Hindawi
Scientific Programming
Volume 2021, Article ID 3406612, 11 pages
https://doi.org/10.1155/2021/3406612

*Research Article*

# Research on Traffic Detection Method of Secure Transmission Industrial Internet of Things Based on Computer Vision

**Donghui Yang**

*Department of Business Administration, Kunlun Tourism College, Heilongjiang Institute of Technology, Harbin 150000, China*

Correspondence should be addressed to Donghui Yang; yangdonghui1998@s.hrbcu.edu.cn

Received 20 October 2021; Revised 24 November 2021; Accepted 3 December 2021; Published 20 December 2021

Academic Editor: Le Sun



In recent years, the process of industrial modernization has intensified, traditional industrial control has been improved and rapidly developed, industrial automation and intelligent unmanned production lines have become a new development trend, and the Internet of Things has become the basic direction of industrial development. In order to improve the effect of safe transmission and industrial IoT traffic detection, this study uses a neural network to improve the industrial IoT traffic detection algorithm. In order to improve the visualization effect of monitoring, this study uses computer vision technology to construct a traffic detection system of secure transmission industrial Internet of Things and builds an intelligent detection model. Finally, this study combines experimental research to verify the performance of the system. From the statistical point of view, it can be seen that the system's security detection and traffic detection effects are very good.

## 1. Introduction

In an industrial production environment, IoT sensor devices are often used to capture data to monitor and adjust the production operation process. The data generated by these devices are collected and organized in different ways and used for various purposes [1]. The transmission speed of IoT sensors is very fast, and the application of a large number of sensor devices will definitely lead to a substantial increase in the output of industrial data. The Internet of Things and big data are closely linked, and the data generated by sensors can also be processed by the big data platform [2]. Industrial Internet of Things and big data are different from internet big data. In addition to the general characteristics of big data, they also have strong relevance and timing. Therefore, traditional Internet big data processing methods are not fully applicable, and new solutions need to be designed specifically to properly analyze IoT data and extract more important information from IoT monitoring equipment. The proposal of "Industry 4.0" [3] has promoted the rapid development of global industries toward intelligence. In order to comply with the intelligent development of the new generation of industries, the number of IoT devices used in industrial production environments is also increasing. Moreover, the amount of data collected by sensors is exponentially increasing [4], so it is necessary to find more effective processing methods for these large-scale industrial data.

Usually, data analysis needs to move a subset of data to a data warehouse, and the speed of data analysis in Hadoop is very slow. However, with the development of SQL query engines, big data technology can already be used in business analysis scenarios. By building a data model in Hadoop or other databases, the original data are turned into meaningful indicators, and the large-scale historical data accumulated and stored for a long time are used in the big data processing system for information mining. The main purpose of adopting a distributed query strategy and using a memory-based computing method is to quickly query information in massive data, give timely feedback to users, and improve query efficiency.

Based on the above analysis, this study combines computer vision to conduct research on safe transmission and industrial Internet of Things traffic detection methods to further improve industrial production safety and enhance the application effect of the Internet of Things in the industry.

## 2. Related Work

Led by the transformation and upgradation of the manufacturing industry, a new round of industrial revolution has started around the world [5]. The application of industrial big data will greatly promote and optimize industrial production efficiency and management. In order to provide better services to users, industrial production should save and mine these data. For industrial information with high data volume and updated speed characteristics, data acquisition and analysis will face certain difficulties. In the processing of industrial IoT big data, various application frameworks have been mentioned in many documents. The literature [6] proposed an industrial big data ingestion and analysis platform (IBDP), which integrates HDFS, Spark, Hive, HBase, Flume, Sqoop, OpenStack, etc., and is suitable for industrial data ingestion and analysis. The literature [7] proposed and developed a smart city system based on the Internet of Things using Hadoop ecosystem use and big data analysis technology and combined with Spark over Hadoop to achieve the efficiency of big data processing. The literature [8] proposed the use of the Hadoop software environment, including data collection, data storage, data normalization and analysis, and data visualization components to realize the parallel processing of large heterogeneous data for IoT network security monitoring. The value of industrial big data has been globally recognized. Therefore, how to combine big data processing technology to store, manage, and analyze industrial big data is a topic of widespread concern for domestic and foreign researchers. In the development of big data in the industrial Internet of Things, China still has shortcomings in terms of format specification, platform technology integration, informatization strategy, and security. In order to better promote the development of industrial big data, it is necessary to realize the unification of data standardization, big data collection, multisource data processing, and information mining analysis and realize the visualization of data in the industrial production process and increase the usability of industrial data. There are many unknown possibilities for the integration between the Industrial Internet of Things and big data technologies, and the corresponding platform framework and technical support are not unique. Through more scientific and in-depth research, and constantly changing needs, the industrial IoT big data operating platform will also be continuously improved due to technological progress [9].

As products have higher and higher requirements for production equipment and processes, making production equipment and production lines more and more complex, remote intelligent control and supervisory analysis become more and more important [10]. The Internet of Things remote data transmission system can automatically upload equipment operating data to the Internet, finally, collect, process, and store in the database in the Internet data service center, then transform and disassemble the data information, and finally use intuitive graphs and line graphs, Pie charts, and tables, etc., so that engineering supervisors can quickly obtain data transmission information and accurately control the production and operation of machinery and equipment. Engineering monitoring personnel can view the status of production equipment through the remote data transmission system of the Internet of Things. In addition, the system can set functions such as abnormal prompts, fault alarms, and preliminary cause investigation [11]. The modern industrial Internet of Things remote data transmission system is mainly composed of a remote monitoring center, a mobile device data transmission terminal, a wireless transmission module, and an intelligent collection terminal. Each module is connected through the Internet, mobile communication base stations, GPRS networks, local wireless transmission equipment, and sensors and connected and communicated with smart collection terminals [12].

However, in dealing with the problems of multivariate and big data in the industry, many data quality problems are often encountered, which will greatly affect the effect of analysis and processing. Traditional machine learning methods are often at a loss for this. In comparison, the application of ensemble learning in this area highlights a huge advantage. Gradient boosting decision tree [13] (GBDT) is a decision tree algorithm based on iterative construction in integrated learning. It has an excellent performance in the practice of processing industrial data. Literature [14] studies the gradient boosting decision tree algorithm, using a large amount of collected power load data for model training, generating a new decision tree above the negative direction of the loss function gradient, and optimizing the prediction accuracy of short-period negative power load. XGBoost (Extreme Gradient Boosting) is based on the optimization of the GBDT algorithm [15]. It can make full use of the CPU's multithreading to perform parallel calculations and, at the same time, optimize and improve the algorithm to a certain extent, which improves the accuracy of the model. XGBoost combines many single decision tree models with low classification accuracy to form a tree model with relatively high classification accuracy. Each time a single weak learner is trained, the weight of the last residual is first increased, then the learning of the current learner is performed, then the previous residual is adjusted by adding a new weak learner, finally, the weight of multiple learners is calculated, and the final result is predicted. Literature [16] uses the XGBoost algorithm to improve the accuracy of rolling bearing fault diagnosis; literature [17] applies the algorithm to the quality inspection stage of the manufacturing industry to achieve the purpose of accurately predicting product quality.

## 3. Traffic Detection of Secure Transmission Industrial Internet of Things

In the case of unsupervised and undesired output, the self-organizing competitive network is a neural network based on unsupervised learning. It is mainly through observing, analyzing, and comparing the characteristics of objective things, prompting the inner law by itself, and classifying according to the similarity of this characteristic. This kind of network is similar to the learning mode of the biological neural network in the human brain, which both accurately discover the type of sample through the extraction of

features and adjust the network parameters through the corresponding learning algorithm.

Competitive networks are generally divided into the input layer and the competition layer. Its network structure is shown as in Figure 1.

We assume that the input layer and the competition layer of the network each have I and C neurons, the link weight between the two layers of nodes is $\omega_{ij}\,(i = 1, 2, 3, \ldots, I\,;\, j = 1, 2, 3, \ldots, C)$, and the arithmetic sum of the ownership values is 1, that is, $\sum_{i=1}^{N} \omega_{ij} = 118$.

The input samples of the competitive network are all binary vectors (O or 1). Only one neuron in each input sample wins and assigns its corresponding label. Different competing layer neurons represent different classifications. The calculation method of the competitive layer neuron $j$ is as follows:

$$S_j = \sum_{i=1}^{N} \omega_{ij} x_i. \tag{1}$$

When a neuron in the competition layer has the largest weight value, the neuron wins. According to the corresponding strategy, the input of the winning neuron is 1, and the rest are 0. As shown in the formula,

$$a_k = \begin{cases} 1, & S_k > S_j,\ \forall j,\ k \neq j, \\ 0, & \text{else.} \end{cases} \tag{2}$$

The neuron that wins the competition will be trained and learned according to the input vector. For all the input layer neuron $i$, there is the following formula:

$$\omega_{ij} = \omega_{ij} + a\left(\frac{x_i}{m} - \omega_{ij}\right). \tag{3}$$

In the formula, a is the learning rate, which generally satisfies $0 < a << l$, and $m$ is the total number of neurons whose output is 1 in the input layer and satisfies $m = \sum_{i=1}^{N} x_i$. This formula shows that if $x_i$ is active, its corresponding $i$-th weight will increase; otherwise, it will decrease [19].

The self-organizing feature map model SOM (self-organizing feature map) makes nearby neurons compete with each other to solve the problem of external information forming concepts in the self-organization of the human brain. For a system, it is to solve the corresponding expression of internal self-organization when a system accepts external roles. This is expressed as the adjustment of the weight coefficient in the artificial neural network.

The SOM network can be multidimensional. For the convenience of description, a two-dimensional planar competition layer array is taken as an example. The $n$ neurons in the input layer are fully connected with the axb neurons in the competition layer. Additionally, axb neurons in the competition layer are also directly or indirectly connected, and neurons in adjacent areas inhibit each other. Its network structure is shown in Figure 2.

The data that enter the SOM network through the input layer will be selectively given the corresponding response by the network through the corresponding strategy. The specific steps are as follows.

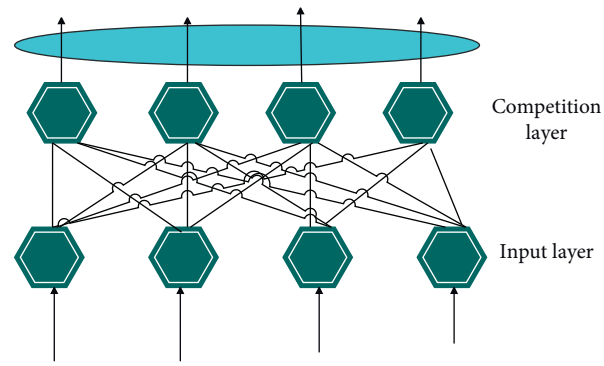


Figure 1: Self-organizing competition network topology diagram.

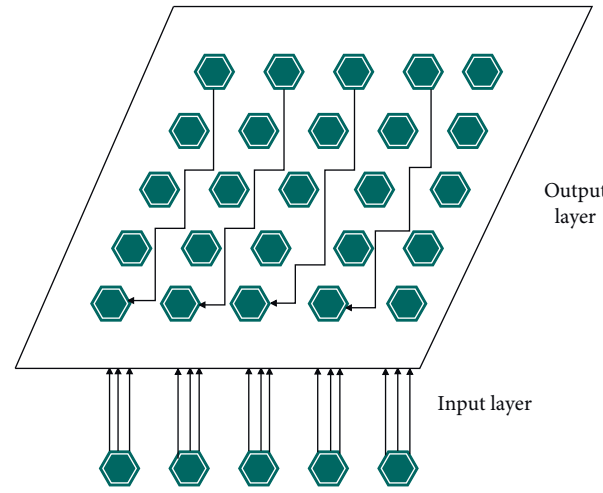


Figure 2: SOM neural network topology diagram.

*3.1. Network Initialization.* The weight is randomly assigned to the initial value of the weight, which is usually small. SJ is a group of J-neighboring neurons, that is, the number of adjacent neurons continues to decrease.

*3.2. Calculating the Euclidean Distance between the Weight Vector of the Mapping Layer and the Input Vector.* In the competition layer, the algorithm calculates the Euclidean distance between the weight vector of each neuron and the input vector. The distance between the $j$ neuron and the input vector is as follows:

$$d_j = \left\| X - W_j \right\| = \sqrt{\sum_{i=1}^{m} \left( x_i(t) - \omega_{ij}(t) \right)^2}. \tag{4}$$

The algorithm takes the neuron with the smallest distance obtained as the winning neuron and records it as $j^*$. For any competitive layer neuron $j$, there is a specific $k$ that satisfies $d_k = \min(d_j)$.

*3.3. Learning of Weights.* According to the input vector, the algorithm trains the weights of $j^*$ and its neighboring neurons as follows:

$$\begin{aligned}\Delta\omega_{ij} &= \omega_{ij}(t+1) - \omega_{ij}(t) \\ &= \eta(t)\left(x_i(t) - \omega_{ij}\right).\end{aligned} \tag{5}$$

Among them, the value of $\eta$ is $0 < \eta < 1$. As time increases, its value gradually approaches 0.

### 3.4. Calculating Output $O_k$

$$o_k = f\left(\min_j \left\| X - W_j \right\|\right). \tag{6}$$

Among them, $f(*)$ is a linear function.

### 3.5. If the Expected Setting Is Not Reached, the Algorithm Returns to Step (2).

The data collected by the model in this study are stored in a sliding queue-based buffer set by the cluster head node, as shown in Figure 3. In the entire interactive cycle, feature sampling is performed on the interactive data stream, and the length of the sliding queue $n$ is determined by the two factors discussed above. Once the queue length is exceeded, the element at the head of the queue is taken out of the queue. Figure 3 details this process.

The algorithm evaluates the collected feature values, including repetition rate evaluation, time regularity evaluation, and abnormal data change ratio evaluation.

#### 3.5.1. Evaluation of Data Repetition Rate.

In general, normal nodes will not continuously send data with a higher repetition rate. The higher the data repetition rate (rep), the smaller the evaluation will be with the increase in rep, and the rate of change should rise. That is, when the rep is getting closer and closer to the critical value of 0, its evaluation is getting lower and lower. This changing trend can be expressed by an exponential function with a base greater than 1, as shown in the following formula [20]:

$$R_1 = \begin{cases} 2 - \beta^{\text{rep}}, & \text{rep} < \theta, \\ 0, & \text{rep} > \theta. \end{cases} \tag{7}$$

Among them, $2 - \beta^{\text{rep}} > 0$, and the critical value 0 is determined by the specific network.

#### 3.5.2. Time Regularity Evaluation.

The regularity of time is evaluated here. ti represents the i-th time interval and normalizes each time interval. This study considers the dispersion of time intervals. The greater the dispersion, the less regularity, and the lower the possibility of it being a malicious node. The smaller the dispersion, the stronger the regularity of the time interval, and it may be a malicious node. The formula is as follows:

$$R_2 = \frac{\sum_{i=1}^{n-1}(t_i - \mu)}{n-1}. \tag{8}$$

When $n = 1$, there is only one interval, which means there is no regularity; then, $R_2 = 1$.

#### 3.5.3. Evaluation of Abnormal Data Parameter Change Ratio.

Without loss of generality, the abnormal data parameter here can be a variety of parameters, including the number of streams corresponding to the substream, the number of packets, the number of bytes, and other parameters. According to different application scenarios or network conditions, it can set one or more parameters to participate in the evaluation. We assume that the parameter at this time uses the number of bytes sent. If a DoS attack is sent during node interaction, a relatively large change will inevitably occur in the amount of data. DoS attacks must continue to attack to achieve results. However, the current DoS attack methods are endless. In order to conceal the attack behavior, it will mix the attack data stream with the normal data stream, which is difficult to be found. The data change rate $X$ is its ratio to the previous data change. It is calculated as follows:

$$X = \frac{|m - \overline{m}|}{m_{\text{var}}}. \tag{9}$$

We assume that $m$ is the feature parameter of the new queue node. There are $k$ historical records in front of it, $\overline{m}$ is the expectation of the previous $k$ historical data, and $m_{\text{var}}$ represents the average value of the previous $k$ data changes. It is calculated as follows [21]:

$$m_{\text{var}} = \sum_{i=1}^{k} \frac{|m_i - \overline{m}|}{k}. \tag{10}$$

The algorithm determines the interval of the parameter under normal conditions according to specific application scenarios and normal interaction data and sets the critical value $\sigma$. The algorithm calculates the number of $n$ data parameters that does not exceed $\sigma$ and uses its proportion as an evaluation of the change ratio of abnormal data parameters.

$$R_3 = 1 - \frac{x}{n}. \tag{11}$$

On the basis of obtaining the above evaluation indicators, the algorithm performs weighted aggregation on them. The algorithm uses weighting coefficients $W1$, $W2$, $W3$, and $W1 + W2 + W3 = 1$. At this point, the node behavior evaluation is as follows:

$$R = W_1 R_1 + W_2 R_2 + W_3 R_3. \tag{12}$$

When the core layer information management center of the Internet of Things network receives the service request sent by the user, it sends the request data to the subordinate management node, and the subordinate agent node has individual requirements for the type of service.

To select and authorize the sensor layer nodes, for the trust management module studied in this study, the basis of authorization is the evaluation of trust. It includes two types of trust factor-historical statistical trust value and recommended trust value. The algorithm calculates the static trust degree of this interaction through the weighted synthesis of two types of trust factors and decides whether to authorize or not according to the security policy by comparing with the threshold.

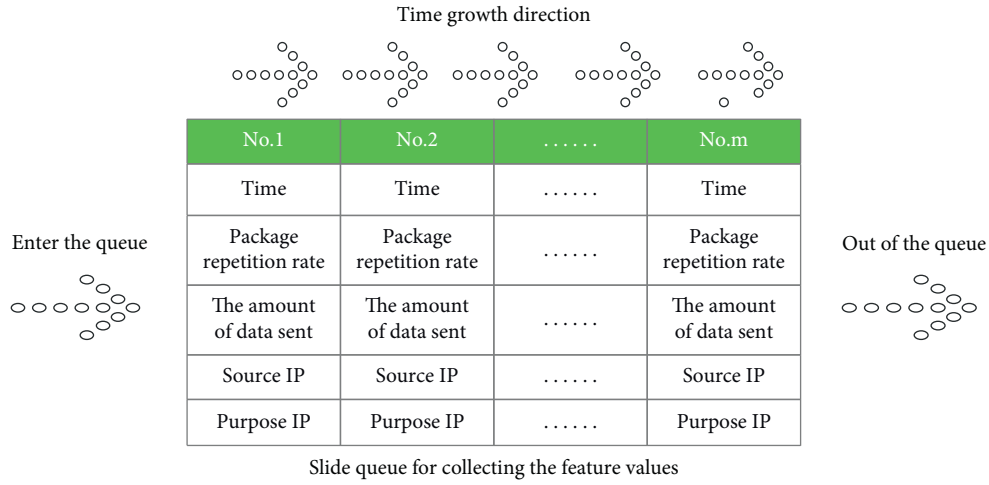


Figure 3: Sliding queue.

*Step l.* The algorithm reads the historical trust sequence $H_{\text{all}}$ locally stored in the cluster head. We assume that Qb is the trust history record of the target node B. That is, $H_b = \{q_1, q_2, q_3, \ldots, q_{dn}\}$, and $n$ is the number of historical interactions. Among them, any element qi ($i > n$) contains information such as flag, time, comprehensive trust, behavior evaluation value, and the address of the interactive node.

*Step 2.* The algorithm checks whether the flag of $q$ is 0. If it is 0, it means that the trust management center has not processed the abnormal node and directly discards the node. If it is not 0, the algorithm continues.

*Step 3.* The algorithm performs attenuation synthesis for each record. The degree of attenuation is represented by the time function $\theta(t) = Q_t + e^{-N_t(t-t_i)}$. Among them, $Q_t$ and $N_t$ are both parameters greater than 0, and they are mainly determined according to the requirements of specific applications. $t$ is the current time, and $t_i$ is the time when $h_i$ occurred.

In summary, the historical comprehensive trust of node A to node B is

$$T^{\text{his}}(A \longrightarrow B) = \sum_{i=1}^{n} \frac{\theta(t_i)}{\sum_{j=1}^{n} \theta(t_j)} [\lambda \text{hist}_i + (1-\lambda) R_i], \quad n > 0,\ 0 < \lambda < 1. \tag{13}$$

In the formula, $\text{hist}_i$ represents the historical interaction satisfaction stored at the $i$-th time, and $R_i$ represents the behavior evaluation value of the i-th interaction.

When there is locally no interactive record, this means that, for unfamiliar nodes, the trusting subject cannot determine whether their behavior is normal or abnormal. To a certain extent, this is similar to the habits of human society.

According to the clusters of the interactive nodes of both parties, it can be divided into intracluster recommendation and intercluster recommendation:

(1) The recommended trust in the cluster is as follows: the subject node and the target node belong to the same cluster, but the subject node A recorded in the cluster head did not interact with the target node B many times or even had no interaction. At this time, the cluster head node selects nodes with a higher reputation value to form the sequence $H_{\text{rec}} = \{h_1, h_2, h_3, \ldots, h_{rn}\}$, and $H_{\text{rec}}$ meets the condition and has multiple interaction records with the target node B. The algorithm takes the trust record in the H sequence as the recommended trust data as $T^{\text{rec}}(A \longrightarrow B)$.

(2) The recommended trust between clusters is as follows: the subject node and the target node are in different clusters. Nodes in different clusters need to pass through the cluster head node to interact. Therefore, compared with the recommendation trust relationship within the cluster, the recommendation trust relationship between clusters is transformed into an indirect relationship, that is, the mutual evaluation between the cluster heads of the two clusters.

The key to recommendation trust calculation is to obtain the trust degree of the main node for the recommendation data. The calculation steps of recommended trust are as follows:

*Step 4.* The cluster head node selects the recommended node from the cluster, and the recommended node C satisfies the following conditions: it has interacted with the main node A, has a high degree of trust, and also has an interaction record with the target node B.

*Step 5.* The algorithm first judges whether the target node B and the subject node A are in the same cluster. If it gets a positive response, the algorithm skips this step and continues to step 3. If it is a different cluster node, the algorithm adds the trust value between clusters as the data for the recommended trust calculation. The trust transfer method

between clusters takes the cluster head node of the cluster where the main node is located as the new main node and the cluster head of the cluster where the target node is located as the recommended node. Then, we substitute the corresponding data into the recommendation trust calculation formula, namely, formula (2) in step 3.

*Step 6.* The algorithm selects the recommended node set $H_{\text{rec}} = \{h_1, h_2, h_3, \ldots, h_{rn}\}$ according to the record, and the recommended trust degree sequence $H_{rt} = \{m_1, m_2, m_3, \ldots, m_{rn}\}$ for the target node B. rn is the number of recommended nodes, $m_i$ is the recommendation trust degree of the corresponding recommended node to the target node, and $0 < i < rn$. Taking the recommended node C in the cluster as an example, the calculation formula of $m_i$ corresponding to the recommended node $C$ is as follows:

$$m_i = T_i^{\text{his}}(A \longrightarrow C) \times T_i^{\text{his}}(C \longrightarrow B), \quad 0 < i < rn. \tag{14}$$

*Step 7.* The algorithm weights and aggregates the recommended trust value provided by each recommended node to obtain the recommended trust value. Considering that malicious nodes deliberately uplift or downgrade the target node to form cooperative deception, the model in this study uses the dispersion of the expected and actual value of the recommended trust degree as the weight to reduce the influence of outliers on the trust evaluation. The reason is that it is almost impossible that most of the recommended nodes are malicious nodes. The formula is as follows:

$$T^{\text{his}}(A \longrightarrow B) = \sum_{i=1}^{rn} \omega_1 \times m_i, \quad 1 < i < rn. \tag{15}$$

Among them, $\omega_i$ is the dispersion degree between the recommendation trust degrees provided by the i-th recommendation node and the overall recommendation trust degree expectation, and it is used as the weight coefficient in formula (3). It can reduce the weight of outlier data in recommendation trust to a certain extent, thereby reducing the impact of malicious recommendation.

$$\omega_i = \frac{1 - \left|E_r(m) - m_i\right|}{\sum_{i=1}^{rn}\left(1 - \left|E_r(m) - m_i\right|\right)}, \quad 0 < i \le rn, \tag{16}$$

where $E_r(m_i)$ is the mathematical expectation of overall recommendation trust.

$$E_r(m) = \frac{m_1 + m_2 + \cdots + m_{rn}}{rn}. \tag{17}$$

The historical statistical trust value $T^{\text{his}}$ and the recommended trust value $T^{\text{rec}}$ obtained by the above calculation and their weighted combination can get the authorized trust value $T^{\text{out}}$. By comparing it with the preset threshold, we arrive at a decision and grant corresponding permissions based on specific network security policies. For nodes with hierarchical authorization, higher-level nodes can be allowed to access and use more resources, and lower-level nodes cannot be accessed beyond authority.

$$T^{\text{out}} = \alpha T^{\text{his}} + (1 - \alpha) T^{\text{rec}}. \tag{18}$$

Among them, $\partial$ represents the historical statistical trust weight, which is obtained by the following formula:

$$\xi = \frac{\left(1/D_h(\text{hist})\right)}{\left(1/D_h(\text{hist})\right) + \left(1/D_r(m)\right)} = \frac{D_r(m)}{D_h(\text{hist}) + D_r(m)}. \tag{19}$$

$D(x)$ is the variance function, which is used to represent the dispersion of the data. For the weight $\alpha$, the general trust model is often determined by means of expert experience and simulation experiment results. Moreover, the result of doing so often lacks science, flexibility, and adaptability.

## 4. Computer Vision-Based Traffic Detection Method of Secure Transmission Industrial Internet of Things

The Internet of Things system comprehensively uses diversified data sensors, radio frequency identification technology, laser scanning equipment, and other tools to collect information on monitored objects, uses the network as a connection to realize data sharing and analysis, and intelligently manages things. This study uses computer vision technology to construct a traffic detection system for the secure transmission of the industrial Internet of Things. The overall layout is shown in Figure 4.

The data service backend, more formally called "middleware," is located between sensing devices and applications. It is the center of providing services for devices and users and is also the core part of our entire IoT remote monitoring system. Its characteristics are shown in Figure 5.

In order to ensure the reliability of transmission, this system adopts TCP-based socket data transmission technology. By analyzing the performance of the transmission framework, a two-tier socket framework is designed, and JSON is used as the message transmission format. Finally, this system uses RSA encryption and integrity verification to achieve reliability and security during data transmission. A detailed explanation of the socket communication process is shown in Figure 6.

It uses a two-layer frame design, as shown in Figure 7. The first layer is N receiving threads to receive client data. The second layer corresponds to N worker threads (processing threads), which process the data received in the receiving threads.

This study adopts a sliding window serial method of abnormal traffic detection with mixed dimensions of time and space. That is, this study first uses the sliding window method based on the time dimension to extract the features of the dataset. After forming several window instances and marking them, this study uses machine learning algorithms for preliminary screening and classification. The purpose of preliminary screening is mainly to locate and find windows with malicious traffic and screen out benign windows. In the second step, this study uses a sliding window method based on spatial dimensions to perform a second accurate detection of the traffic of the malicious window classified for the first time. Moreover, this study re-extracts the features and

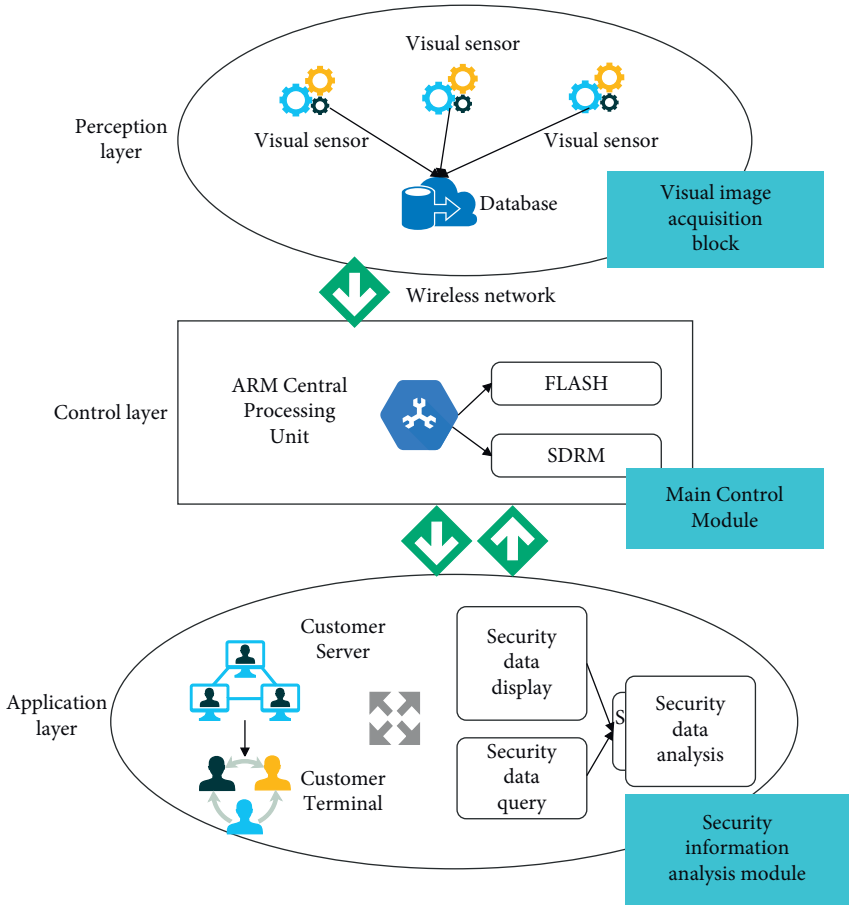


Figure 4: The overall layout of the traffic detection system of secure transmission industrial Internet of Things.

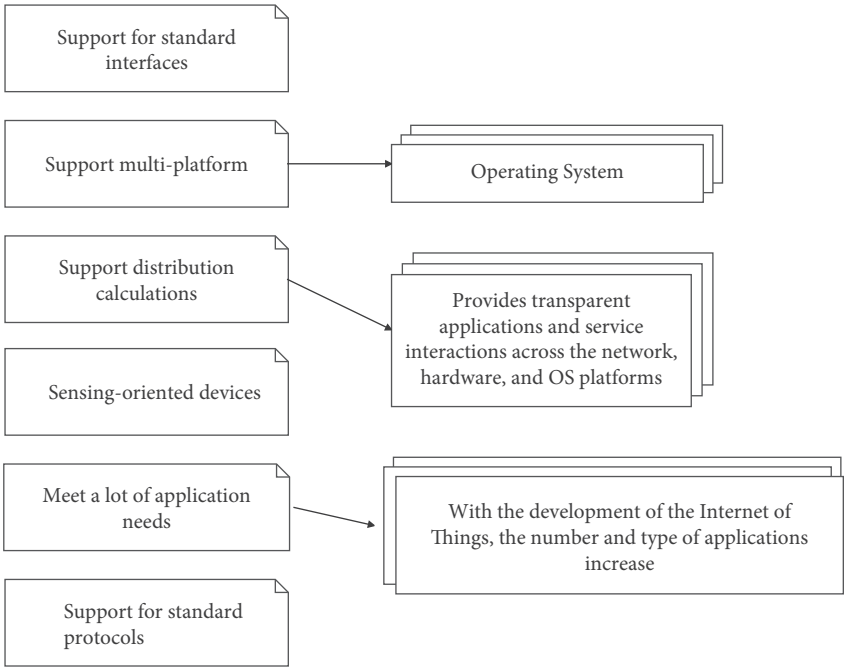


Figure 5: Features of middleware.

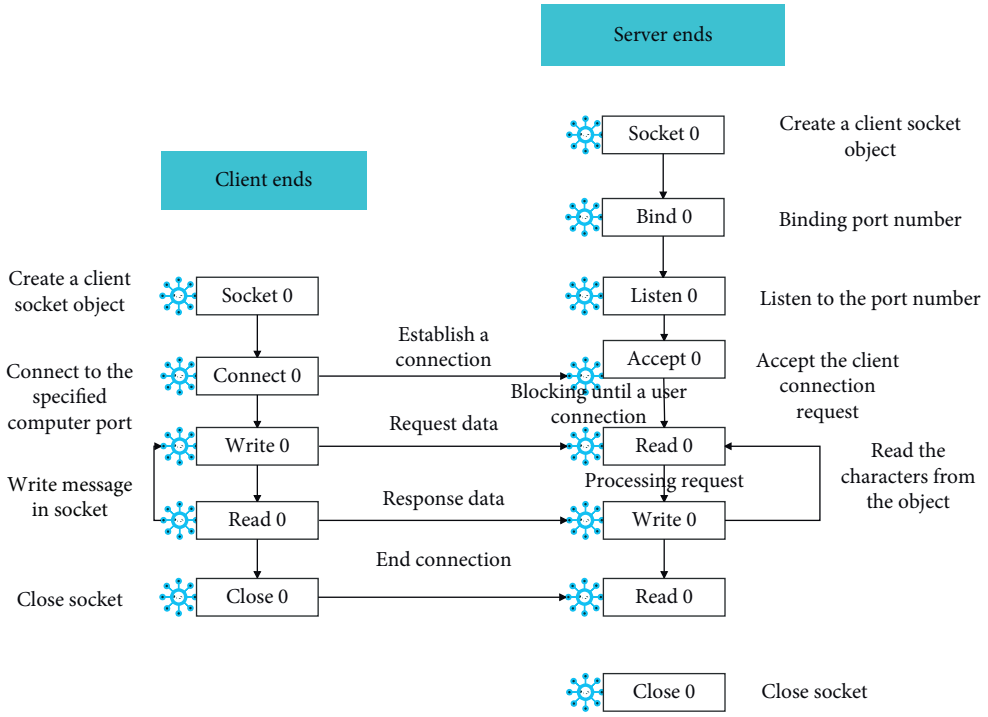


Figure 6: Detailed explanation of the socket communication process.

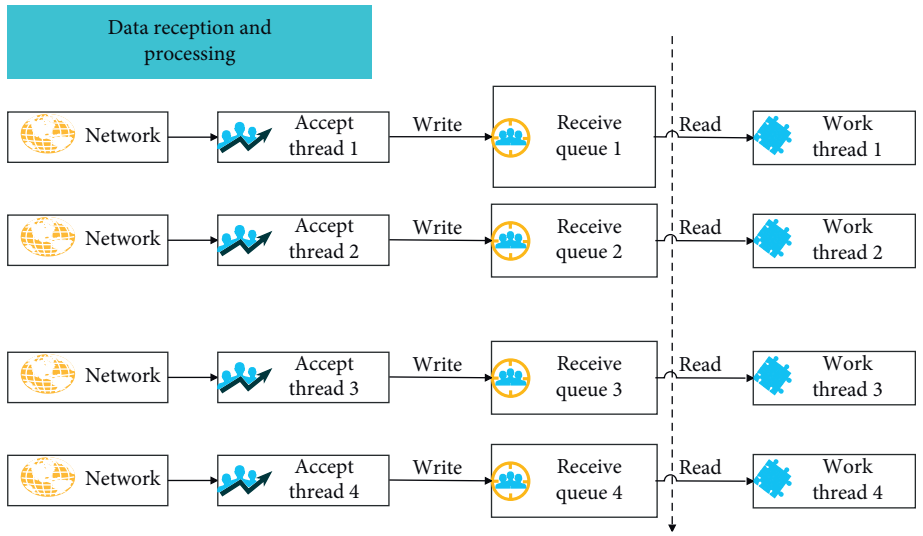


Figure 7: Socket frame design.

marks the flow of this type of window and uses the neural network detection algorithm for fine screening. Figure 8 shows the specific architecture.

Figure 9 shows the overall architecture of the program. The detection scheme is mainly divided into a flow acquisition module, a sliding window abnormal flow detection module based on the time dimension, and a sliding window abnormal flow detection module based on the space dimension.

After constructing the above model, the performance of the model is verified. Under the condition of ensuring that it is not interfered with by other external factors, this study constructs a simulation system of the industrial Internet of Things, performs security detection and flow detection on the system, and uses multiple sets of data to detect and calculate the test results, as shown in Table 1.

It can be seen from the above research that the effects of security detection and traffic detection are very good, so the traffic detection method of computer vision-based secure transmission industrial Internet of Things proposed in this study is very effective.

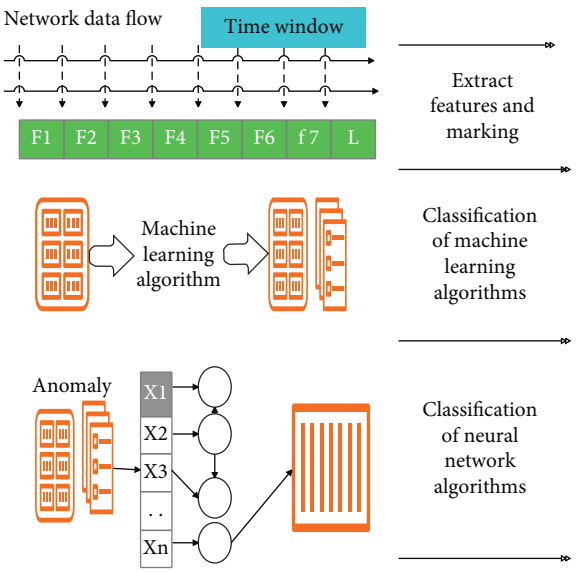


Figure 8: Abnormal traffic detection architecture.

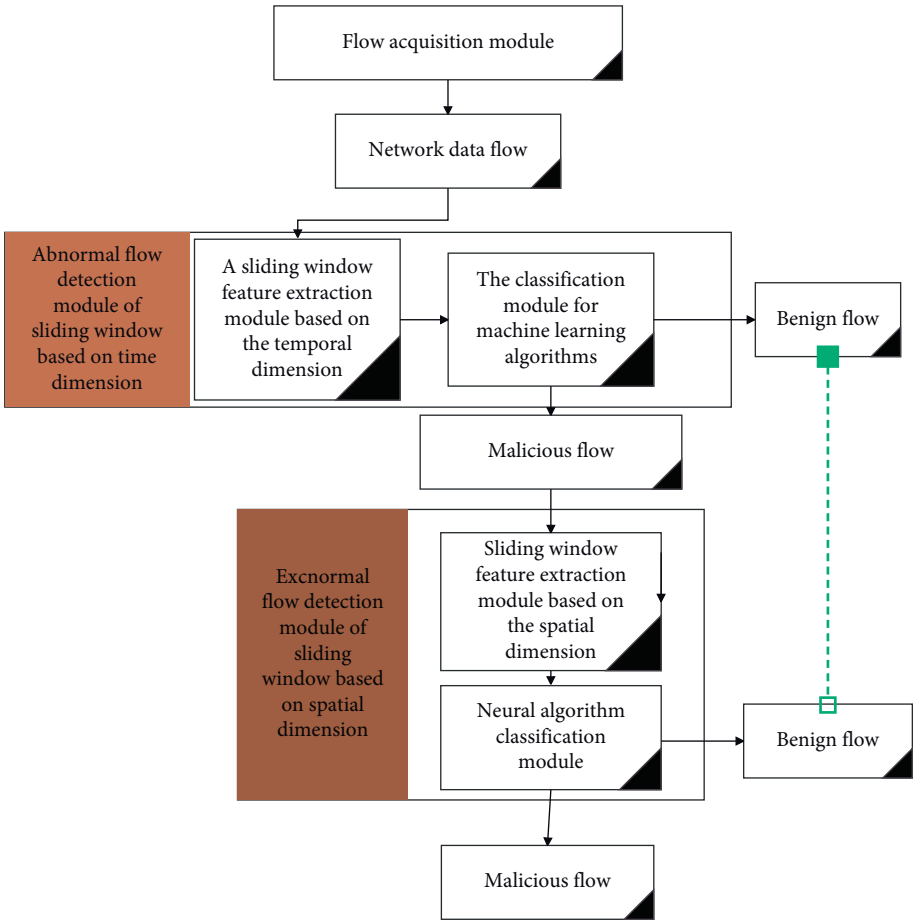


Figure 9: Overall structure diagram.

Table 1: Security detection and flow detection.

| Number | System security | Flow detection | Number | System security | Flow detection | Number | System security | Flow detection |
|---|---|---|---|---|---|---|---|---|
| 1 | 81.9 | 88.0 | 19 | 82.1 | 96.2 | 37 | 84.7 | 92.7 |
| 2 | 79.4 | 89.0 | 20 | 84.9 | 90.3 | 38 | 84.8 | 89.7 |
| 3 | 86.7 | 94.3 | 21 | 89.1 | 85.0 | 39 | 89.3 | 88.6 |
| 4 | 91.2 | 95.4 | 22 | 87.6 | 96.2 | 40 | 91.5 | 92.9 |
| 5 | 92.2 | 85.8 | 23 | 86.8 | 88.2 | 41 | 86.5 | 95.8 |
| 6 | 83.3 | 90.1 | 24 | 85.4 | 87.6 | 42 | 89.5 | 92.4 |
| 7 | 80.1 | 90.1 | 25 | 80.2 | 86.6 | 43 | 87.8 | 87.4 |
| 8 | 79.3 | 86.8 | 26 | 79.2 | 95.1 | 44 | 92.0 | 90.7 |
| 9 | 93.3 | 94.8 | 27 | 88.1 | 90.9 | 45 | 86.1 | 94.2 |
| 10 | 83.9 | 85.5 | 28 | 82.8 | 85.4 | 46 | 82.6 | 91.1 |
| 11 | 86.8 | 87.8 | 29 | 90.6 | 86.1 | 47 | 84.0 | 94.7 |
| 12 | 91.3 | 96.7 | 30 | 85.5 | 92.0 | 48 | 79.3 | 86.9 |
| 13 | 86.3 | 91.7 | 31 | 84.6 | 85.1 | 49 | 84.4 | 95.0 |
| 14 | 86.2 | 91.4 | 32 | 92.8 | 87.5 | 50 | 84.6 | 89.1 |
| 15 | 84.0 | 88.5 | 33 | 89.8 | 92.6 | 51 | 93.1 | 86.4 |
| 16 | 84.4 | 86.2 | 34 | 88.0 | 96.8 | 52 | 82.7 | 88.2 |
| 17 | 81.2 | 89.1 | 35 | 79.7 | 90.9 | 53 | 89.3 | 85.6 |
| 18 | 89.8 | 91.8 | 36 | 90.2 | 90.4 | 54 | 83.4 | 90.5 |

## 5. Conclusions

Currently, the application of big data processing and analysis technology is mostly focused on the Internet field, but there is a lack of rational use of massive industrial data in industrial production scenarios. Therefore, it is necessary to use the existing big data technology to design a better data processing platform suitable for industrial IoT big data. In order to improve the analysis efficiency of industrial Internet of Things data, it is necessary to implement a low-latency query system based on a Hadoop data warehouse, which can handle a large number of concurrent query requests and a single query can return results faster, thereby improving the efficiency of data processing and analysis. This study combines computer vision to conduct research on safe transmission and industrial Internet of Things traffic detection methods to further improve industrial production safety. Moreover, this study constructs an intelligent model to verify the performance of the system combined with experimental research. From the data statistics point of view, the model's security detection and flow detection effects are very good, so the computer vision-based security transmission industrial Internet of Things flow detection method proposed in this study is very effective.

## Data Availability

The labeled dataset used to support the findings of this study is available upon request to the author.

## Conflicts of Interest

The author declares no conflicts of interest.

## Acknowledgments

This work was sponsored by the Kunlun Tourism College, Heilongjiang Institute of Technology.

## References

[1] J. H. Abawajy and M. M. Hassan, "Federated internet of things and cloud computing pervasive patient health monitoring system," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 48–53, 2017.

[2] D. He, R. Ye, S. Chan, M. Guizani, and Y. Xu, "Privacy in the internet of things for smart healthcare," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 38–44, 2018.

[3] P. P. Ray, "Internet of things for smart agriculture: technologies, practices and future direction," *Journal of Ambient Intelligence and Smart Environments*, vol. 9, no. 4, pp. 395–420, 2017.

[4] Y. A. Qadri, A. Nauman, and Y. B. Zikria, "The future of healthcare internet of things: a survey of emerging technologies," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1121–1167, 2020.

[5] B. H. Dobkin, "A rehabilitation-internet-of-things in the home to augment motor skills and exercise training," *Neurorehabilitation and Neural Repair*, vol. 31, no. 3, pp. 217–227, 2017.

[6] J. Yao and N. Ansari, "Caching in energy harvesting aided Internet of Things: a game-theoretic approach," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3194–3201, 2018.

[7] J. E. Siegel, S. Kumar, and S. E. Sarma, "The future internet of things: secure, efficient, and model-based," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2386–2398, 2017.

[8] M. A. Abd-Elmagid, N. Pappas, and H. S. Dhillon, "On the role of age of information in the Internet of Things," *IEEE Communications Magazine*, vol. 57, no. 12, pp. 72–77, 2019.

[9] A. Sheth, U. Jaimini, and H. Y. Yip, "How will the internet of things enable augmented personalized health?[J]," *IEEE Intelligent Systems*, vol. 33, no. 1, pp. 89–97, 2018.

[10] G. J. Joyia, R. M. Liaqat, A. Farooq, and Rehman, "Internet of Medical Things (IOMT): applications, benefits and future challenges in healthcare domain," *Journal of Communication*, vol. 12, no. 4, pp. 240–247, 2017.

[11] N. Kshetri, "The evolution of the internet of things industry and market in China: an interplay of institutions, demands and supply," *Telecommunications Policy*, vol. 41, no. 1, pp. 49–67, 2017.

[12] S. Siboni, V. Sachidananda, and Y. Meidan, "Security testbed for Internet-of-Things devices," *IEEE Transactions on Reliability*, vol. 68, no. 1, pp. 23–44, 2019.

[13] Y. Yang, M. Zhong, H. Yao, and Yu, "Internet of things for smart ports: technologies and challenges," *IEEE Instrumentation and Measurement Magazine*, vol. 21, no. 1, pp. 34–43, 2018.

[14] M. Mayer and A. J. Baeumner, "A megatrend challenging analytical chemistry: biosensor and chemosensor concepts ready for the internet of things," *Chemical Reviews*, vol. 119, no. 13, pp. 7996–8027, 2019.

[15] M. Saez, F. P. Maturana, and K. Barton, "Real-time manufacturing machine and system performance monitoring using internet of things," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 4, pp. 1735–1748, 2018.

[16] V. Jagadeeswari, V. Subramaniyaswamy, R. Logesh, and V. Vijayakumar, "A study on medical Internet of Things and Big Data in personalized healthcare system," *Health Information Science and Systems*, vol. 6, no. 1, pp. 1–20, 2018.

[17] S. Smys, A. Basar, and H. Wang, "Hybrid intrusion detection system for internet of things (IOT)," *Journal of ISMAC*, vol. 2, no. 4, pp. 190–199, 2020.

[18] I. Butun, P. Österberg, and H. Song, "Security of the internet of things: vulnerabilities, attacks, and countermeasures," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 616–644, 2019.

[19] T. Qiu, N. Chen, K. Li, and M. Zhao, "How can heterogeneous internet of things build our future: a survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2011–2027, 2018.

[20] A. Heiskanen, "The technology of trust: how the Internet of Things and blockchain could usher in a new era of construction productivity," *Construction Research and Innovation*, vol. 8, no. 2, pp. 66–70, 2017.

[21] M. Wolf and D. Serpanos, "Safety and security in cyber-physical systems and internet-of-things systems," *Proceedings of the IEEE*, vol. 106, no. 1, pp. 9–20, 2017.