The delayed hours due to intrusions significantly decrease, with ICNC 1 reducing from 195 to 11 hours and ICNC 4 from 160 to 7 hours. This reduction in delayed hours leads to a substantial increase in production efficiency. Figure 11(d) contrasts the revenue loss before and the revenue saved after implementing the PLO algorithm. The PLO algorithm saves a significant portion of revenue by preventing intrusions, with ICNC 1 saving $51,869 and ICNC 4 saving $43,042. Despite ICNC 2 incurring a relatively high revenue loss of $34,587 before PLO implementation, the machine managed to save $33,199 after applying the PLO algorithm. Overall, the figure clearly demonstrates the proposed CPS security system's positive impact on production, efficiency, and revenue across the ICNC machines.

TABLE 7. Performance of the proposed security system in a simulated manufacturing industry 4.0.

ICNC Machines	Max Capacity	Average Production	Revenue (USD)	Successful Intrusion	Delayed Hours	Unit Lost	Revenue Loss	Intrusion After PLO	Delayed Hour	Unit Lost	Revenue Saved (USD)
ICNC 1	5000	4576	832832	39	195	306	55006	4	11	17	51869
ICNC 2	5000	4453	810446	63	126	192	34587	5	5	8	33199
ICNC 3	5000	4711	857402	84	84	136	24394	5	13	21	20577
ICNC 4	5000	4566	831012	80	160	250	45035	4	7	11	43042
ICNC 5	5000	4782	870324	37	74	121	21814	9	8	13	19429
ICNC 6	5000	4792	872144	46	62	102	18315	7	12	20	14730

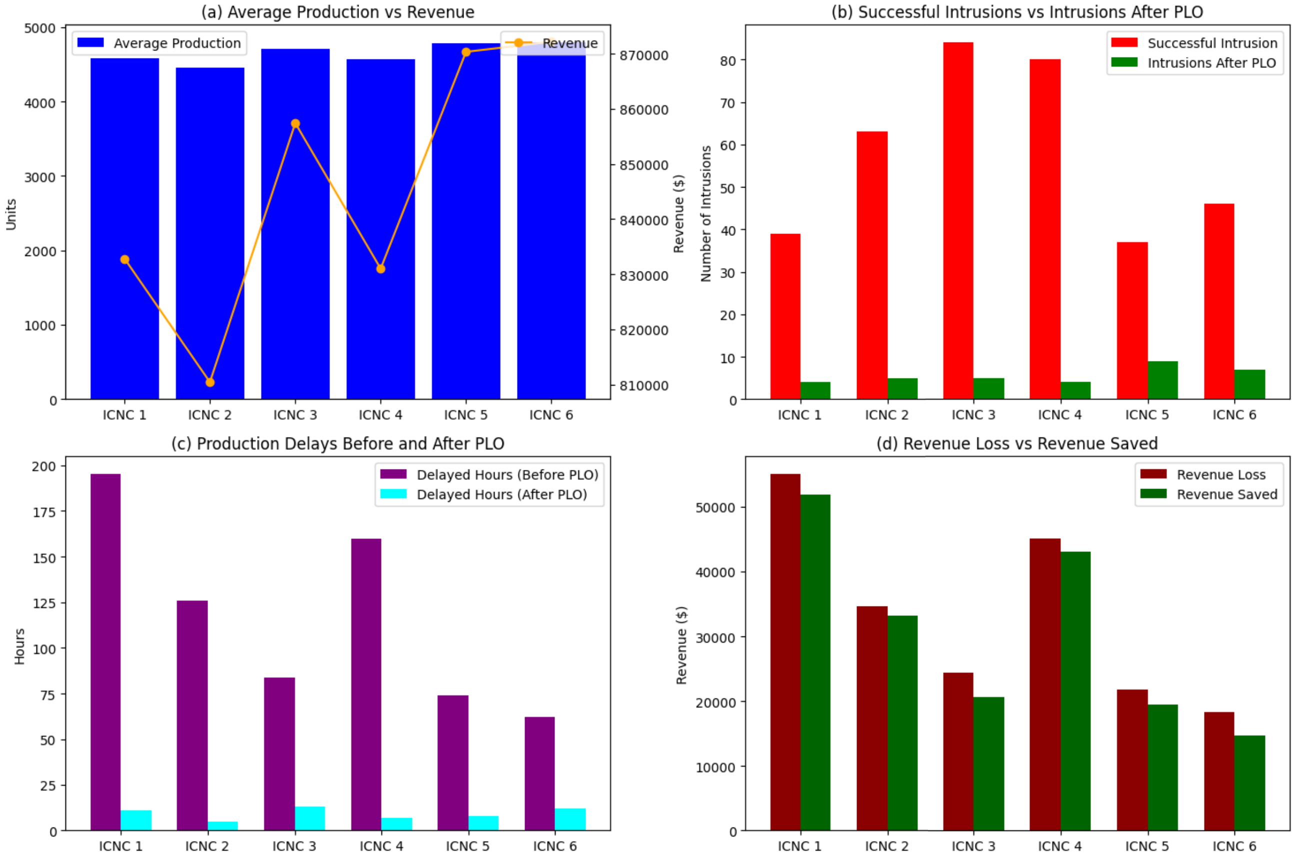


FIGURE 11. Performance analysis of ICNC machines in the CPS security system: (a) average production vs. revenue; (b) successful intrusions vs. intrusions after PLO; (c) production delays before and after PLO; (d) revenue loss vs. revenue saved.

VI. LIMITATIONS AND FUTURE DIRECTION

According to Razaque et al. [42], every security system has vulnerabilities. It is impractical to consider the proposed security system as an exception. The proposed methodology and experimental setup were critically explored to identify potential limitations. The weaknesses discovered have been highlighted in this section. These limitations are the scope of conducting further research and strengthening the security offered by it.

A. SIMULATED ENVIRONMENT

The experiment was conducted in a simulated environment. It would be more practical to apply the methodology in real-world industry production. However, the simulated environment was created using real-world data obtained from the physical ICNC machine. From this context, this limitation is mitigated. However, a comparison between the performances in real-world and simulated environments is necessary, and will be addressed in the future scope of this paper.

B. MALICIOUS INSIDER

The proposed system has no defense against malicious insiders. It has been implemented on the onboard computers of the ICNC machines. These computers are locally accessible. As a result, a malicious insider can turn off the protection layer. An internal status update report system would be beneficial to defend against such attacks, which would notify the operator if the security layer is compromised. The scope of integrating such features will be addressed in the future research of this paper.

C. ADVERSARIAL MACHINE LEARNING (AML) ATTACK

The proposed security system is an ML-based approach. The AML attack has become a new concern in the cyber security industry, which has not been addressed in this paper [43]. The proposed security system is defenseless against AML attacks. This weakness will be overcome in the future.

D. AUTOMATIC RISK ASSESSMENT

After detecting intrusion, the PLO algorithm pauses the manufacturing process to prevent a waste of resources caused by malicious signals and generates an alarm to alert the operator. However, it does not have any risk assessment module, which would be useful for characterizing the risk and mitigating it automatically without requiring any human intervention. This facility will be introduced in the subsequent upgrade of the PLO algorithm.

The CPS security layer presented in this paper is an excellent solution for manufacturing industries that use ICNC machines. Overcoming the limitations discussed in this section, the scope of further strengthening the security level opens the door to conducting more research in this field.

VII. CONCLUSION

Cyber-physical systems have made industrial production facilities faster, more efficient, and more productive. Data-driven decision-making, AI-assisted management, and access to the Internet have made today's manufacturing industry more flexible than ever. However, this flexibility has made it a popular playground for cybercriminals. A successful intrusion into a manufacturing industry can cause significant financial loss; therefore, CPS security has become one of the top priorities in Industry 4.0 research. This paper has introduced a practical solution to strengthen CPS security using deep learning technology.

The LSTM network and CNN parallel orchestration coordinated by the PLO algorithm protects the ICNC machines against intrusions conducted from both text and image data. Besides, if an intrusion is detected, it maintains the production progress logs to resume manufacturing without hampering the quality and precision. This innovative approach reduces the production delay and saves the industry from losing revenue. The PLO algorithm detects intrusions from text and images with 91.66% and 90.85% accuracy, respectively. The simulated results show that it helps save $30,474 annually when the machines function at their maximum capacity. It reduces the number of successful intrusions from 184 to only 15. As a result, it reduces the 95.69% production delay caused by intrusions.

Despite its outstanding performance and a significant positive impact on the production rate, the CPS security layer developed in this paper suffers from several limitations. The experiment was conducted in a real-life-like simulated environment, which does not include all the uncertainty of the actual environment. Besides, it does not have any defense against the malicious insider. Moreover, it is not capable of avoiding AML attacks. These limitations pave the way for more experiments on this security system, further strengthening it, and establishing it as one of the most practical security solutions for CPS in the manufacturing industry.

REFERENCES

[1] S. Munirathinam, "Industry 4.0: Industrial Internet of Things (IIOT)," in *Advances in Computers*, vol. 117. Amsterdam, The Netherlands: Elsevier, 2020, pp. 129–164.

[2] M. Kerin and D. T. Pham, "A review of emerging Industry 4.0 technologies in remanufacturing," *J. Cleaner Prod.*, vol. 237, Nov. 2019, Art. no. 117805.

[3] H. Ravichandran, *Intelligent Safety: How to Protect Your Connected Family From Big Cybercrime*. New York, NY, USA: Simon and Schuster, 2023.

[4] H. Kayan, M. Nunes, O. Rana, P. Burnap, and C. Perera, "Cybersecurity of industrial cyber-physical systems: A review," *ACM Comput. Surv.*, vol. 54, no. 11s, pp. 1–35, Jan. 2022.

[5] M. F. Franco, F. Künzler, J. von der Assen, C. Feng, and B. Stiller, "RCVaR: An economic approach to estimate cyberattacks costs using data from industry reports," *Comput. Secur.*, vol. 139, Apr. 2024, Art. no. 103737.

[6] Z. Humienny, "New ISO geometrical product specification standards as a response to Industry 4.0 needs," in *Proc. 5th Int. Conf. Ind. 4.0 Model Adv. Manuf., Acad. Manage. Perspect.* Cham, Switzerland: Springer, Jan. 2020, pp. 306–312.

[7] S. M. Zahraee, S. R. Golroudbary, A. Hashemi, J. Afshar, and M. Haghighi, "Simulation of manufacturing production line based on arena," *Adv. Mater. Res.*, vol. 933, pp. 744–748, May 2014.

[8] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, p. 5941, Jun. 2023.

[9] K. A. Asmitha, V. Puthuvath, K. A. R. Rehiman, and S. L. Ananth, "Deep learning vs. Adversarial noise: A battle in malware image analysis," *Cluster Comput.*, vol. 27, no. 7, pp. 9191–9220, Oct. 2024.

[10] D. G. S. Pivoto, L. F. F. de Almeida, R. da Rosa Righi, J. J. P. C. Rodrigues, A. B. Lugli, and A. M. Alberti, "Cyber-physical systems architectures for industrial Internet of Things applications in Industry 4.0: A literature review," *J. Manuf. Syst.*, vol. 58, pp. 176–192, Jan. 2021.

[11] S. J. Oks, M. Jalowski, M. Lechner, S. Mirschberger, M. Merklein, B. Vogel-Heuser, and K. M. Möslein, "Cyber-physical systems in the context of Industry 4.0: A review, categorization and outlook," *Inf. Syst. Frontiers*, vol. 26, no. 5, pp. 1731–1772, Oct. 2024.

[12] G. Lampropoulos and K. Siakas, "Enhancing and securing cyber-physical systems and Industry 4.0 through digital twins: A critical review," *J. Softw., Evol. Process*, vol. 35, no. 7, p. 2494, Jul. 2023.

[13] K. Manasa and L. M. I. L. Joseph, "IoT security vulnerabilities and defensive measures in Industry 4.0," in *Advanced Technologies and Societal Change*. Cham, Switzerland: Springer, 2023, pp. 71–112.

[14] B. Li, Y. Wu, J. Song, R. Lu, T. Li, and L. Zhao, "DeepFed: Federated deep learning for intrusion detection in industrial cyber–physical systems," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5615–5624, Aug. 2021.

[15] P. O'Donovan, C. Gallagher, K. Bruton, and D. T. J. O'Sullivan, "A fog computing industrial cyber-physical system for embedded low-latency machine learning Industry 4.0 applications," *Manuf. Lett.*, vol. 15, pp. 139–142, Jan. 2018.

[16] S. Sharma and K. Guleria, "Machine learning techniques for intelligent vulnerability detection in cyber-physical systems," in *Proc. Int. Conf. Data Analytics Bus. Ind. (ICDABI)*, Oct. 2022, pp. 200–204.

[17] A. S. Rajawat, S. B. Goyal, P. Bedi, N. B. Constantin, M. S. Raboaca, and C. Verma, "Cyber-physical system for industrial automation using quantum deep learning," in *Proc. 11th Int. Conf. Syst. Model. Advancement Res. Trends (SMART)*, Dec. 2022, pp. 897–903.

[18] U. Awan, L. Hannola, A. Tandon, R. K. Goyal, and A. Dhir, "Quantum computing challenges in the software industry. A fuzzy AHP-based approach," *Inf. Softw. Technol.*, vol. 147, Jul. 2022, Art. no. 106896.

[19] M. Abdullahi, H. Alhussian, N. Aziz, S. J. Abdulkadir, and Y. Baashar, "Deep learning model for cybersecurity attack detection in cyber-physical systems," in *Proc. 6th Int. Conf. Comput., Commun., Control Autom. (ICCUBEA*, Aug. 2022, pp. 1–5.

[20] R. Alguliyev, Y. Imamverdiyev, and L. Sukhostat, "Hybrid DeepGCL model for cyber-attacks detection on cyber-physical systems," *Neural Comput. Appl.*, vol. 33, no. 16, pp. 10211–10226, Aug. 2021.

[21] M. Alabadi and A. Habbal, "Next-generation predictive maintenance: Leveraging blockchain and dynamic deep learning in a domain-independent system," *PeerJ Comput. Sci.*, vol. 9, p. e1712, Dec. 2023.

[22] Y. Wu, H. Cao, G. Yang, T. Lu, and S. Wan, "Digital twin of intelligent small surface defect detection with cyber-manufacturing systems," *ACM Trans. Internet Technol.*, vol. 23, no. 4, pp. 1–20, Nov. 2023.

[23] I. W. R. Taifa, S. G. Hayes, and I. D. Stalker, "Computer modelling and simulation of an equitable order distribution in manufacturing through the Industry 4.0 framework," in *Proc. Int. Conf. Electr., Commun., Comput. Eng. (ICECCE)*, Jun. 2020, pp. 1–6.

[24] A. Dogan and D. Birant, "Machine learning and data mining in manufacturing," *Expert Syst. Appl.*, vol. 166, Mar. 2021, Art. no. 114060.

[25] S. R. Burri, S. Ahuja, A. Kumar, and A. Baliyan, "Exploring the effectiveness of optimized convolutional neural network in transfer learning for image classification: A practical approach," in *Proc. Int. Conf. Advancement Comput. Comput. Technol. (InCACCT)*, May 2023, pp. 598–602.

[26] X. Lei, H. Pan, and X. Huang, "A dilated CNN model for image classification," *IEEE Access*, vol. 7, pp. 124087–124095, 2019.

[27] P. Gambini, M. Renaud, C. Guillemot, F. Callegati, I. Andonovic, B. Bostica, D. Chiaroni, G. Corazza, S. L. Danielsen, P. Gravey, P. B. Hansen, M. Henry, C. Janz, A. Kloch, R. Krahenbuhl, C. Raffaelli, M. Schilling, A. Talneau, and L. Zucchelli, "Transparent optical packet switching: Network architecture and demonstrators in the KEOPS project," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 7, pp. 1245–1259, Sep. 1998.

[28] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: Hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837–99849, 2022.

[29] S. M. Al-Selwi, M. F. Hassan, S. J. Abdulkadir, and A. Muneer, "LSTM inefficiency in long-term dependencies regression problems," *J. Adv. Res. Appl. Sci. Eng. Technol.*, vol. 30, no. 3, pp. 16–31, May 2023.

[30] H. Zhao, S. Sun, and B. Jin, "Sequential fault diagnosis based on LSTM neural network," *IEEE Access*, vol. 6, pp. 12929–12939, 2018.

[31] Z. Qin, S. Yang, and Y. Zhong, "Hierarchically gated recurrent neural network for sequence modeling," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2023.

[32] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," *Phys. D, Nonlinear Phenomena*, vol. 404, Mar. 2020, Art. no. 132306.

[33] S. Huda, K. Liu, M. Abdelrazek, A. Ibrahim, S. Alyahya, H. Al-Dossari, and S. Ahmad, "An ensemble oversampling model for class imbalance problem in software defect prediction," *IEEE Access*, vol. 6, pp. 24184–24195, 2018.

[34] S. Susan and A. Kumar, "The balancing trick: Optimized sampling of imbalanced datasets—A brief survey of the recent state of the art," *Eng. Rep.*, vol. 3, no. 4, p. 12298, Apr. 2021.

[35] S. M. Kasongo and Y. Sun, "A deep learning method with wrapper based feature extraction for wireless intrusion detection system," *Comput. Secur.*, vol. 92, May 2020, Art. no. 101752.

[36] Y. Xu, G. J. Jones, J. Li, B. Wang, and C. Sun, "A study on mutual information-based feature selection for text categorization," *J. Comput. Inf. Syst.*, vol. 3, no. 3, pp. 1007–1012, 2007.

[37] A. H. Mirza and S. Cosan, "Computer network intrusion detection using sequential LSTM neural networks autoencoders," in *Proc. 26th Signal Process. Commun. Appl. Conf. (SIU)*, May 2018, pp. 1–4.

[38] L. Dong, D. Fang, X. Wang, W. Wei, R. Damaševičius, R. Scherer, and M. Woźniak, "Prediction of streamflow based on dynamic sliding window LSTM," *Water*, vol. 12, no. 11, p. 3032, Oct. 2020.

[39] D. Gibert, C. Mateu, J. Planes, and R. Vicens, "Using convolutional neural networks for classification of malware represented as images," *J. Comput. Virol. Hacking Techn.*, vol. 15, no. 1, pp. 15–28, Mar. 2019.

[40] L. Nataraj, V. Yegneswaran, P. Porras, and J. Zhang, "A comparative assessment of malware classification using binary texture analysis and dynamic analysis," in *Proc. 4th ACM Workshop Secur. Artif. Intell.*, Oct. 2011, pp. 21–30.

[41] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, Dec. 2018, pp. 8792–8802.

[42] A. Razaque, F. Amsaad, M. J. Khan, S. Hariri, S. Chen, C. Siting, and X. Ji, "Survey: Cybersecurity vulnerabilities, attacks and solutions in the medical domain," *IEEE Access*, vol. 7, pp. 168774–168797, 2019.

[43] I. Alsmadi, N. Aljaafari, M. Nazzal, S. Alhamed, A. H. Sawalmeh, C. P. Vizcarra, A. Khreishah, M. Anan, A. Algosaibi, M. A. Al-Naeem, A. Aldalbahi, and A. Al-Humam, "Adversarial machine learning in text processing: A literature survey," *IEEE Access*, vol. 10, pp. 17043–17077, 2022.

SALMAN SAEIDLOU is a Principal Lecturer/an Associate Professor in mechanical/material engineering with the School of Engineering Technology and Design, Canterbury Christ Church University (CCCU), U.K. He is also a Senior Fellow of the Higher Education Academy (SFHEA) and has broad ranging teaching and supervision experience in undergraduate and postgraduate engineering courses with the U.K. higher education institutions. His research interests include intelligent manufacturing systems, distributed systems, agent-based modeling, big data analytics in manufacturing, data mining, and machine learning. He is a Chartered Mechanical Engineer (C.Eng., MIMechE).

NIKDOKHT GHADIMINIA received the B.Eng. and M.Sc. (Hons.) degrees in civil engineering from the University of Birmingham, U.K., and the Ph.D. degree in security-minded digital transformation in the built environment from Birmingham City University. She is the Chartered Construction Manager and a Senior Lecturer with a strong focus on digital technology. She is a Researcher and the co-author with the IoT Security Foundation (IoTSF) and also part of the leading cybersecurity team in the built environment working group. As a Chartered Member of CIOB (MCIOB), a Future Leader's CIOB Board Member, and a fellow of the Higher Education Academy (FHEA), she is actively involved in research in digital construction, digital twins, BIM, and cybersecurity in the built environment.

KWADWO OTI-SARPONG is currently a Senior Research and Teaching Associate in urban systems and infrastructure management with Cambridge Centre for Smart Infrastructure and Construction (CSIC), Department of Engineering, University of Cambridge. In this role, he conducts research into responsible placed based leadership for developing and implementing digital innovations in the urban built environment to deliver public value. He also leads the development of educational material focused on Leadership of Urban Digital Innovation for Public Value (LeadUP) to be run with the University of Cambridge.