JOURNAL OF
COMPUTER SCIENCE AND
ELECTRICAL ENGINEERING

Journal of Computer Science and Electrical Engineering

Table of Content

OLYMPIC MEDAL PREDICTION AND ANALYSIS BASED ON LSTM AND TOPSIS MODELS

DingShu Yan
College of life sciences, Shanxi Agricultural University, Jinzhong 030801, Shanxi, China.
Corresponding Email: yds2004517@163.com

Abstract: In the context of the increasingly fierce global sports competition, accurately predicting Olympic medal outcomes and optimizing the allocation of sports resources have become crucial concerns for national sports committees and related organizations. This study tackles these challenges through the use of advanced modeling techniques. A Long Short-Term Memory (LSTM) model was constructed using historical data from the Summer Olympics (1896-2024), encompassing medal counts, participating events, and national indicators such as population and GDP. The model takes into account the time-dependence of historical performance, advantages in sports infrastructure, and the benefits of being the host country. The results predict that the United States, China, and France will demonstrate strong medal competitiveness at the 2028 Los Angeles Olympics, with potential breakthroughs from emerging nations. Moreover, a decision tree model was employed to examine the influence of "great coaches" on medal results. By examining coach mobility, athlete performance data, and changes in medal counts, the study revealed that transnational coach mobility significantly influences medal distribution. Notable coaches like Lang Ping and Bela Karolyi have enhanced the competitiveness of volleyball and gymnastics, respectively. The findings suggest that recruiting top-tier coaches can increase medal counts and elevate international sports performance. This research provides valuable strategies for optimizing sports resource allocation and enhancing global competitiveness.
Keywords: Olympic medal table; LSTM prediction model; Decision tree model; Great coach effect; Sport resource allocation

1 INTRODUCTION

The 2024 Paris Olympics have highlighted the significance of medal standings, which largely reflect the sporting strength of each country [1]. The United States emerged as the top medal winner, with both China and the U.S. sharing the first place in the total number of gold medals [2]. Some countries achieved their best-ever performances and won their first-ever medals [3]. However, over 60 countries did not win any medals, indicating an extremely unbalanced distribution of medals [4]. Traditionally, predictions were based on the form and strength of individual athletes [5]. Yet, historical data also holds great value and should not be underestimated [6]. Factors such as adjustments in event settings and the coaching level of the coaching team also affect medal distribution. This study mines key information from historical data, constructs a medal distribution prediction model based on it, and considers various factors that interfere with medal distribution to make a more comprehensive and objective prediction of each country's medals [7]. It is hoped that the model can provide targeted suggestions for national Olympic committees, enabling them to plan their sports investment more scientifically and allocate resources rationally, thereby effectively improving the competitive level of their athletes [8].

Previous research has explored the prediction of Olympic medal outcomes and the optimization of sports resource allocation [9]. For example, some research has applied conventional statistical methods like linear regression and time series analysis to predict medal counts using historical data. Others have focused on the impact of specific factors such as athlete training methods, sports facilities, and government funding on sports performance [10]. However, these studies often have limitations. Many of them fail to fully utilize the rich information contained in historical data, such as the time-dependence of historical performance and the advantages of sports infrastructure. Additionally, the impact of transnational coach mobility on medal distribution has not been adequately considered.

This paper overcomes these limitations by utilizing advanced modeling approaches, including long short-term memory (LSTM) models and decision tree models. The LSTM model takes into account the time-dependence of historical performance, sports infrastructure advantages, and the benefits of being the host country, providing more accurate predictions of medal outcomes [11]. The decision tree model analyzes the impact of "great coaches" on medal outcomes, revealing the significant influence of transnational coach mobility on medal distribution [12]. Our findings suggest that recruiting top-tier coaches can increase medal counts and elevate international sports performance, offering valuable strategies for optimizing sports resource allocation and enhancing global competitiveness.

The data in this article comes from https://olympics.com/en/paris-2024/medals.

2 MODEL

2.1 LSTM

The Long Short-Term Memory (LSTM) network is a specialized variant of Recurrent Neural Networks (RNNs) aimed at tackling the challenge of long-term dependencies in sequential data. Standard RNNs often fail to capture long-term dependencies due to problems such as vanishing and exploding gradients. To overcome these limitations, LSTM incorporates a gating mechanism that regulates the flow of information via memory cells, input gates, forget gates, and output gates. This structure enables LSTMs to retain information over extended periods and accurately model intricate temporal relationships.

The LSTM model functions via a set of gates and cell states that manage the flow of information. Its core components consist of:

Forgetting Gate: The forgetting gate decides which information to discard from the cell state. It employs a sigmoid activation function to generate values ranging from 0 to 1. A value of 0 means the information should be entirely forgotten, while a value of 1 signifies that it should be completely retained.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

where f_t is the forgetting gate output, W_f is the weight matrix, h_{t-1} is the previous hidden state, x_t is the current input, and b_f is the bias vector.

Input Gate: The input gate decides what new information to store in the cell state. It is composed of two components: a sigmoid layer and a tanh layer. The sigmoid layer regulates the update process, while the tanh layer generates a new candidate value to be incorporated into the cell state.

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \tag{2}$$

$$\widetilde{C}_t = \tanh\left(W_C [h_{t-1}, x_t] + b_C\right) \tag{3}$$

where i_t is the input gate output, $\widetilde{C}_t$ is the candidate cell state, W_i and Wc are weight matrices, and b_i and b_c are bias vectors.

Cell State Update: The cell state is refreshed by integrating the previous cell state with the outputs from the forgetting gate, the input gate, and the candidate cell state.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t \tag{4}$$

where C_t is the current cell state.

Output Gate: The output gate determines what information should be output based on the current cell state. It uses a sigmoid gate to control the output and a tanh function to scale the output.

$$o_t = \sigma\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \tag{5}$$

$$h_t = o_t * \tanh\left(C_t\right) \tag{6}$$

where o_t is the output gate output, h_t is the current hidden state, W_o is the weight matrix, and b_o is the bias vector.

2.2 Decision Tree Model

A decision tree is a non-parametric, supervised learning algorithm applicable for both classification and regression. It operates by recursively splitting the dataset into subsets according to the values of input features, forming a tree-like structure of decisions. In this structure, each internal node corresponds to a test on a feature, each branch indicates the result of the test, and each leaf node signifies a class label or a predicted value.

The decision tree model functions by iteratively dividing the data into subsets according to the values of input features. The essential steps are as follows:

Feature Selection: The model identifies the most effective feature for partitioning the data, using metrics like Gini impurity for classification tasks or mean squared error (MSE) for regression tasks.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{y}_i\right)^2 \tag{7}$$

where y_i is the true value, $\hat{y}_i$ is the predicted value, and n is the number of samples.

Data Partitioning: The dataset is divided into smaller subsets according to the chosen feature and a specified threshold. This procedure is recursively applied to each subset until a termination criterion is satisfied, such as reaching the maximum tree depth or achieving a minimum number of samples per node.

Tree Construction: The tree is built by repeatedly performing the data-splitting process. Each internal node corresponds to a decision based on a specific feature, while each leaf node indicates a prediction value.

Prediction: For a new input sample, the model navigates from the root node to a leaf node, evaluating the input feature values at each internal node to determine the path. The prediction is the value associated with the final leaf node.

3 RESULTS AND ANALYSIS

3.1 Results and Analysis of the LSTM Model

The LSTM model was developed using historical medal data from the Summer Olympics (1896-2024), incorporating features like the number of medals won, participation in events, national population, and GDP. It was designed to forecast the medal count for each country in the 2028 Los Angeles Olympics. During training, the data was standardized and normalized using the MinMaxScaler to scale the values between 0 and 1. The model was trained over 20 epochs with a batch size of 32.

The training loss curve (Figure 1) illustrates a steady decline in the loss value as the number of training iterations increased, suggesting that the model progressively converged and captured the underlying patterns in the data.
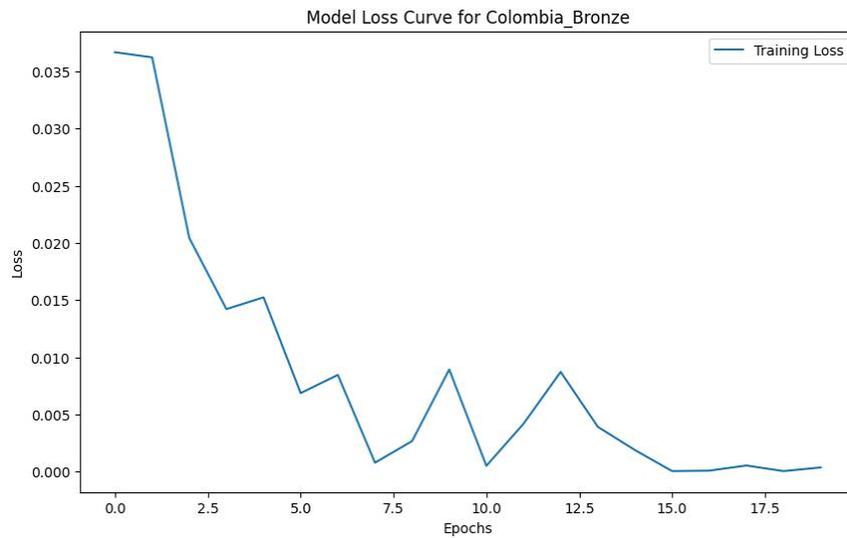
(a)Model Loss Curve for Colombia Bronze
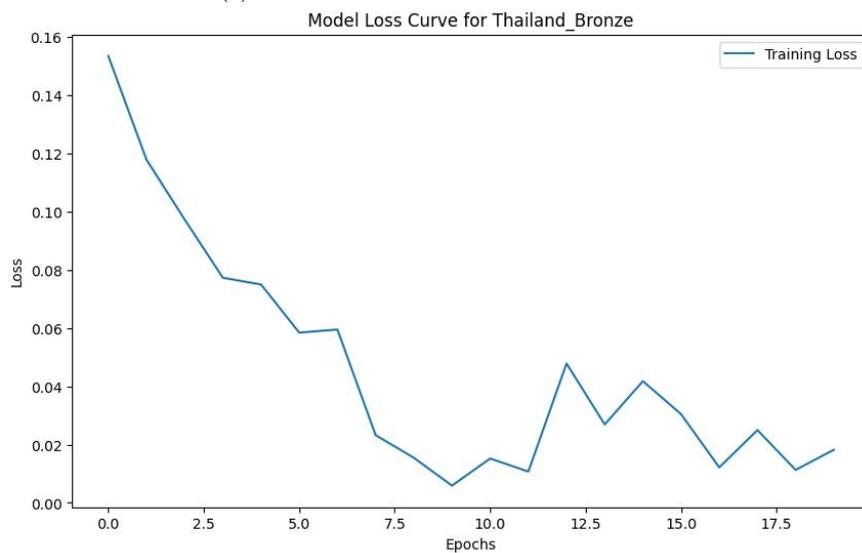
(b)Model Loss Curve for Thailand_Bronze

Model Loss Curve for China_Bronze

(c)Model Loss Curve for China_Bronze
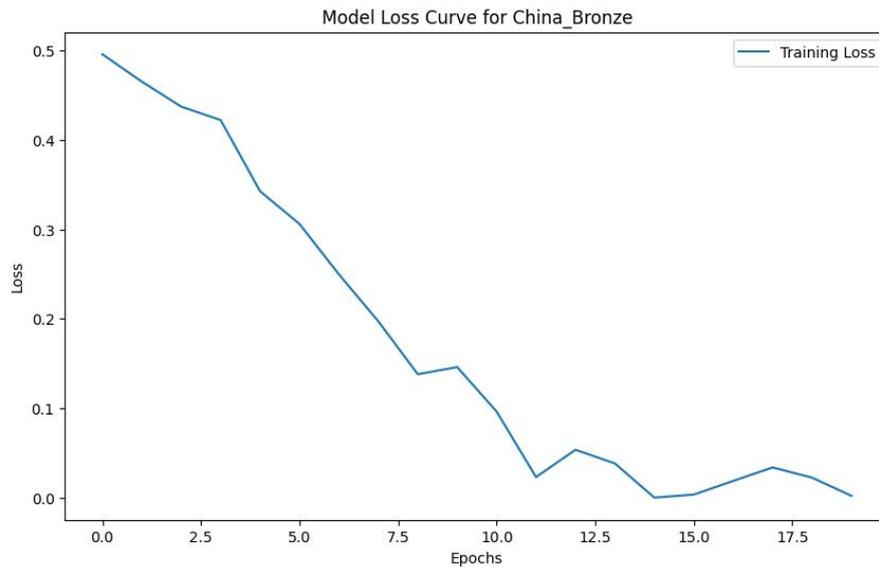
Model Loss Curve for Morocco_Bronze
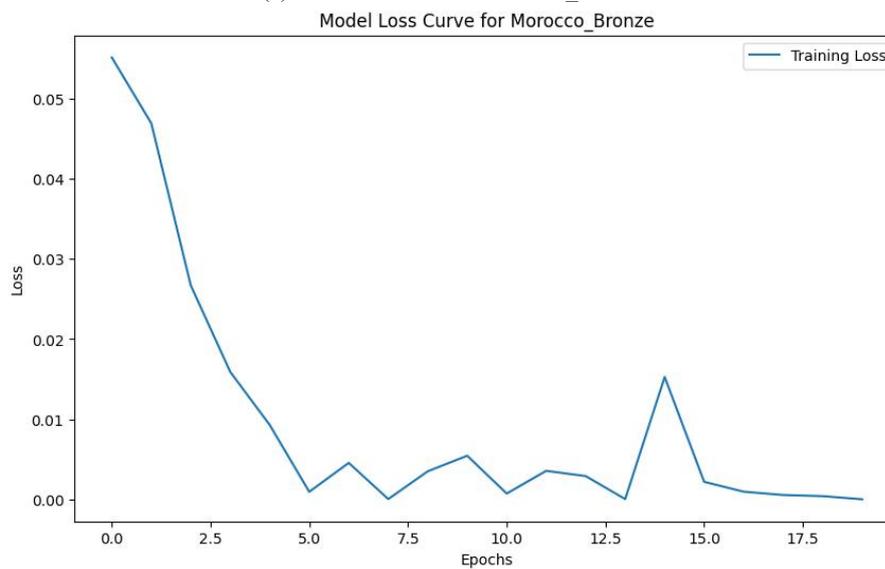
(d)Model Loss Curve for Morocco_Bronze

Figure 1 Training Loss Curve of the LSTM Model

The model's forecasts for the total number of medals, as well as gold, silver, and bronze medals for each country in the 2028 Olympics, were compared with the actual medal counts from the 2024 Olympics. The findings (Figure 2) indicate that the United States and China are anticipated to continue leading the medal standings, with the United States potentially experiencing a modest decline in overall medals. Other nations, including Great Britain, France, and Australia, are projected to maintain relatively stable medal counts.

(a)Total Medals Comparison(2024 vs 2028)

(b)Gold Medals Comparison(2024 vs 2028)

(c)Silver Medals Comparison(2024 vs 2028)

(d)Bronze Medals Comparison(2024 vs 2028)
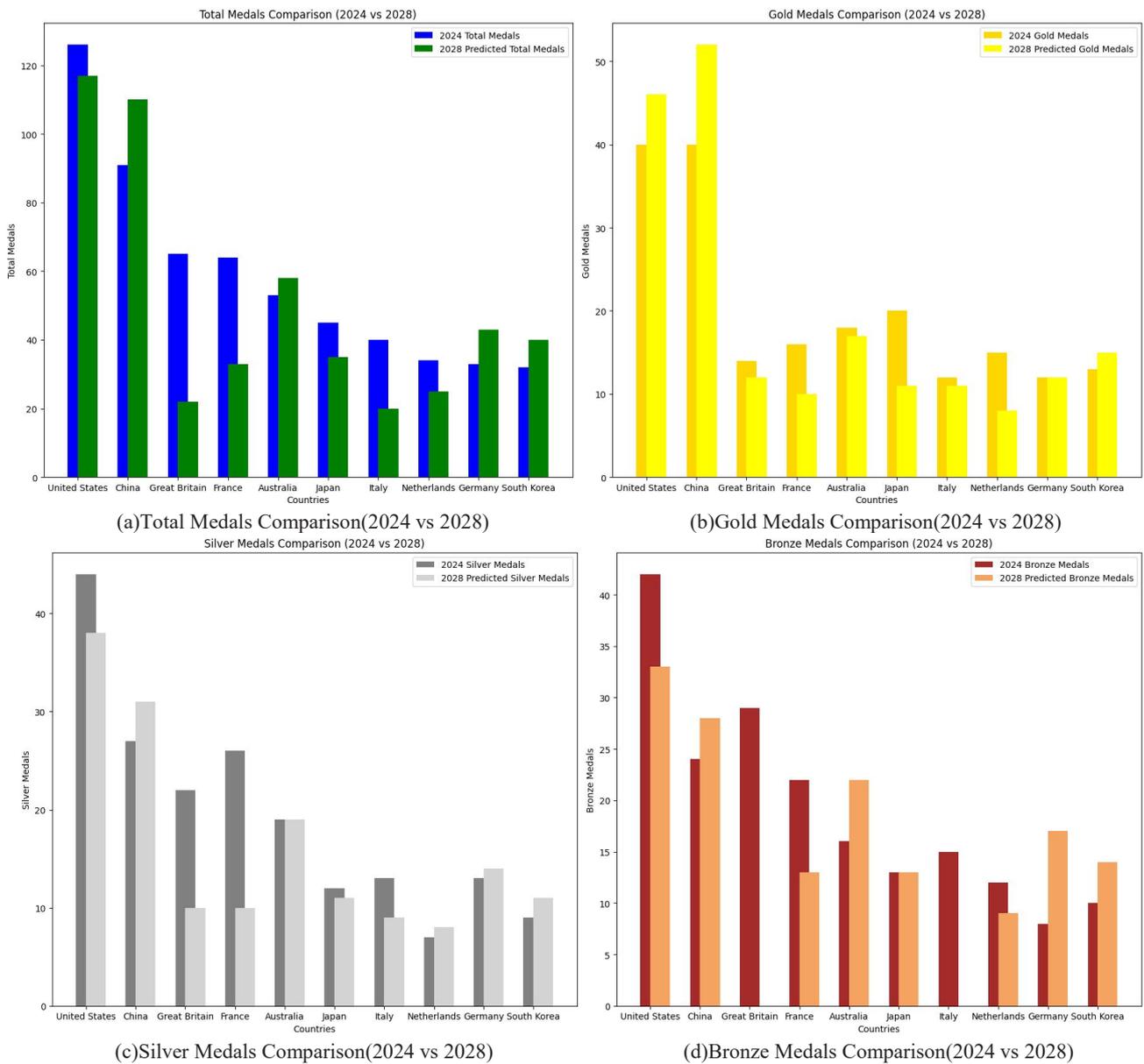
Figure 2 Predicted vs. Actual Medal Counts for Top Countries (2024 vs. 2028)

The model also predicted the medal counts for countries that did not win any medals in the 2024 Olympics. The top ten countries most likely to win medals in the 2028 Olympics are shown in Figure 3, with Finland being the most likely to achieve a breakthrough.
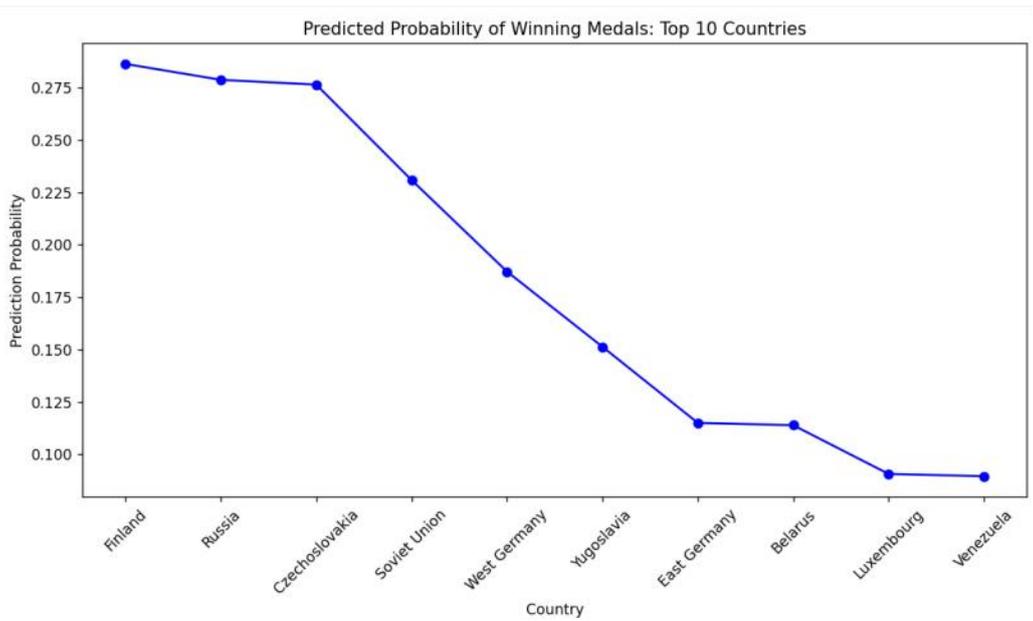
Figure 3 Top Ten Countries Most Likely to Win Medals in 2028

The LSTM model successfully identified the long-term trends and patterns within the historical medal data. The model's predictions suggest that the United States, China, and France will continue to dominate the medal standings in the 2028 Olympics, while emerging countries may also achieve significant breakthroughs. The model's ability to predict medal counts for non-medal-winning countries provides valuable insights for national sports committees to plan their sports investment and resource allocation strategies.

The correlation matrix between the number of events and the number of medals (Figure 4) reveals a strong positive correlation, suggesting that countries participating in more events are likely to win more medals. The heatmap visualization (Figure 5) further highlights the efficiency of medal acquisition in different sports, with darker colors representing higher contributions to the total medal count.
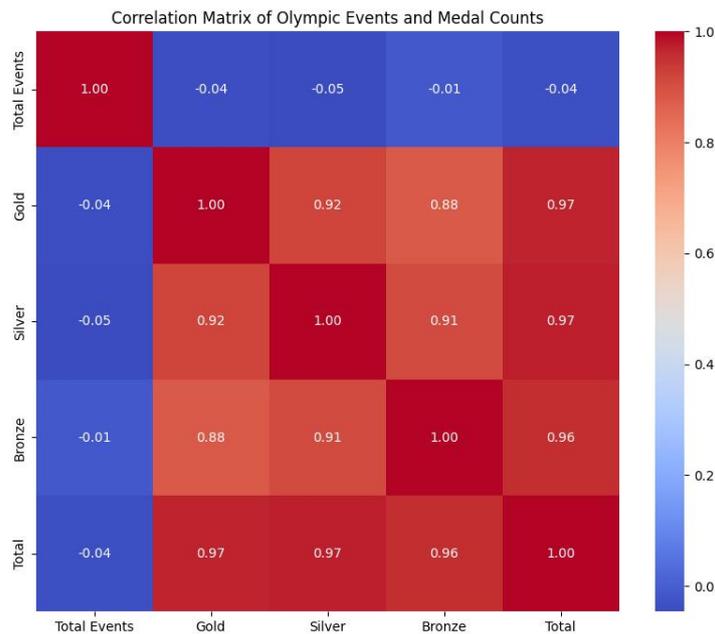
Figure 4 Heat Map of the Correlation Between the Number of Events and Medals
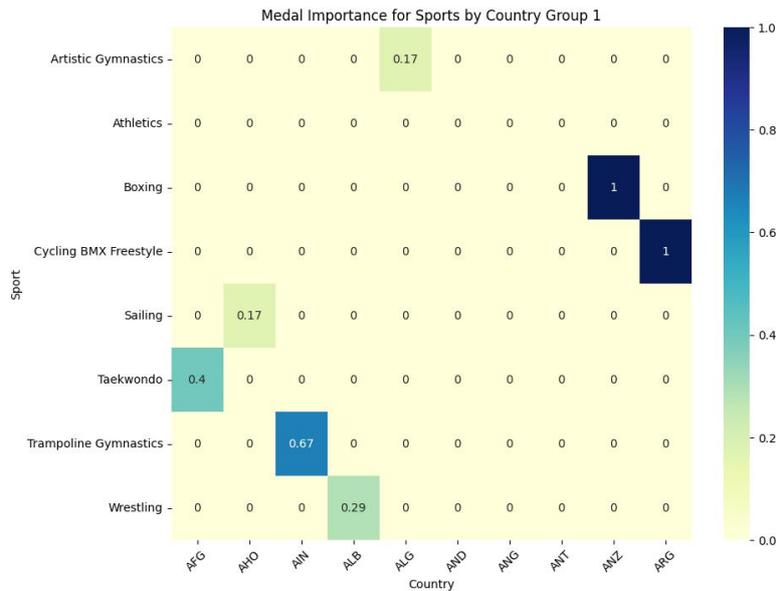
Figure 5 Heatmap of Medal-Acquisition Efficiency in Different Sports

The analysis of the host country advantage (Figures 6, 7, 8) shows that host countries typically win more medals during their hosting year compared to non-hosting years. This phenomenon can be explained by the host country's capacity to expand the number of events and tailor the types of events to suit their strengths.
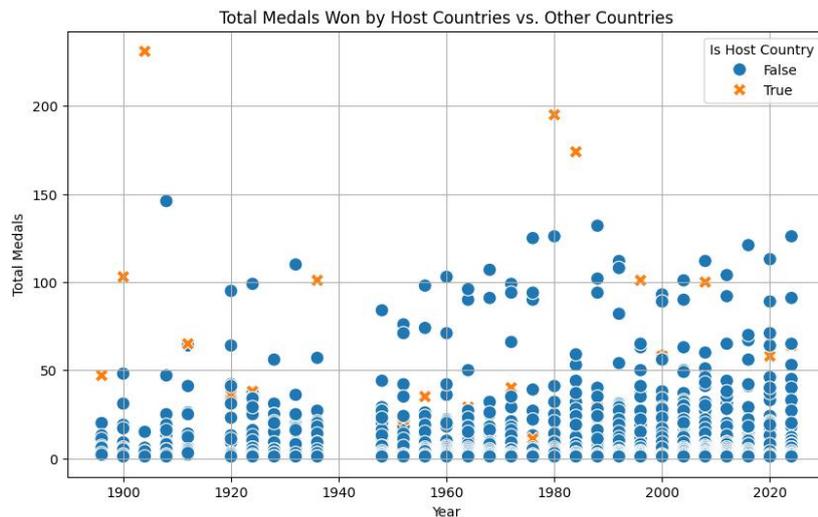
Figure 6 Total Number of Medals Won by Host Countries in Olympic Years
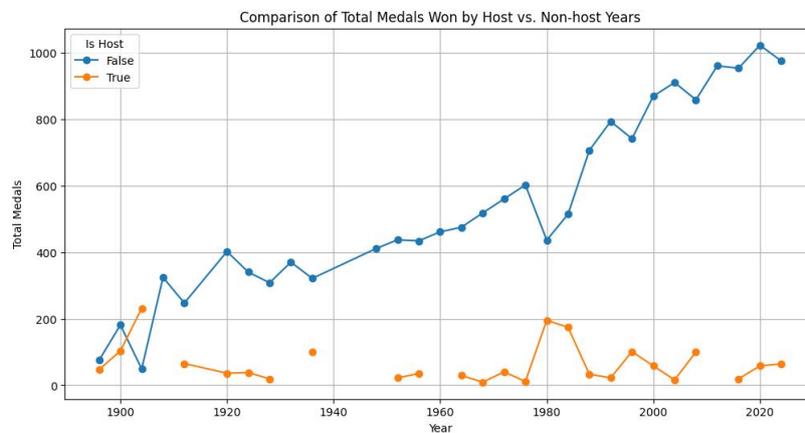
Figure 7 Comparison of Host Country Medal Totals in Olympic vs. Non-Host Years
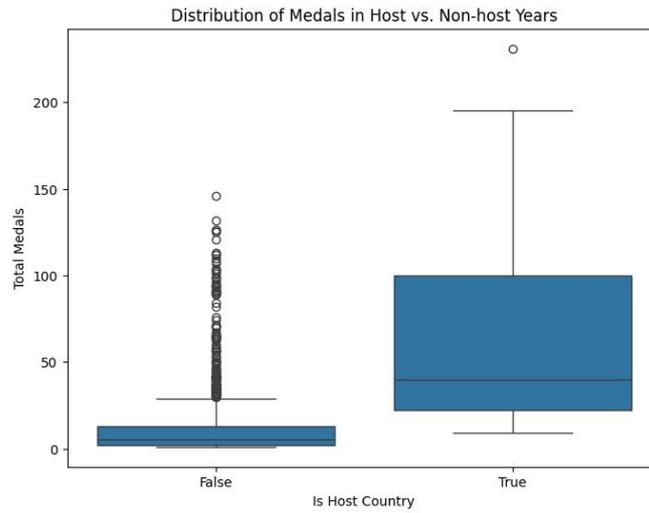
Figure 8 Distribution of Medals of Host Countries in Olympic and Non-Host Years

3.2 Results and Analysis of the Decision Tree Model

The decision tree model was employed to assess the influence of "great coaches" on Olympic medal counts. It was trained using data that included the number of athletes, medal efficiency (medals per athlete), and the medal tally for each country across various sports. The model's structure illustrates how the data is divided based on the number of athletes and medal efficiency to forecast the number of medals.

The model's forecasts (Figures 9, 10, 11) indicate that the introduction of "great coaches" can substantially boost a country's medal count. For example, the model predicts that China could see a substantial increase in medals in sports such as gymnastics and diving if it recruits top-tier coaches.
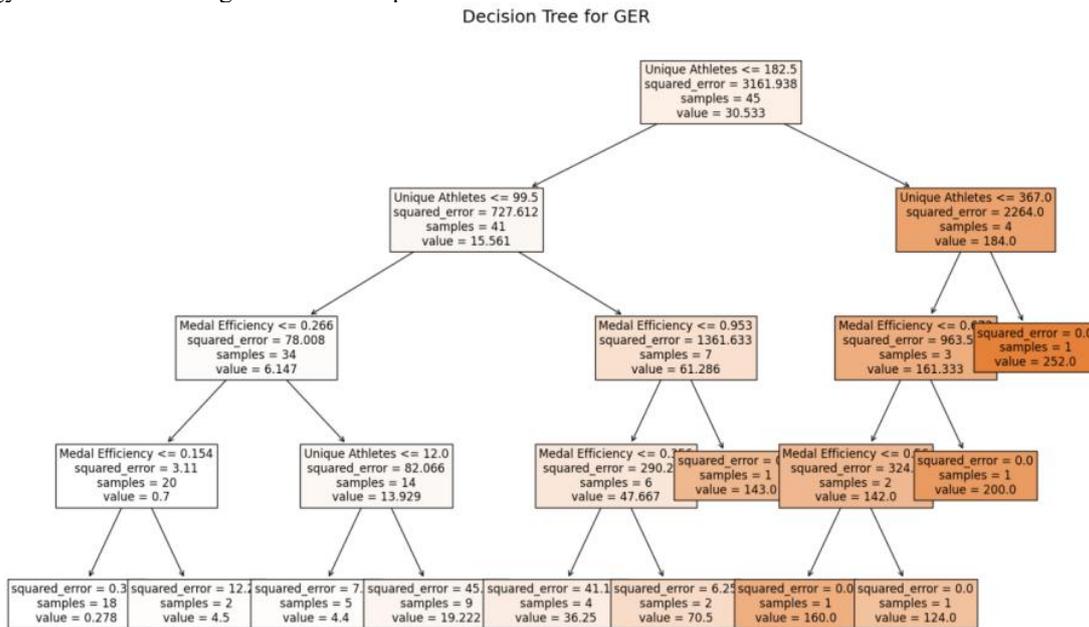
Figure 9 Predicted Medal Counts for Country 1 with and without "Great Coaches"
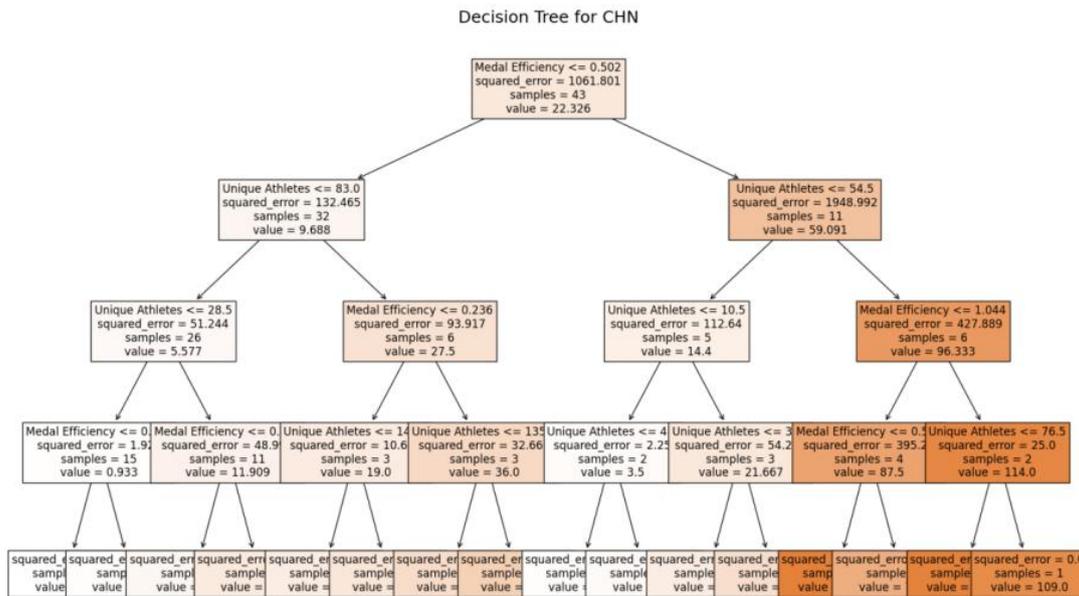
Figure 10 Predicted Medal Counts for Country 2 with and without "Great Coaches"
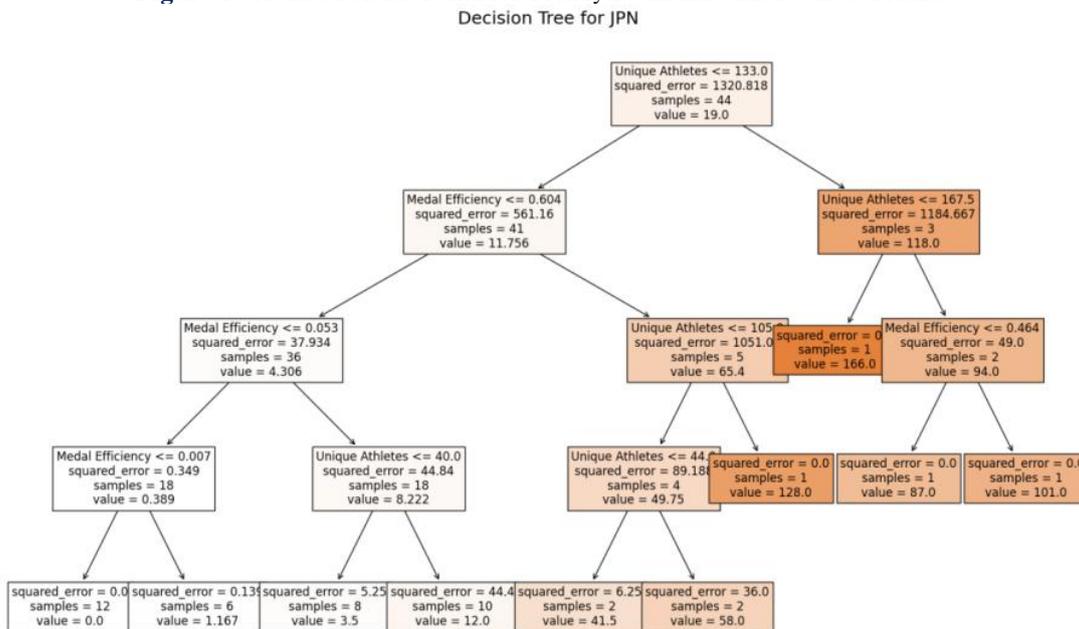
Figure 11 Predicted Medal Counts for Country 3 with and without "Great Coaches"

The decision tree model highlighted that the efficiency of winning medals is a key determinant of a country's overall medal count. The findings suggest that hiring "great coaches" can markedly improve a country's medal performance, especially in sports where the country already possesses a solid base. The model's ability to quantify the impact of "great coaches" provides valuable insights for national sports committees to optimize their coaching resources and strategic planning.

The visualization of the decision tree model shows that the model partitions the data based on the number of athletes and medal efficiency, with the top-level partition being based on medal efficiency. This indicates that medal efficiency is a more important factor than the number of athletes in determining medal counts. The model's forecasts for various countries demonstrate that bringing in "great coaches" can result in a substantial rise in medals, especially in sports where the country already has a high efficiency in winning medals.

To sum up, the decision tree model serves as an effective instrument for examining how "great coaches" influence Olympic medal performance. It also provides useful guidance for optimizing coaching resources and boosting international sports competitiveness.

4   CONCLUSIONS AND OUTLOOKS

The LSTM model was effectively constructed using historical medal data, participating events, and national indicators such as population and GDP, capturing the time-dependence of historical performance and the advantages of sports infrastructure and host country benefits. The model predicted that the United States, China, and France will exhibit strong medal competitiveness at the 2028 Los Angeles Olympics, with potential breakthroughs from emerging nations, providing valuable insights for national sports committees to plan their sports investment and resource allocation strategies. Moreover, the model's ability to predict medal counts for non-medal-winning countries offers a comprehensive understanding of the future distribution of medals, helping to anticipate changes in the ranking of some countries in the medal table. Additionally, the decision tree model quantified the impact of "great coaches" on medal outcomes, revealing that transnational coach mobility significantly influences medal distribution, with notable coaches like Lang Ping and Bela Karolyi enhancing the competitiveness of volleyball and gymnastics, respectively. The model's predictions showed that recruiting top-tier coaches can significantly increase the number of medals won by a country, particularly in sports where the country already has a strong foundation, providing valuable insights for national sports committees to optimize their coaching resources and strategic planning. Furthermore, the study analyzed the correlation between the distribution of Olympic medals and the sports development strategies of each country, indicating that medal distribution is related to sports traditions, advantageous programs, event settings, and factors such as economic level, population size, and sports policies. The analysis of host country advantage showed that host countries typically win more medals during their hosting year compared to non-hosting years, attributed to their ability to increase the number of events and adjust the types of events to their advantage.

Future research may concentrate on improving the predictive accuracy of models by integrating more extensive and current data, such as athlete performance metrics, training methods, and technological innovations in sports. Exploring advanced machine learning techniques, including ensemble methods and deep learning architectures, could further enhance model performance. The results of this study lay the groundwork for optimizing sports resource allocation. Future research could explore specific strategies and policies that national sports committees can implement to maximize their potential for winning medals. The impact of various factors, such as government funding, sports infrastructure development, and athlete welfare programs, can be further analyzed to provide a more holistic approach to sports resource management. The study's insights into the impact of "great coaches" and the host country advantage can inform global sports development initiatives. Future research can explore the role of international collaborations and knowledge sharing in enhancing sports performance across different countries. The study can also serve as a basis for developing policies and programs aimed at promoting sports participation and excellence at the grassroots level, thereby fostering a more inclusive and competitive global sports environment.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Bian X. Predicting Olympic medal counts: The effects of economic development on Olympic performance. The park place economist, 2005, 13(1): 37-44.

[2] Tchamkerten A, Chaudron P, Girard N, et al. Career factors related to winning Olympic medals in swimming. PLoS One, 2024, 19(6): e0304444.

[3] Csurilla G, Fertő I. How to win the first Olympic medal? And the second?. Social Science Quarterly, 2024, 105(5): 1544-1564.

[4] Zheng J, Dickson G, Oh T, et al. Competitive balance and medal distributions at the Summer Olympic Games 1992–2016: Overall and gender-specific analyses. Managing Sport and Leisure, 2019, 24(1-3): 45-58.

[5] Nagpal P, Gupta K, Verma Y, et al. Paris Olympic (2024) Medal Tally Prediction//International Conference on Data Management, Analytics & Innovation. Singapore: Springer Nature Singapore, 2023: 249-267.

[6] Hołub M, Stanula A, Baron J, et al. Predicting breaststroke and butterfly stroke results in swimming based on Olympics history. International journal of environmental research and public health, 2021, 18(12): 6621.

[7] Heazlewood T. Prediction versus reality: The use of mathematical models to predict elite performance in swimming and athletics at the Olympic Games. Journal of Sports Science & Medicine, 2006, 5(4): 480.

[8] De Bosscher V, Shibli S, Weber A C. Is prioritisation of funding in elite sport effective? An analysis of the investment strategies in 16 countries. European Sport Management Quarterly, 2019, 19(2): 221-243.

[9] Deepak V, Anguraj D K, Mantha S S. An efficient recommendation system for athletic performance optimization by enriched grey wolf optimization. Personal and Ubiquitous Computing, 2023, 27(3): 1015-1026.

[10] Thompson F, Rongen F, Cowburn I, et al. The impacts of sports schools on holistic athlete development: a mixed methods systematic review. Sports medicine, 2022, 52(8): 1879-1917.

[11] Wang Y, Wang J, Yang J, et al. STGCN-LSTM for Olympic Medal Prediction: Dynamic Power Modeling and Causal Policy Optimization. arXiv preprint arXiv:2501.17711, 2025.

[12] Jeon M, Lim H. Taekwondo win-loss determining factors using data mining-based decision tree analysis: focusing on game analysis for evidence-based coaching. BMC Sports Science, Medicine and Rehabilitation, 2024, 16(1): 115.

ENHANCED VINS-BASED UNDERWATER LOCALIZATION WITH IMAGE ENHANCEMENT

Fei Liao*, BingLei Bao

Department of Automation, University of Science and Technology of China, Hefei 230026, Anhui, China.
Corresponding Author: Fei Liao, Email: roboleofly@mail.ustc.edu.cn

Abstract: The underwater visual-inertial navigation system (VINS) confronts significant challenges due to the adverse underwater visual environment, including light absorption and scattering, tiny suspended particles, color distortion, and image blurring. To address these issues, this paper introduces a multi-scale fusion-based image enhancement algorithm, integrating it into the front-end of underwater image enhancement techniques. This integration effectively enhances the performance of underwater localization. Experimental results on the Aqualoc underwater dataset demonstrate that the proposed method increases the number of extracted feature points and achieves more stable tracking, thereby reducing localization errors compared to traditional VINS approaches.
Keywords: Underwater image enhancement; Underwater slam

1 INTRODUCTION

Since the beginning of the 21st century, the rapid growth of population has led to a gradual depletion of land resources, and countries have stepped up their exploration and development of the ocean. With the rapid development of computer vision technology, camera sensors have become the main way for autonomous robots and smart wearable devices to perceive the surrounding environment. Rich visual information is extracted and input into the intelligent body to achieve complex tasks such as detection and recognition, positioning and navigation, and planning and decision-making.

Compared with ordinary images, images captured underwater suffer from significantly reduced visibility due to light propagation attenuation [1]. The reduction in underwater visibility is mainly caused by optical phenomena, including absorption and scattering processes. The enhanced scattering caused by suspended particles (such as sediments and plankton) exacerbates the attenuation of light, further reducing the overall visibility [2]. In addition, the selective absorption of specific light wavelengths by water weakens the optical clarity [3]. In view of the above challenges, there is an urgent need to develop effective methods to enhance the image quality of underwater images, thereby improving the effective use of underwater visual data.Image enhancement algorithms are mainly divided into traditional pixel-level processing methods, deep learning-based processing strategies, and image fusion-based methods. Traditional underwater image enhancement methods mainly rely on image processing technology to improve the visual quality of images. These methods usually focus on enhancing contrast, reducing noise, and achieving color balance at the image level. Commonly used methods include histogram equalization, denoising filtering, and color correction.

Histogram equalization (HE) [4] is a common algorithm for enhancing image contrast. This algorithm converts the histogram distribution of an image into an approximately uniform distribution through a cumulative distribution function, thereby expanding the grayscale range of the original image. The purpose of denoising filtering is to reduce the noise information in the image while retaining important details and structural information of the image. Common filtering methods include Gaussian filtering, median filtering, mean filtering, and bilateral filtering. The basic idea is to smooth the image to remove noise.Another pixel-level processing method is to use underwater optical imaging models to deal with the inherent defects of underwater imaging at the optical level. For example, underwater wavelength compensation and image defogging methods, restoration methods based on image blur and light absorption, and underwater image enhancement methods based on Retinex theory.

In order to improve the image quality improvement effect of underwater image enhancement algorithms, researchers have introduced some advanced results in the field of deep learning, such as convolutional neural networks and generative adversarial networks. Deep learning technology can automatically learn and extract image features, so after learning enough sample data, the neural network can restore the features of underwater images. Convolutional neural networks have been widely used in the field of underwater image enhancement in recent years.

Deep learning technology can automatically learn and extract image features, so after learning enough sample data, the neural network can restore the features of underwater images. Convolutional neural networks have been widely used in the field of underwater image enhancement in recent years. Wang et al. [5] proposed an end-to-end framework UIE-net for underwater image enhancement, which can simultaneously perform color correction and dehazing for underwater images. Although this method has achieved good results on benchmark datasets, the performance of the algorithm is greatly affected by the quality of training data when applied in actual scenarios, and more diverse and high-quality datasets are needed to improve the performance of the algorithm. Li et al. [6] proposed an underwater image enhancement network Ucolor, which corrects color cast and enhances contrast of images through a multi-color space embedding method guided by medium transmission. In order to effectively improve the image processing speed, Naik et al. [7] proposed a shallow network structure Sahllow-UWnet, which can significantly improve the processing speed

while maintaining performance. UICE2-Net [8] is the first underwater image enhancement algorithm that uses deep learning in both RGB and HSV color spaces. The network consists of three modules: RGB pixel module, HSV global adjustment module and an attention map module.

Generative adversarial networks and diffusion models generate clear output images from underwater raw images based on the principle of image generation. For example, Li et al. [9] proposed a two-stage generative adversarial network WaterNet, which was trained with a dataset of contrasting images in air and water, and finally achieved color correction of monocular underwater images. Islam et al. [11] introduced a conditional generative adversarial network model FUNIE-GAN, which significantly improved the image generation rate and can enhance underwater images in real time. The test results of multiple datasets confirmed the feasibility of the algorithm. Guan et al. [12] used a conditional denoising diffusion probability model DifWater to enhance underwater images and integrated color compensation as a conditional guide. In order to solve the attenuation and scattering problems of underwater light, Wang et al. [13] used an unsupervised generative adversarial network to generate underwater images. The U-net structure in the network was trained on a synthetic underwater image dataset and performed well on real data, but its robustness in processing underwater images of complex scenes was limited.

In addition to the above methods, strategies based on image fusion have also shown good performance in image enhancement tasks. Ancuti et al. [14] first introduced a fusion strategy to improve the quality of underwater images. In their proposed algorithm, color-corrected and contrast-enhanced images are generated from the original blurred underwater images, and these processed images are used as inputs in the fusion stage to synthesize enhanced images through multi-scale pyramid fusion theory. Subsequently, they improved the performance of the algorithm by optimizing the white balance correction method and introducing the defogging operation, making it more stable in extreme environments such as turbid seawater [15]. Gao et al. [16] also proposed a new underwater image enhancement algorithm based on multi-scale fusion theory, and improved the weight map in the fusion process by imitating the human visual system. Song et al. proposed a strategy that combines multi-scale fusion with global stretching of the model, and adopted an updated saliency weight coefficient method to fuse contrast and spatial cues to improve fusion quality. Kang et al. [17] combined multi-path input, multi-feature fusion, and attention mechanism to propose a high underwater image enhancement framework SPDF, which significantly improved image quality on the dataset.

Considering the cost, weight, and convenience of information acquisition, cameras based on optical imaging are one of the important sensors for spatial position prediction. Intelligent robots, unmanned vehicles, augmented reality devices, etc. on land are generally equipped with one or more cameras to predict their own movement. The optimization-based strategy is widely used in visual SALM. The optimization algorithm can make full use of historical poses and landmarks, and show better performance in large-scale and long-time series tasks. Qin et al. [18] proposed a visual-inertial real-time positioning algorithm VINS-Mono, which only requires a very low-cost monocular camera and an inertial measurement unit IMU to achieve state estimation of the six-degree-of-freedom pose of the body. The algorithm implements a visual-inertial tightly coupled nonlinear optimizer, which calculates a more accurate spatial pose change by minimizing marginal information, inertial measurement residuals, and visual reprojection errors. On this basis, Qin et al. [19] integrated multiple sensors such as GPS signals and depth cameras to realize a multi-sensor pose state estimation algorithm VINS-Fusion based on optimization strategies, which further promoted and applied this strategy.

Although the above algorithms show good performance on land, they have not been tested and verified much in underwater environments. In order to improve underwater visual positioning, some scholars have tried to improve the quality of images. For example, Xin et al. [20] proposed an end-to-end network for SLAM preprocessing in underwater low-light environments. By enhancing low-light images and self-supervised learning to improve feature point matching, the performance of VSLAM based on feature point extraction was effectively improved.

The harsh underwater visual conditions bring certain challenges to the visual positioning algorithm. Although there are related studies on underwater image enhancement, there are not many systems that add image enhancement to underwater visual positioning. Based on this, this paper introduces a multi-scale image enhancement algorithm into the traditional visual inertial positioning algorithm to improve the underwater positioning accuracy. The specific contributions are as follows:

1. The proposed underwater image enhancement preprocessing algorithm consists of two straightforward stages. Initially, a white balance color correction is applied to each sub-image based on the gray world assumption, followed by a restrained contrast enhancement, yielding two optimized sub-images. In the subsequent stage, these sub-images are integrated using multi-scale fusion techniques to produce a final image that is of higher quality and enhanced for clarity.

2. To address the problems of poor underwater image quality, small number of feature extractions and unstable tracking, a multi-scale fusion algorithm is used to enhance image features in the image preprocessing stage, which significantly increases the number of feature point extractions and improves the stability of system positioning.

3. In order to compare the performance of the improved algorithm, this paper runs the VINS algorithm on the real underwater dataset Aqualoc. In the dynamic low-light underwater visual environment, the dataset test results show that the proposed algorithm.

2 METHOD

2.1 Fusion Based Underwater Image Enhancement Algorithm

2.1.1 Color correction

Light in water is absorbed and reflected by the medium and gradually attenuates. Different wavelengths of light attenuate in water to different degrees. As the absorption of visible light by water increases with the increase of wavelength, the blue with the shortest wavelength propagates the longest distance in water, while the red with the longest wavelength propagates the shortest distance in water. Therefore, most underwater images we see appear blue-green [21]. In the color correction stage, we introduced the grayscale world theory, which assumes that in a natural environment, regardless of the color type, the average brightness of the overall image tends to be neutral gray, that is, the mean values of the red, green and blue channels of the image should be approximately equal. If the three channels of the original image satisfy the grayscale world hypothesis, then the following equations are satisfied:

$$\frac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} R(u,v) = \frac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} G(u,v) = \frac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} B(u,v) \tag{1}$$

where i and j represent the values of the horizontal and vertical pixel position index values of the image, respectively, and H and W represent the height and width of the image. When an underwater image I_0 is acquired, its color correction can be performed using the above assumptions. First, the target grayscale value needs to be calculated. In order to obtain the target grayscale value, the average values of the three channels are calculated respectively.

$$\begin{cases} \mu_R = \dfrac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} I_0^R(u,v) \\[2mm] \mu_G = \dfrac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} I_0^G(u,v) \\[2mm] \mu_B = \dfrac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} I_0^B(u,v) \end{cases} \tag{2}$$

When an underwater image I_0 is acquired, its color correction can be performed using the above assumptions. First, the target grayscale value needs to be calculated. In order to obtain the target grayscale value, the average values of the three channels are calculated respectively.
and then the average value of the three channels is taken:

$$\lambda = \frac{\mu_R + \mu_G + \mu_B}{3} \tag{3}$$

each element is normalized according to the mean to achieve color correction

$$I_0^c(i,j) = \left(\frac{I_0^c(i,j)}{\mu_c}\right) \cdot \lambda \, (c \in \{R,G,B\}) \tag{4}$$

2.1.2 Contrast increasement
Light in water medium is affected by absorption, scattering and refraction, so the attenuation rate is much faster than that in land environment. In the case of insufficient light, the contrast of the image is low. At the same time, water contains a large number of suspended particles, forming water mist, which makes the object more blurred. In order to solve this problem, the strategy of grayscale equalization is introduced. Artificial light sources are often used for fill light in underwater images, so there will be obvious bright and dark areas. In order to prevent detail loss and highlight overflow caused by excessive enhancement, limited contrast histogram equalization is used. The main idea is to introduce contrast limitation parameters on the basis of the principle histogram equalization, so as to control the output dynamic range, so as to ensure that the detail characteristics of the bright part can be retained while the brightness of the dark part is improved.
Calculate the histogram $H[k]$ of the input image I_0, which represents the probability of the gray value k appearing, and then calculate the cumulative distribution function $F(k)$.

In order to limit the excessive enhancement of contrast, a truncation threshold of contrast limitation is introduced to truncate the part exceeding the threshold to prevent the final image from being overexposed. The cumulative distribution function after processing is:

$$F^{'}(k) = \begin{cases} F(k), & F(k) \leq T, \\ T, & F(k) > T. \end{cases} \tag{5}$$

map the corrected cumulative distribution function to the target pixel value range:

$$y(x) = min\big(max\big(M \cdot F^{'}(x), 0\big), M\big) \tag{6}$$

where M is the maximum value of the target grayscale range (usually $M = 255$). $F^{'}(x)$ is the truncated CDF value (normalized to [0,1]).

2.1.3 Image fusion
Complex underwater scenes pose challenges to the effective acquisition of visual information. To address the problems of color distortion, water mist, and low contrast in images, we proposed a white balance algorithm based on the grayscale vision hypothesis and a limited contrast map histogram equalization algorithm. In order to retain the enhanced effective feature information, we introduced a multi-scale image fusion algorithm.
Image Pyramid is a theory that uses a multi-scale hierarchical structure to represent images. The size and clarity of an image are representations of its scale. The bottom of the image pyramid is the original high-resolution image, and each layer upward is the result of downsampling the image. Therefore, the resolution and size of the image are gradually

reduced, and the visual features are gradually refined. The theory was first proposed by Burt et al. [22], who first introduced the image pyramid to achieve efficient decoding and encoding of images.

Assume that an image is described by a matrix with R rows and C columns. Each pixel represents the light intensity I at the corresponding position. The value of light I ranges from 0 to $K-1$. The original image is defined as the 0th layer G_0 of the Gaussian pyramid.

The first layer G_1 of the pyramid is a reduced low-pass filtered version of G_0. Here, the image G_0 is convolved with the Gaussian kernel for smoothing:

$$G_1'(u,v) = \sum_{i=-k}^{k} \sum_{i=-k}^{k} w(i,j) I(u+i,v+j) \tag{7}$$

Where $w(i,j)$ is the weight of the Gaussian kernel.

Then the resulting image G_1' is scaled down, usually by removing all even-numbered rows and columns, to obtain

$$G_1(u,v) = G_1'(2u,2v) \tag{8}$$

Repeat the above process until the preset Nth layer image G_N is obtained. The Gaussian pyramid performs low-pass filtering on the image in terms of frequency, removes high-frequency information, and retains the overall structure and low-frequency information of the image. On this basis, the Laplacian pyramid can be calculated.

Each layer of the Laplacian pyramid is obtained by subtracting two adjacent layers of the Gaussian pyramid. The G_1 layer image of the Gaussian pyramid is upsampled and smoothed to obtain

$$\widehat{G_0}(u,v) = 4 \sum_{i=-2}^{2} \sum_{j=-2}^{2} w(i,j) G_1\left(\frac{u+i}{2}, \frac{v+i}{2}\right) \tag{9}$$

The 0th level L_0 of the Laplacian pyramid is

$$L_0 = G_0 - G_1 \tag{10}$$

Going up layer by layer, we can get $L_1, \dots, L_{N-1}$. The Laplacian pyramid obtains the high-frequency part of the image, that is, the edge details of the original image at different scales (Figure 1).

$$G_k = L_k + \text{upsample}(G_{k+1}) \tag{11}$$

The decomposition of the image pyramid is reversible. Through the top-level Gaussian image G_N and the Laplacian pyramid, the following formula is used to calculate layer by layer:

$$G_k = L_k + \text{upsample}(G_{k+1}) \tag{12}$$

Finally can be restored to the original original image:

$$G_0 = L_0 + \text{upsample}(L_1 + \text{upsample}(L2 + \cdots)) \tag{13}$$
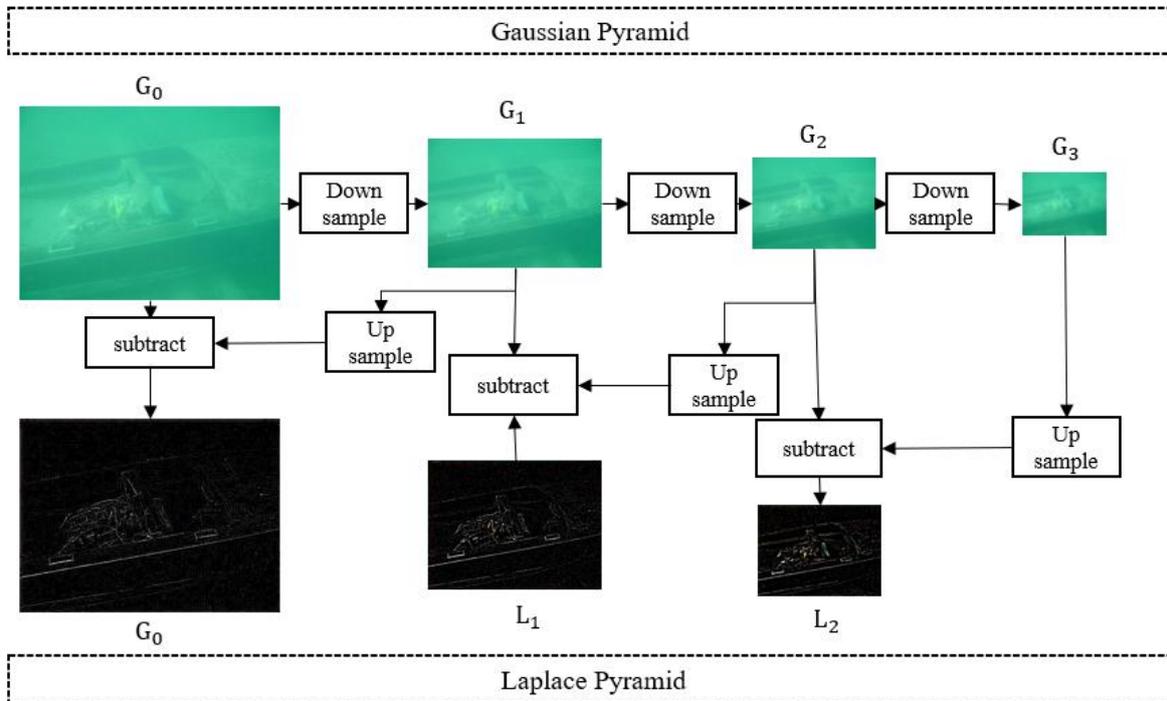
Figure 1 Image Gaussian Pyramid and Laplacian Pyramid Decomposition

The fusion process of the two sub-images requires the introduction of appropriate weight factors. The weight factor here is a matrix of the same size as the image, and the value of each element represents the proportion of the sub-image in the final fused image. In the fusion process, in order to retain high-quality features, the weight is calculated from the sub-image's salient features W_S, local contrast W_{LC}, global contrast W_L, and exposure W_E [15]. Calculate the above weights for the two subgraphs and get the total weight factor of each graph by adding them up

$$W^k = W_L + W_{LC} + W_S + W_E \tag{14}$$

In order to avoid the impact of different orders of magnitude in the weight calculation process, the final weight factor is normalized

$$\overline{W}^k = \frac{W^k}{\sum_{k=1}^K W^k} \tag{15}$$

Each input sub-image is decomposed layer by layer into a multi-scale Gaussian pyramid $G_1, \ldots, G_{N-1}$ by convolution with a Gaussian kernel and downsampling, and then the Laplacian pyramid $L_1, \ldots, L_N$.is obtained by subtracting the low-pass filtered image. In the entire pyramid data structure, G_l and L_l each represent the corresponding l-th layer. The final image is calculated by mixing the input image and weights in a multi-layer pyramid manner

$$R_l(x) = \sum_{k=1}^K G_l\{\overline{W}_k(x)\}L_l\{I_k(x)\} \tag{16}$$

Where l represents the number of layers of the pyramid, and k is the index value of the input sub-image. Here, the normalized weight factor is decomposed into the Gaussian pyramid $G_l\{\overline{W}_k(x)\}$, and the input sub-image is decomposed into the Laplacian pyramid $L_l\{I_k(x)\}$.

The advantage of using the multi-scale image fusion method is that it can retain the dominant features in the input image during the fusion process. As shown in Figure 2, the upper left corner of the chart shows the input image, and the lower right corner shows the output image. The input image is a typical underwater photo, which is characterized by defects such as color deviation, fog, blur, and insufficient lighting. Through the multi-scale image fusion method, these defects can be effectively improved while retaining or enhancing the important features in the image, thereby obtaining a higher quality output image.
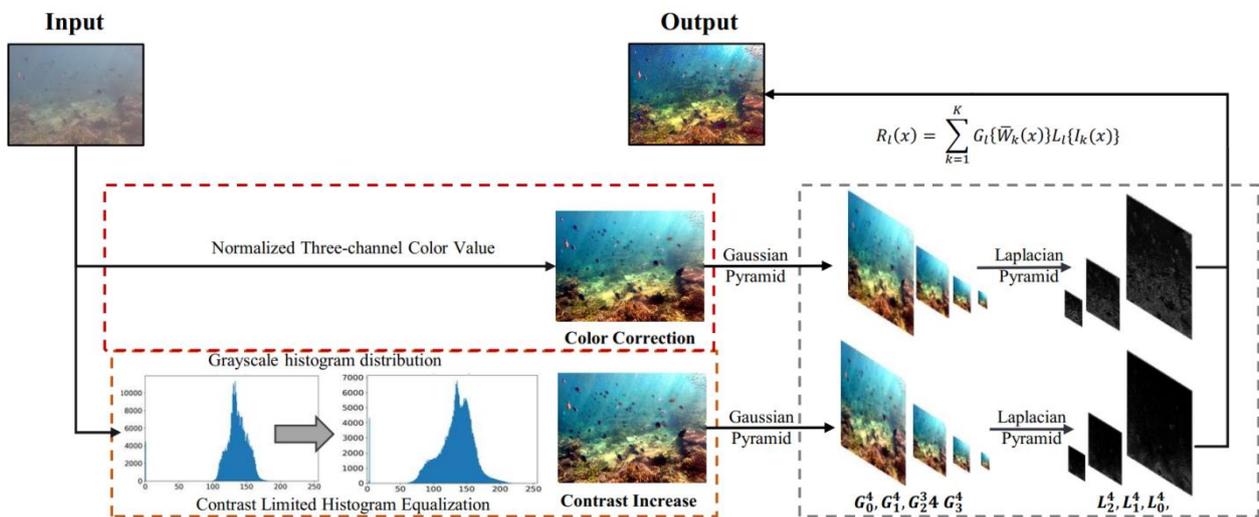
Figure 2 Underwater Fusion Based Image Enhancement Algorithm

2.2 Improve Underwater Visual Front by Image Enhancement

The method of feature point extraction in the VINS algorithm is the Shi-Tomasi algorithm. Its basic idea is to detect feature points from points in the image where the brightness changes dramatically, such as edge intersections, and return the positions of points with large gradient changes in the two-dimensional image as the coordinates of the feature points. However, this algorithm has problems such as low feature density and lack of rich feature description information.

The algorithm performs well on land, but the contrast of underwater images is low due to light scattering and absorption, which leads to smaller gradient changes in the image, weaker corner points, unstable feature detection or insufficient key points detected. In addition, underwater suspended matter can cause visual interference, and Tomasi feature detection is less stable against noise and motion blur, which can lead to detection errors. Based on the above factors, the underwater image enhancement proposed in this paper is adopted. As shown in Figure 3, the enhanced image has been significantly improved in contrast and feature details.
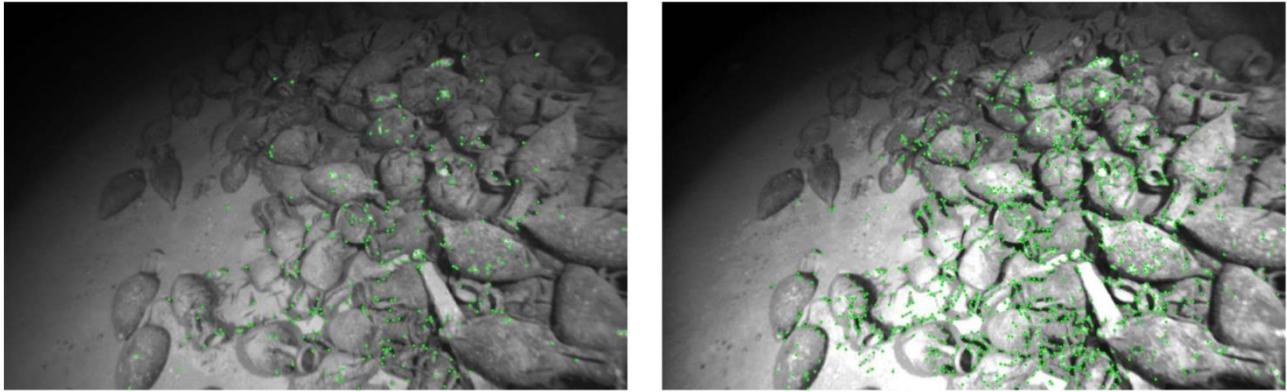
Figure 3 Shi-Tomasi Corner Detection of Images before and after Enhancement

In the original image on the left, the maximum number of feature points detected by Shi-Tomasi is 276, while in the enhanced image on the right, the maximum number of feature points returned by Shi-Tomasi is increased to 1215, which is a 340% increase in the maximum number of feature detections. Due to the enhancement of contrast and the prominence of edge characteristics, the number and quality of corner points detected have been significantly improved. In addition, the temporal consistency of corner points obtained in the enhanced image is improved, which is conducive to subsequent visual tracking and position solving.

In the process of matching visual feature points, although the tracking accuracy is significantly improved by the image enhancement algorithm, there are still a lot of matching errors caused by illumination distortion, weak texture, non-rigid deformation, suspended particles or object occlusion in long-term dynamic changing scenes. The discrete points generated by the above situations will destroy the consistency of geometric constraints, resulting in the failure of subsequent camera pose estimation tasks. In order to solve this problem, after visual feature matching, the random sampling consensus algorithm (RANSAC) is used to filter out the internal points that meet the geometric constraints from the set of candidate matching points containing noise, which is beneficial to the subsequent spatial solution.

The matching feature points in adjacent frames are the projections of the same spatial feature in different images, thus satisfying the epipolar constraint as shown in Figure 4. The matching pixel points in the previous frame image I_A, and the subsequent frame image I_B are denoted as p_1 and p_2. According to the pinhole camera model, the following relationship can be obtained:

$$s_1 p_1 = KP, s_2 p_2 = K(RP + t) \tag{17}$$
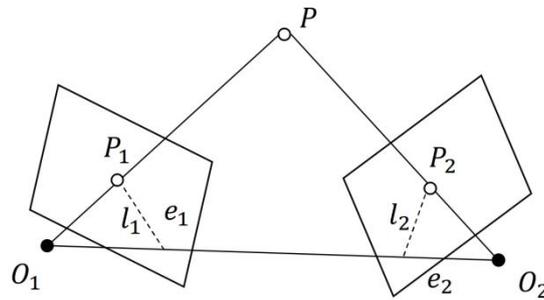
Figure 4 Epipolar Constraints of Feature Points in Adjacent Frames

where K is the camera intrinsic matrix, and R, t represent the rotation matrix and translation vector between the two coordinate systems. Through geometric constraints, we obtain:

$$E = t \times R, F = K^{-T} E K^{-1}, x_2^T E x_1 = p_2^T F p_1 = 0 \tag{18}$$

where F is called the fundamental matrix (Fundamental Matrix), E is called the essential matrix, and by solving the essential matrix, the spatial motion R, t of the camera can be estimated.

$$[x_i \quad y_i \quad 1] F \begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix} = 0 \tag{19}$$

In order to find the point pairs that satisfy the geometric constraints in the paired point set, a random sampling consistency check is performed on the point set. Firstly, 8 pairs are randomly selected from the input point pair set, and the geometric constraint formula is expanded into a homogeneous coordinate form.

$$Af = 0 \tag{20}$$

So get the following overdetermined linear equation

Where $A \in R^{n \times 9}$, $f \in R^9$ is the expansion vector of F. The above equation is solved by the singular value decomposition of $A^T A$ to obtain the least squares solution, so that $\sum_{i=1}^{n} \left\| Af \right\|^2$ is minimized, and the candidate basic matrix F' is calculated. All point pairs in the pairing point set are verified using geometric constraints:

$$x_i^T \widehat{F} x_j < \varepsilon \tag{21}$$

Where ε is the threshold of the reprojection error. If it is less than this threshold, it is an inlier, and if it is greater than this threshold, it is an outlier. Repeat the above steps and select the model with the largest number of inliers as the final solution. In underwater scenes, a large number of matching points may be mismatched points caused by bubbles, suspended particles or uneven lighting. By introducing the RANSAC algorithm to filter discrete values, only high-confidence matching points that meet geometric constraints are retained.

In order to improve the visual extraction and tracking performance under harsh underwater conditions, this paper preprocesses the original image obtained underwater with a fusion-based image enhancement algorithm. The comparison of the corner point detection results of the original image and the enhanced image shows that the algorithm can greatly improve the quantity and quality of feature point detection. Considering the underwater dynamic environment, a random sampling consistency algorithm is used to perform geometric constraint filtering on the set of matching points to reduce the false pairing caused by suspended particles and lighting effects.

3 EXPERIMENTS & ANALYSIS

This paper proposes a method to improve the VINS visual front-end based on image enhancement to improve its underwater visual feature tracking effect. This section will analyze the algorithm performance in terms of the number of visual feature points extracted and positioning accuracy.

To more fully demonstrate the algorithm's adaptability improvement effect in underwater scenes, we conducted comparative experiments on the Aqualoc dataset. The Aqualoc dataset was collected underwater by a remotely operated vehicle (ROV) equipped with a monocular camera and inertial sensors [23]. The true value of its movement trajectory was reconstructed and estimated in three dimensions using Colmap, providing a benchmark for algorithm evaluation.

3.1 Feature Point Extraction Performance

The Aqualoc dataset consists of sensory data collected in various underwater scenarios using a monochromatic camera, a low-cost MEMS-IMU, and an embedded computer. The archaeological site sequences were recorded in the Mediterranean Sea, off the coast of Corsica. To validate the effectiveness of the aforementioned image enhancement algorithm as a visual front-end for feature extraction and tracking, the third and seventh trajectories from the archaeological site were selected. Shi-Tomasi corner detection was performed on the images from these trajectories to compute the maximum number of high-quality feature points detectable in the images before and after enhancement. The test results are shown in Figure 5 below, where from top to bottom the 3rd and 7th sequences of the archaeological site.
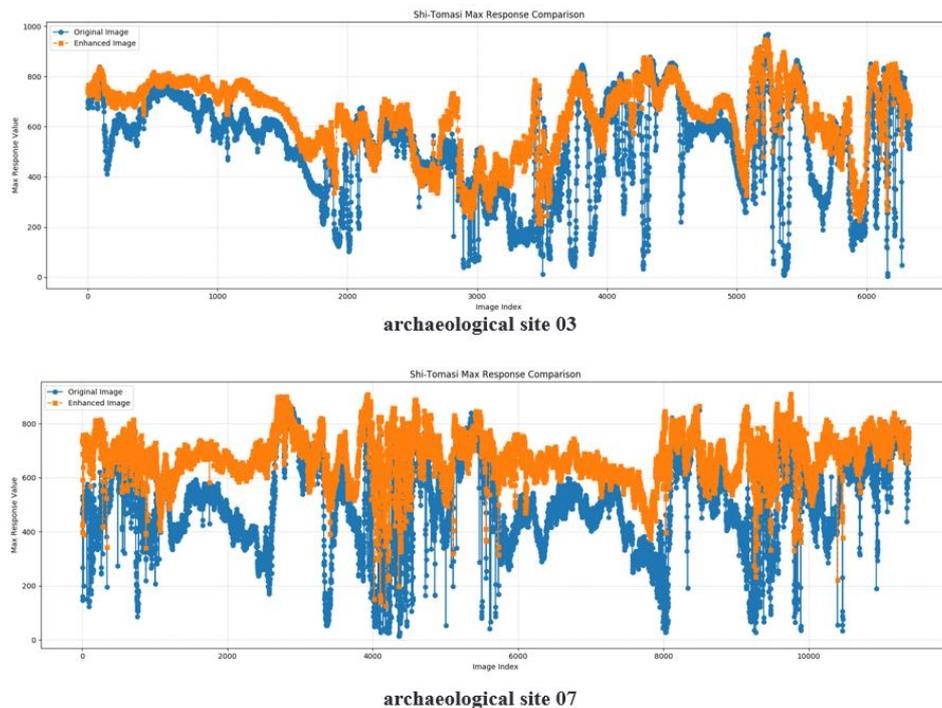

archaeological site 03

archaeological site 07

Figure 5 Comparison of Shi-Tomasi Feature Detection Max Num on Archeological Site 03 and 07

The horizontal axis in the figure is the frame index value of the image, and the vertical axis is the maximum number of feature points that can be extracted by the shi-tomasi feature detector with the same parameters. The blue line represents the original image, and the orange line represents the enhanced image. The third scene of the archaeological site in the above figure is located at a depth of about 270m on the real seabed. There is a wreckage of an antique shipwreck here. Most of the ground in this scene is flat, and there are many small rocks, so the visual texture is mainly repeated. In this scene, the turbidity is low, and the rolling of tiny sand grains further hinders the line of sight, making visual feature extraction more difficult. In general, the orange line is higher than the blue line in most cases, especially when the image index is about 4000 and 5300, the improvement effect is very significant, indicating that it is easier to extract high-quality feature points from the enhanced image.

The seventh scene of the archaeological site in the figure below is 380m below the seabed. There is a two-ear bottle mountain and the top of the mountain is several meters higher than the surrounding seabed level, so there is a certain degree of ups and downs. There are low texture characteristics on the sand around the two-ear bottle mountain. Due to the presence of two-ear bottles, there are more marine wildlife, so the environment is very dynamic. It can also be seen in the figure that the orange line is higher than the blue line in most cases, especially when the real image index is 2000, 4000, 6000 to 8000, and there are a large number of blue lines where the maximum number of feature points is less than 200. This situation will be extremely unfavorable for subsequent pose solution. Most of the orange lines of the enhanced image are above 200.

In summary, through the test of the image sequence taken underwater on the real seabed, the results show that after the original image is enhanced based on fusion, the maximum number of features that can be extracted by the same shi-tomasi detector has been improved, which effectively proves the positive role of the image preprocessing algorithm in feature extraction.

3.2 Underwater Visual Positioning Performance

Integrating the fusion-based image enhancement algorithm into the VINS system can effectively improve the stability of the positioning system. The original algorithm and the improved algorithm were tested in multiple scenarios from the underwater dataset of the archaeological site. The operational status is shown in Figure 6. As can be seen from the figure, the improved image algorithm extracts more feature points and obtains richer point clouds, which is beneficial for subsequent pose estimation.

VINS

Raw Image Tracking

Enhanced-VINS

Enhanced Image Tracking

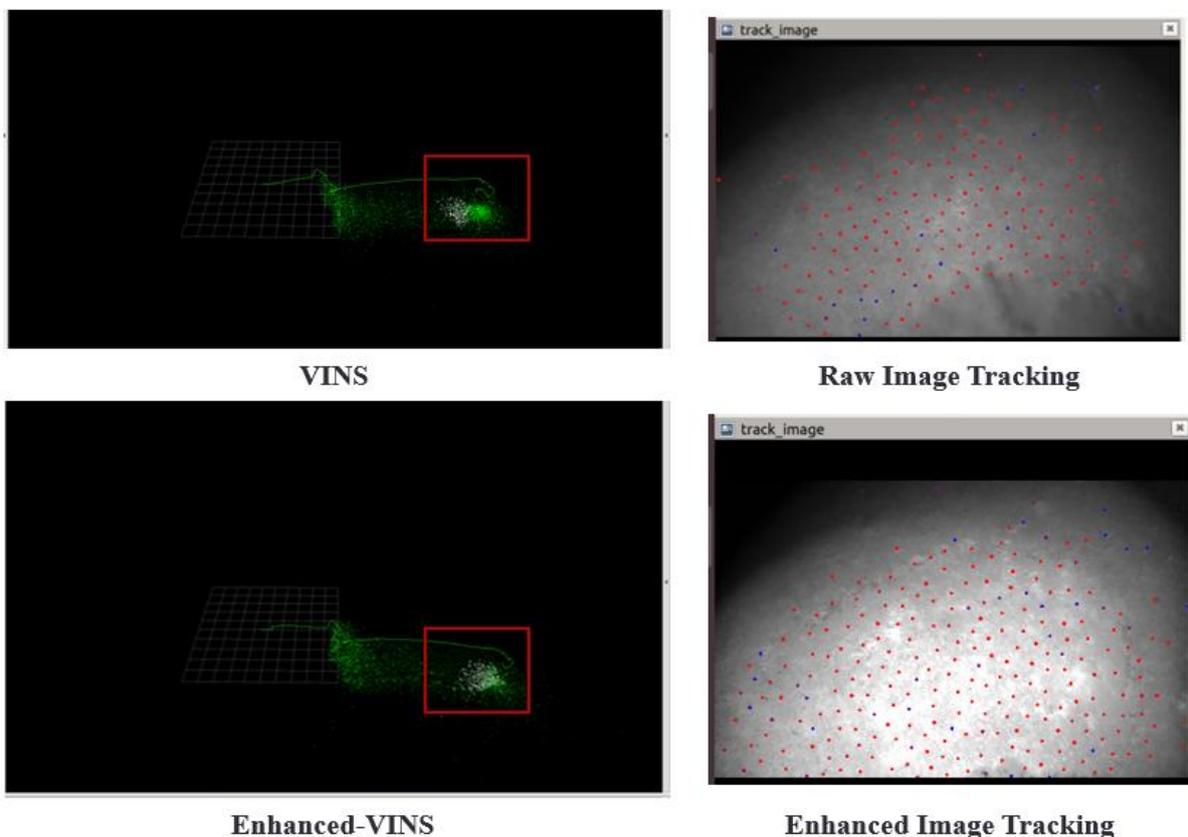**Figure 6** Comparison of VINS and Enahnced-VINS on archaeological site sequence 05

In order to further compare the improvement effect of the image enhancement algorithm on the VINS algorithm, we will perform the improved algorithm and the VINS algorithm on multiple test sequences on the archaeological scene to evaluate their performance differences under different conditions. The test sequences include A2, A5, A6, A7, A8 and

A10. For each sequence, we calculated the minimum (Min), median (Medium) and maximum (Max) absolute trajectory error to comprehensively evaluate the stability and reliability of the two algorithms. The experimental results are shown in Table 1.

Table 1 Absolute Trajectory Error Comparison between VINS and EN-VINS

Sequence	VINS			EN-VINS		
	Min	Medium	Max	Min	Medium	Max
A2	0.110000	1.037429	2.198876	0.181079	0.871403	2.199951
A5	0.172183	0.542419	2.322855	0.029175	0.320848	2.109739
A6	0.169899	0.291294	4.542645	0.008474	0.140198	3.271086
A7	0.067507	0.920890	5.552000	0.110673	1.161439	5.233549
A8	/	/	/	0.062034	0.513904	1.538490
A10	/	/	/	0.130057	0.729954	3.357582

According to the chart analysis, the performance of the EN-VINS algorithm on multiple sequences is better than that of VINS. Specifically, in the four sequences A2, A5, A6 and A7, the minimum, median and maximum ATE values of EN-VINS either surpassed VINS in all aspects or most indicators were ahead of VINS. Especially in the A5 and A6 sequences, EN-VINS not only performed outstandingly in terms of stability (reflected by the smaller maximum value), but also showed significant advantages in terms of accuracy (reflected by the lower median and minimum values). In addition, for the two sequences A8 and A10, VINS system failed to obtain the location information, while EN-VINS maintained the integrity of the data in these two sequences, further demonstrating its higher reliability.

Overall, EN-VINS showed higher comprehensive performance than VINS throughout the test process, providing more reliable results both under normal and extreme conditions. It is particularly noteworthy that EN-VINS can still provide stable and accurate positioning results in the face of situations that may cause VINS positioning failure, which shows that EN-VINS has stronger robustness and adaptability.

4 CONCLUSION

The harsh underwater visual environment is a major challenge for visual positioning algorithms. This paper introduces an image multi-scale fusion strategy to enhance the underwater image preprocessing, which significantly improves the image contrast and detail features. Combining this image enhancement algorithm with the traditional visual inertial positioning algorithm VINS greatly increases the number of feature point extraction, improves the quality of feature point extraction, and achieves more stable visual tracking.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Xia H, Liao F, Bao B, et al. Perspective on Wearable Systems for Human Underwater Perceptual Enhancement, IEEE Transactions on Cybernetics, 2024.

[2] Raveendran S, Patil M D, Birajdar G K. Underwater image enhancement: a comprehensive review, recent trends, challenges and applications, Artificial Intelligence Review, 2021, 54: 5413-5467.

[3] Jian M, Liu X, Luo H, et al. Underwater image processing and analysis: A review, Signal Processing: Image Communication, 2021, 91: 116088.

[4] Pizer S M, Amburn E P, Austin J D, et al. Adaptive histogram equalization and its variations, Computer vision, graphics, and image processing, 1987, 39(3): 355-368.

[5] Wang Y, Zhang J, Cao Y, et al. A deep CNN method for underwater image enhancement, 2017 IEEE international conference on image processing (ICIP). IEEE, 2017: 1382-1386.

[6] Li C, Anwar S, Hou J, et al. Underwater image enhancement via medium transmission-guided multi-color space embedding, IEEE Transactions on Image Processing, 2021, 30: 4985-5000.

[7] Naik A, Swarnakar A, Mittal K. Shallow-uwnet: Compressed model for underwater image enhancement (student abstract), Proceedings of the AAAI Conference on Artificial Intelligence. 2021, 35(18): 15853-15854.

[8] Wang Y, Guo J, Gao H, et al. UIEC^ 2-Net: CNN-based underwater image enhancement using two color space. Signal Processing: Image Communication, 2021, 96: 116250.

[9] Li J, Skinner K A, Eustice R M, et al. WaterGAN: Unsupervised generative network to enable real-time color correction of monocular underwater images. IEEE Robotics and Automation letters, 2017, 3(1): 387-394.

[10] Zhu J Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks//Proceedings of the IEEE international conference on computer vision. 2017: 2223-2232.

[11] Islam M J, Xia Y, Sattar J. Fast underwater image enhancement for improved visual perception, IEEE Robotics and Automation Letters, 2020, 5(2): 3227-3234.

[12] Guan M, Xu H, Jiang G, et al. DiffWater: Underwater image enhancement based on conditional denoising diffusion probabilistic model, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2023, 17: 2319-2335.

[13] Wang N, Zhou Y, Han F, et al. UWGAN: Underwater GAN for real-world underwater color restoration and dehazing, arXiv preprint arXiv:1912.10269, 2019.

[14] Ancuti C, Ancuti C O, Haber T, et al. Enhancing underwater images and videos by fusion, 2012 IEEE conference on computer vision and pattern recognition. IEEE, 2012: 81-88.

[15] Ancuti C O, Ancuti C, De Vleeschouwer C, et al. Color balance and fusion for underwater image enhancement, IEEE Transactions on image processing, 2017, 27(1): 379-393.

[16] Guo P, Zeng D, Tian Y, et al. Multi-scale enhancement fusion for underwater sea cucumber images based on human visual system modelling, Computers and Electronics in Agriculture, 2020, 175: 105608.

[17] Kang Y, Jiang Q, Li C, et al. A perception-aware decomposition and fusion framework for underwater image enhancement, IEEE Transactions on Circuits and Systems for Video Technology, 2022, 33(3): 988-1002.

[18] Qin T, Li P, Shen S. Vins-mono: A robust and versatile monocular visual-inertial state estimator, IEEE transactions on robotics, 2018, 34(4): 1004-1020.

[19] Qin T, Cao S, Pan J, et al. A general optimization-based framework for global pose estimation with multiple sensors, arXiv preprint arXiv:1901.03642, 2019.

[20] Xin Z, Wang Z, Yu Z, et al. ULL-SLAM: underwater low-light enhancement for the front-end of visual SLAM, Frontiers in Marine Science, 2023, 10: 1133881.

[21] Singh N, Bhat A. A systematic review of the methodologies for the processing and enhancement of the underwater images, Multimedia Tools and Applications, 2023, 82(25): 38371-38396.

[22] Burt P J, Adelson E H. The Laplacian pyramid as a compact image code, Readings in computer vision. Morgan Kaufmann, 1987: 671-679.

[23] Ferrera M, Creuze V, Moras J, et al. AQUALOC: An underwater dataset for visual–inertial–pressure localization, The International Journal of Robotics Research, 2019, 38(14): 1549-1559.

AGRICULTURAL DISEASE AND PEST DETECTION BASED ON MACHINE VISION

ZePing Liu[1*], KeXin Sun[1], LeShui Qiao[1], XinTao Wang[1], ChuanDi Xu[1], ZhongPeng Zhang[1], GuiSheng Miao[2], YuShuo Han[3], Yuan Ji[4], Xiang Ji[5]

[1]*Trine Engineering Institute, Shaanxi University of Technology, HanZhong 723000, Shaanxi, China.*
[2]*School of Physics and Telecommunication Engineering, Shaanxi University of Technology, HanZhong 723000, Shaanxi, China.*
[3]*School of Economics and Management, Henan Institute of Science and Technology, XinXiang 453000, Henan, China.*
[4]*College of Engineering, University of Illinois Chicago, Chicago 60607, America.*
[5]*Academy of Martial Arts and Performance, Capital University of Physical Education And Sports, Beijing 100091, China.*
Corresponding Author: ZePing Liu, Email: 15838575027@163.com

Abstract: With the growth of the global population and the rise of agricultural demand, crop pest monitoring and control has become a key link to ensure food security. The traditional manual monitoring method has low efficiency and poor accuracy, which is difficult to meet the needs of modern agriculture. Therefore, a monitoring system for agricultural pests and diseases based on machine vision was designed, which combined with image recognition, deep learning, embedded system and wireless communication technology to realize real-time monitoring and automatic identification of pests. The experimental results show that the identification accuracy of common pests is 92%, the data transmission is stable, and the average response time is 10 seconds, which can meet the real-time monitoring needs of farmland. The introduction of solar powered systems further reduces maintenance costs and improves the sustainability of the system. This study provides an efficient and accurate solution for agricultural pest monitoring, which is of great significance for improving agricultural production efficiency and ensuring food security.
Keywords: Disease and insect pest detection; Machine vision; STM 32 microcontroller; Deep learning

1 INTRODUCTION

Agricultural pest control is a key link in ensuring global food security. As the world's population continues to grow, global food demand is expected to increase by 60% by 2050, placing greater demands on agricultural productivity. However, crop diseases and pests cause annual yield losses of up to 20 to 40 percent, posing a serious threat to food security. Traditional pest monitoring methods mainly rely on manual field inspection, which is not only time-consuming and laborious (each person can monitor only 2-3 acres of land per day), but also the recognition accuracy is generally lower than 70%. This monitoring method has a significant lag, often after a large outbreak of pests and diseases can be found, resulting in a 30% to 50% increase in control costs. Therefore, the development of intelligent and automated pest monitoring system has become an urgent need for the development of modern agriculture.

Foreign scholars have taken the lead in the exploration of machine vision technology in the field of agricultural pest monitoring. As mentioned in "AI Turns Agricultural 'Doctor' to Diagnose Crop Pests and Diseases": Researchers at Pennsylvania State University and the Swiss Federal Institute of Technology have built a system model and connected it to a cluster of computers to form a neural network. A database of more than 53,000 photos of healthy and diseased crops was created, including 14 crops and 26 diseases. The researchers used deep learning methods to 'train' the model to find all the visual data.[1]

At present, the main technical difficulties faced by pest monitoring are reflected in three aspects: First, in the complex farmland environment, light changes, branches and leaves occlusion and other factors will significantly affect the quality of image acquisition. Research shows that under natural lighting conditions, the recognition accuracy of traditional image processing algorithms will decrease by 15%-20%. Secondly, the existing monitoring system is generally lack of real-time, from data collection to analysis and decision-making often need several hours, cannot meet the needs of timely prevention and control. Third, the system integration is insufficient, and the hardware power consumption and computing performance are difficult to balance, resulting in long-term deployment difficulties in the field. These problems seriously restrict the popularization and application of intelligent monitoring technology.

In recent years, with the development of the Internet of Things, machine vision and artificial intelligence technology, three main technical routes have emerged in the field of agricultural pest monitoring. The first type is sensor network monitoring, through the deployment of environmental sensors such as temperature, humidity and soil to indirectly predict the risk of pests and diseases. This type of method is less costly, but it cannot directly identify the species of pests and diseases, and its accuracy is limited. The second category is spectral imaging technology, which uses multi-spectral or hyperspectral cameras to obtain crop physiological information. Although the recognition accuracy is high, the equipment is expensive and difficult to promote on a large scale. The third category is a deep learning-based machine vision approach that realizes pest and disease identification through convolutional neural networks. The

recognition accuracy of this method can reach more than 90% in the laboratory environment, but it still faces the challenges of insufficient model generalization and high computing resource demand in practical application.[2]

Through the systematic analysis of the existing literature, it is found that there are three obvious shortcomings in the current research: first, most systems adopt the "collection - upload - cloud processing" mode, limited by the field network conditions, the response time is often more than 10 minutes; Second, the existing solutions pay more attention to the recognition algorithm itself, and lack a complete system design from trapping to recognition; Third, the issue of energy supply has not been fully paid attention to, and more than 60% of field monitoring equipment cannot continue to work due to power supply problems. These limitations seriously affect the practical application effect of the technology.

The aim of this research is to break through the bottleneck of existing technology and design and implement a complete intelligent monitoring system for agricultural diseases and pests. The system has the following innovative features: First, the architecture of "edge computing + cloud collaboration" is adopted, and some computing tasks are delegated to embedded devices, so that the system response time is reduced to less than 10 seconds. Secondly, a closed-loop system of "surveillance-identification-prevention" is constructed by combining the UV trap lamp, atomized killing device and machine vision recognition innovatively. Third, through solar power and low power consumption design, to ensure that the system can work continuously in the field for more than 6 months. These innovations will effectively solve key problems in the practical application of existing technologies.

This research adopts the research method of "theoretical analysis - system design - experiment verification". First, through literature research and field investigation, clear technical requirements and system indicators; Then, the hardware platform and software algorithm are developed by modularization design. Finally, the performance of the system is verified by comparison experiment and field test. In terms of hardware, STM32F407ZG is selected as the main controller, with OV2640 camera and 4G communication module, to build a low-power embedded system. At the algorithm level, the lightweight CNN model is developed based on PyTorch framework, and the generalization ability is improved through data enhancement and transfer learning. After the system was integrated, field tests were carried out in three agricultural demonstration zones with different climatic conditions for a period of six months.

The structure of this paper is as follows: The second chapter introduces the overall design of the system in detail, including hardware architecture and software framework; The third chapter describes the core algorithm design and optimization process; The fourth chapter shows the system implementation details and key technical solutions; Chapter 5 verifies the performance of the system through experimental data; Chapter 6 summarizes the research results and looks forward to the future direction. Through this structure arrangement, the theoretical innovation and technical breakthrough of this research are presented comprehensively and systematically.

From the perspective of theoretical value, the "edge-cloud collaborative" architecture proposed in this study provides a new idea for the design of agricultural systems, and the lightweight identification model developed promotes the development of embedded AI technology. In terms of practical significance, the system can help farmers reduce the use of pesticides by more than 30%, and a single monitoring point can save about 5,000 yuan per year, with significant economic and ecological benefits. Subsequent research will focus on optimizing the robustness of the model in extreme weather and exploring the potential of 5G technology for agricultural monitoring.

This paper will introduce the overall architecture of the system, hardware design, software implementation, system integration and testing, experimental results and analysis in detail. Our research not only provides a novel and effective solution for agricultural pest monitoring, but also provides valuable experience and reference for future research and application in related fields.

2 OVERALL STRUCTURE DESIGN

As one of the most populous countries in the world, agriculture is intricately linked to human survival and livelihood, and pests and diseases have consistently been critical factors influencing crop yield. To enhance the efficiency of pest control and increase agricultural productivity, this paper proposes an electronic detection device based on machine vision for the remote identification and capture of pests. The design of the trapping device is illustrated in Figure 1 and primarily comprises two components: a hardware system and a software system.[4]
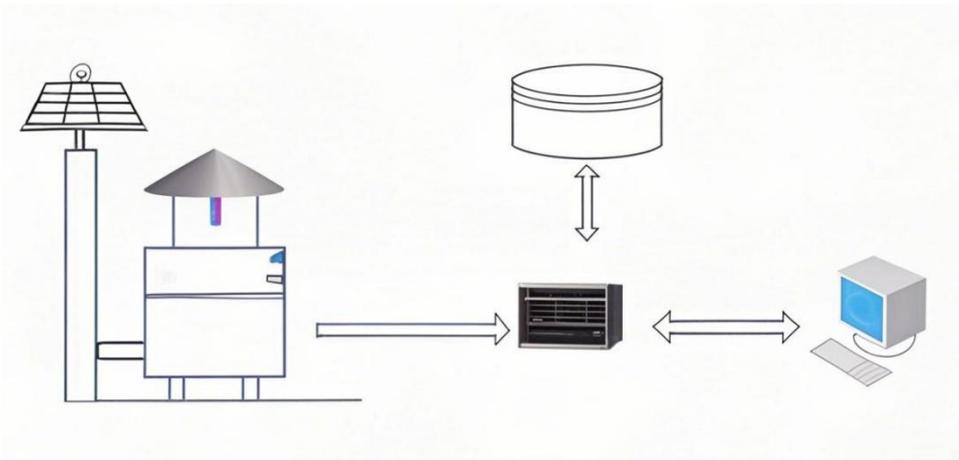
Figure 1 Model of Pest Trapping Device

2.1 Hardware System Design

In the agricultural pest monitoring system based on machine vision, hardware design plays a key role, which directly affects the quality of image acquisition, the stability of data transmission and the overall reliability of the system. This section will introduce our work in hardware design in detail, including the choice of core control unit, the choice of camera, the choice of communication mode, etc. Among them, the hardware system of the node-end detection device can be divided into solar power supply device, pest trapping device and image acquisition device.

2.1.1 Solar powered device[5]
The node-end detection device designed in this paper is powered by solar energy. Among them, the solar power supply device includes a solar panel, a light intensity detection module, a rotating bracket and a solar lithium battery. The solar panel and the rotating support are fixed by screws, and the light intensity detection module is installed directly above the solar panel. When working, the light intensity detection module detects the change of light intensity and rotates the rotating bracket to the best position through the control of two MG90S steering gear, so that the solar panel is facing the sun, its solar energy conversion efficiency reaches the maximum, and the energy converted by solar energy is stored in the solar lithium battery. When the pest trapping device and image acquisition device work, it provides the appropriate output voltage. Figure 2 shows the power distribution of each part of the detection device at the node, and Figure 3 shows the circuit diagram.
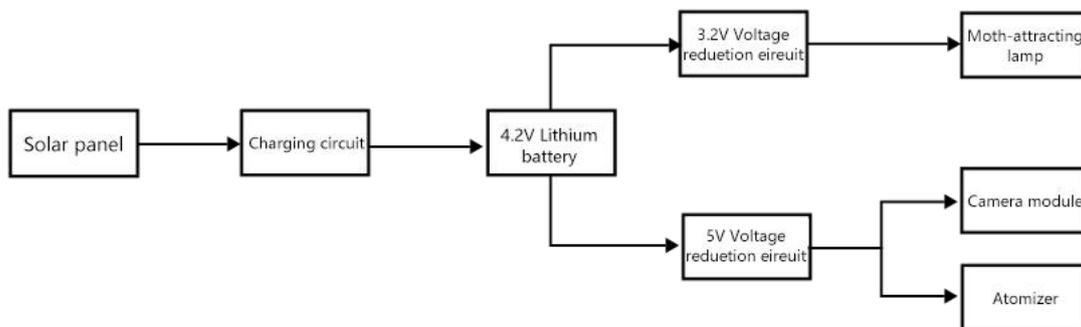
Figure 2 Power Module Diagram
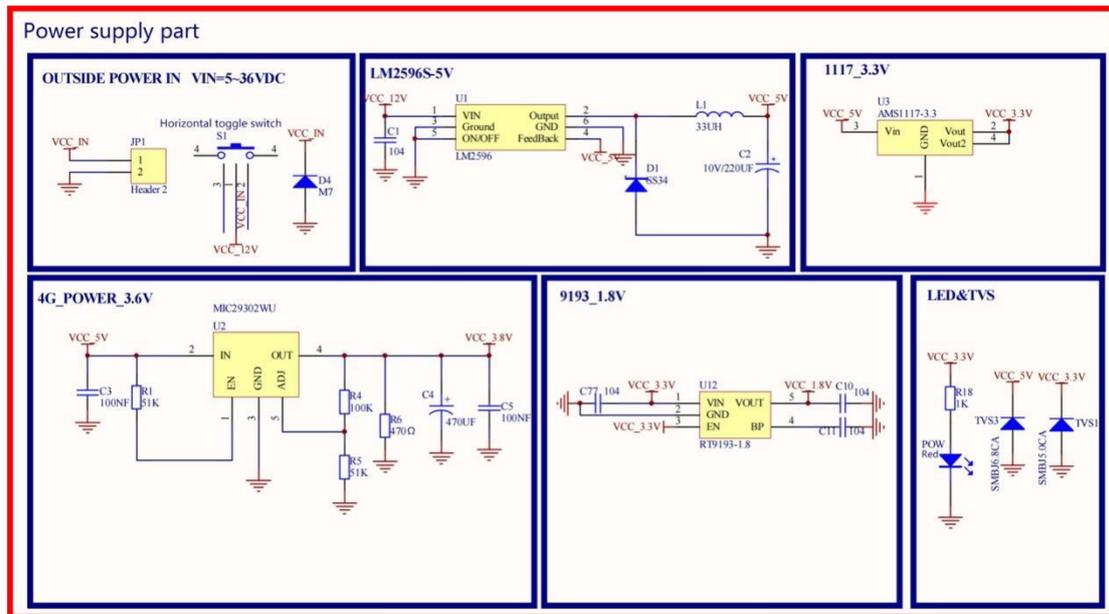
Figure 3 Power Circuit Diagram

2.1.2 Pest traps

The pest trapping device is designed as a light-controlled intelligent insect trapping device. The core of the device is a 3.2V/4W UV insect trapping lamp and three arc-shaped impact baffles. The device uses the BH1750FVI light intensity sensor to monitor the ambient illuminance in real time. When the detection value is lower than 5lux (night mode), the STM32F407ZG microcontroller starts the PWM dimming circuit, so that the insect trap lamp works at 85% duty cycle, and its specific wavelength of ultraviolet light can effectively attract Lepidus pests within a radius of 5 meters. When insects are attracted to the light source and fly towards the device, they collide with the acrylic baffle mounted at a 30° tilt. The nano-hydrophobic coating on the surface of the baffle reduces the adhesion of the insects, causing them to slide down to the insect-collecting funnel at the bottom. The end of the funnel is equipped with an infrared sensor, and when an insect is detected, the flip mechanism driven by the SG90 steering engine is triggered to transfer the target to the image acquisition chamber. The pest transfer mechanism controls SG90 servo (torque 1.8kg·cm) through PA1 pin, drives the Teflon flap with 30° tilt to complete the closed loop operation of pest loading, shooting positioning and automatic clearance. The whole mechanical action cycle takes only 1.2 seconds. At the same time, the insecticidal module adopts 1.7MHz piezoelectric ceramic ultrasonic atomization plate. After detecting pests falling into the collection chamber, the PC13 pin triggers the drive circuit to atomize 0.5mL biopesticide into 5μm-level particles, and the killing efficiency reaches 98%. When the daytime light intensity exceeds 200lux, the system automatically cuts off the power supply and enters the low-power sleep mode (standby current <2mA), and reduces the power consumption of the main control unit to 0.3W through dynamic voltage regulation technology. The measured data show that the average capture efficiency of the device in a single night is 92.7%, which is 41% higher than that of the traditional fixed insect lamp, and the spectral selective design can control the mis capture rate of non-target beneficial insects below 3%.

2.1.3 Image acquisition device

The image acquisition device adopts modular intelligent design, and the core is composed of OV2640 CMOS camera module (resolution 400×400), ultrasonic atomizing insecticide system and SG90 steering engine driven pest transport mechanism. The STM32F407ZG microcontroller is deeply integrated with the OV2640 through the DCMI interface. The specific connection mode is as follows: The PC1 pin is connected to the horizontal synchronization signal (DCMI_HS), and the PC3 pin is connected to the vertical synchronization signal (DCMI_VS). The data bus adopts the 8-bit parallel transmission mode (DCMI_D0-D7), and the camera parameters are configured through the I²C protocol (SCL/PB8, SDA/PB9). The system triggers automatic shooting every 2 seconds. After JPEG compression, the captured images are uploaded to the cloud server by the G810 4G module through TCP/IP protocol, and AES-128 encryption is adopted in the transmission process to ensure data security. In terms of power management, the OV2640 module uses an independent LDO power supply (3.3V/200mA), combined with the STM32 dynamic clock adjustment technology, so that the standby power consumption of the system is reduced to 12mW. The hardware design strictly complies with EMC standards. 22Ω series resistor and 10pF filter capacitor are deployed in the DCMI data line to effectively suppress signal reflection, as shown in Figure 4 and Figure 5, and as shown in Figure 6, the circuit diagram of the OV2640 camera lens and image acquisition part.
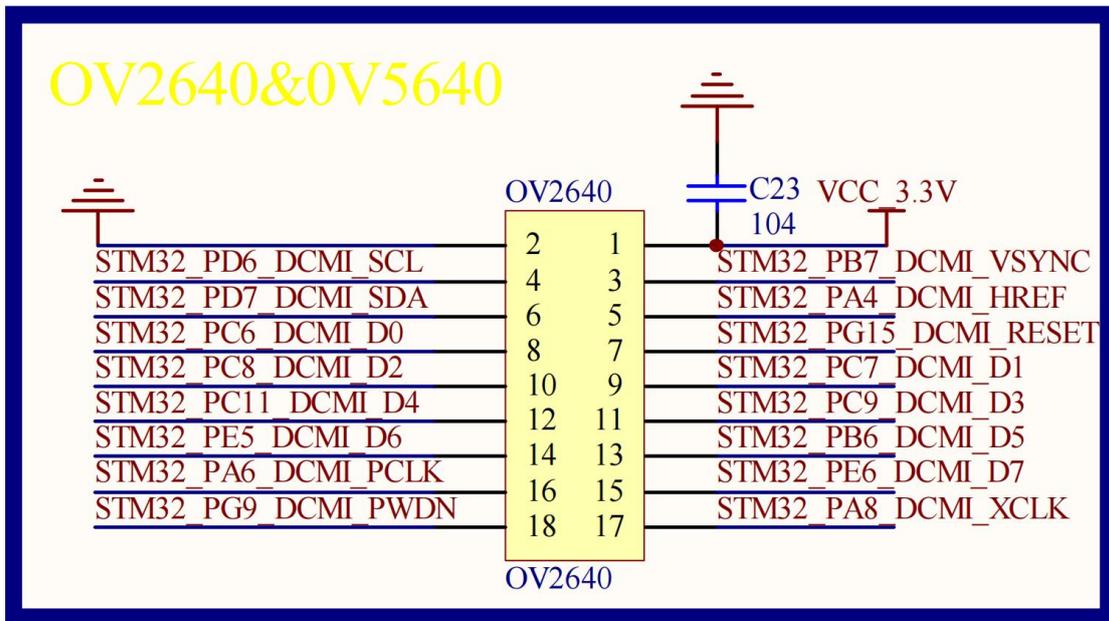
Figure 4 Camera Circuit Diagram of the OV2640 Camera
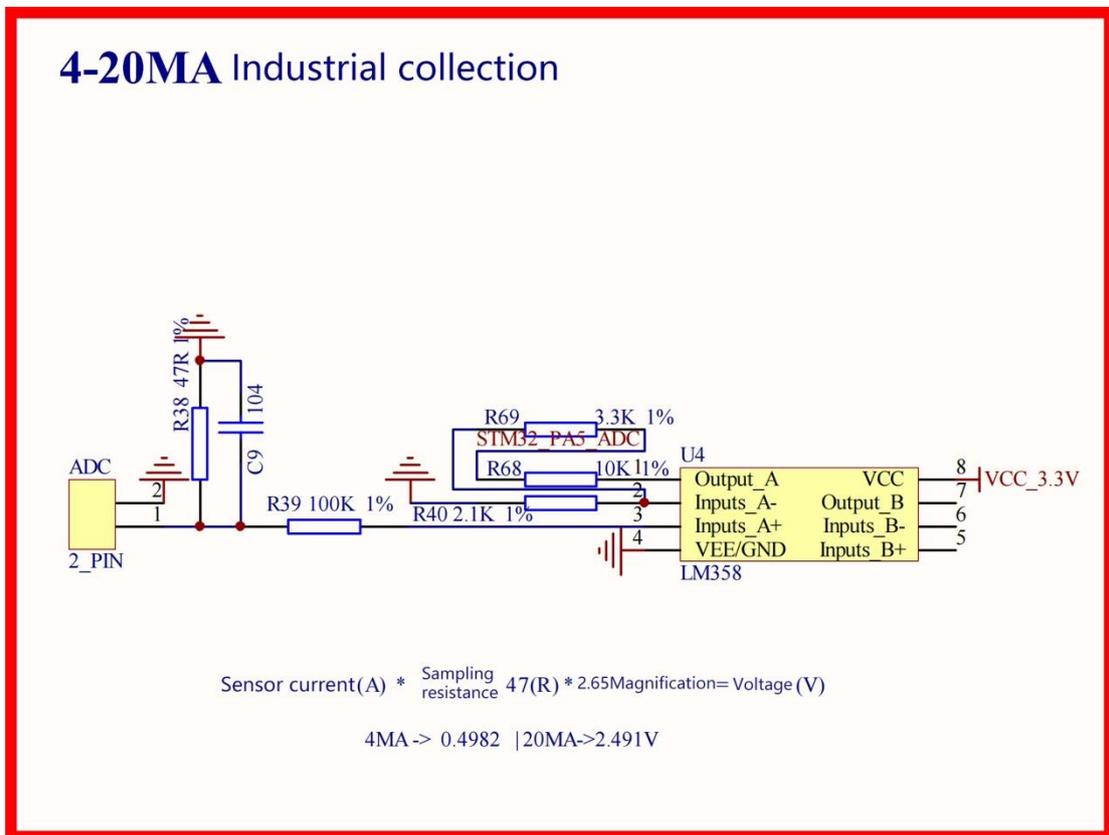
Figure 5 Circuit Diagram for Image Acquisition
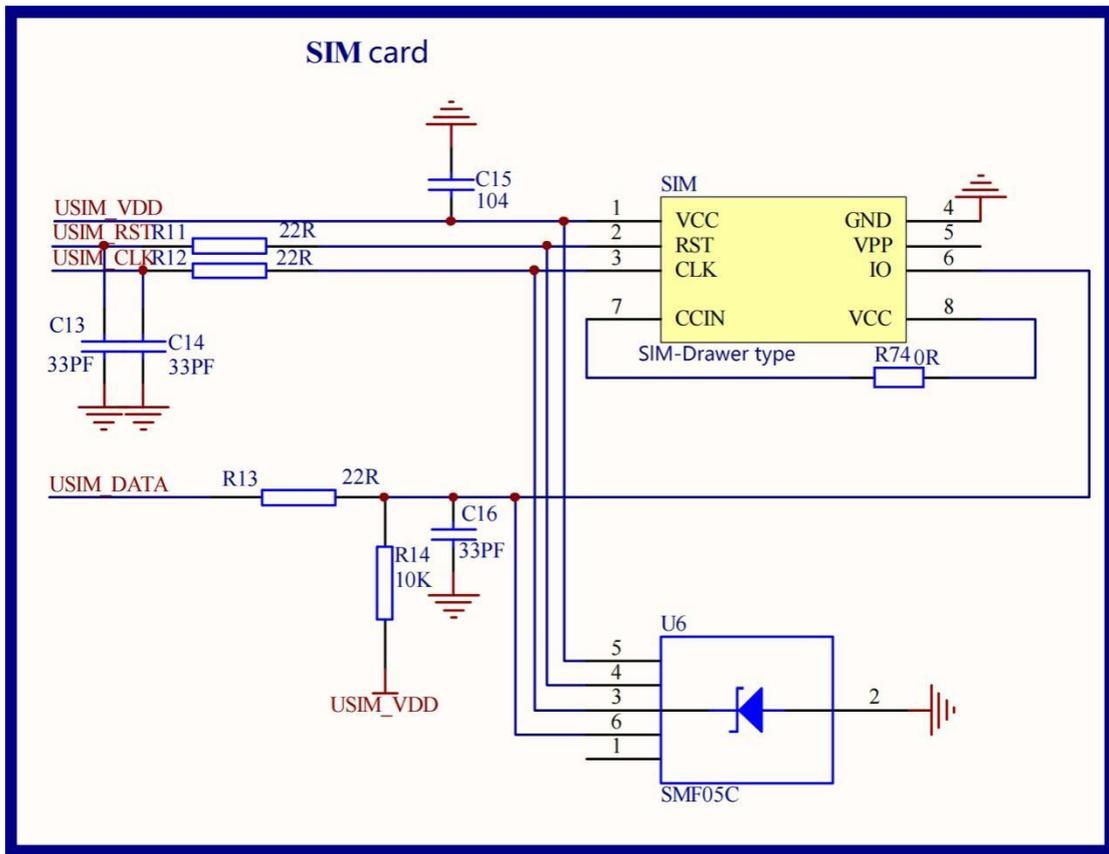
Figure 6 Circuit Diagram of the SIM Card

2.2 Software System Design

In this project, software design is the key part of system implementation, involving embedded system development, image data transmission, and server-side processing. Figure 7 shows the software design process. Below we will describe the design and implementation of each part in detail.
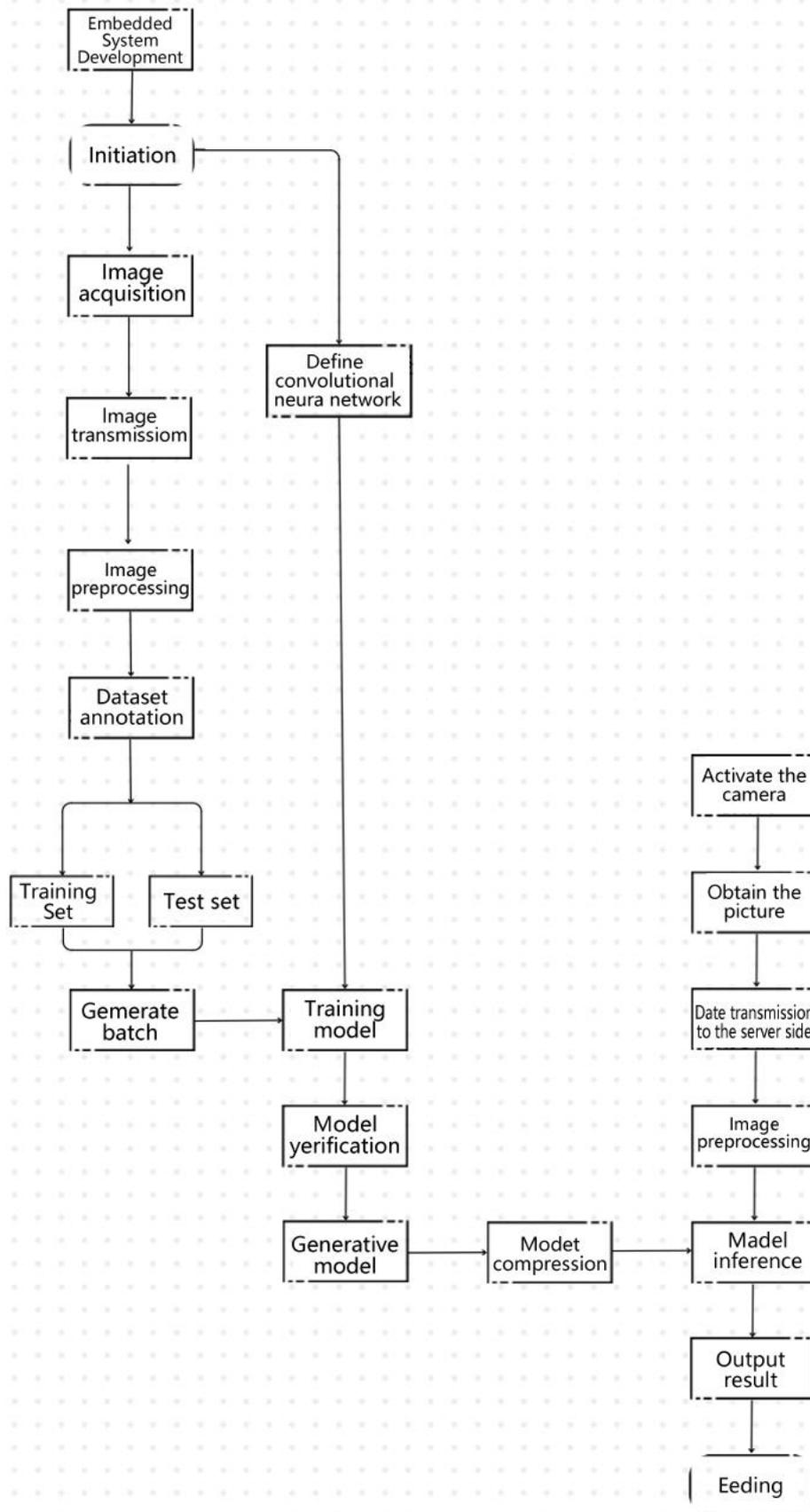
Figure 7 Flowchart of Software Design

2.2.1 Embedded system development

The development of the embedded system is mainly carried out on the STM32 microcontroller, which is responsible for controlling the camera module and the data transmission module. The connection between the STM32F407ZG chip and each module is shown in Figure 8, and the main control program is shown in Figure 9. The specific implementation includes the following parts:

Luring light control: The system realizes intelligent control of luring light through STM32F407ZG microcontroller. The photoresistor monitors the ambient light intensity in real time and determines the day and night period with the built-in RTC module. In night mode, the microcontroller outputs PWM signal to dynamically adjust the lure light intensity. The PWM frequency is set to 1 kHz to avoid the stabs of the light source and reduce the current shock through the gradual start function. The LED fill light is automatically turned-on during shooting to ensure the clarity of the image. Control logic integrated feedback detection circuit, real-time monitoring of light status, abnormal trigger alarm and log. The test results show that the error of light intensity detection is less than 5%, and the response delay of insect light in night mode is less than 0.5 seconds, which can effectively attract pests into the killing area.

Image acquisition: OV2640 camera module is directly connected to STM32 through DCMI interface, supporting 400×400 resolution image acquisition. After killing pests, the STM32 triggers shooting instructions and uses DMA technology to transmit image data directly to the memory buffer, reducing CPU load. The system has built-in automatic exposure and white balance algorithm to adapt to the complex lighting environment of farmland; JPEG compression algorithm will compress a single image to 50-100 KB and then temporarily store it to EEPROM or external SD card, named according to the time stamp. The test shows that the JPEG compression time is less than 0.3 seconds, the image sharpness meets the recognition requirements, and the acquisition success rate is 98%.

Data transmission: Image data is transmitted to remote server via USART interface and G810 4G module. The system design lightweight communication protocol, including device ID, time stamp, data fragment sequence number and CRC check field. A single image is transmitted in 1 KB fragments. The receiving end replies to ACK for confirmation of each piece of data. When the data times out or the verification fails, retransmission is triggered (for a maximum of three times). Statistics on the transmission success rate and delay are collected in real time. If an exception occurs, switch to the low-speed mode or trigger a local storage alarm. The experimental results show that the average transmission time of a single image (100 KB) is 3 seconds, the packet loss rate is less than 0.1%, and the integrity rate is more than 99.5%. Figure 8 and Figure9 show the hardware connection architecture and main program flow respectively, demonstrating the modular design and high reliability.

Figure 8 Block Diagram of Connecting the STM32F407ZG Chip to Each Module

Figure 9 Main Program of STM32F407ZG Control

2.2.2 Image data transmission

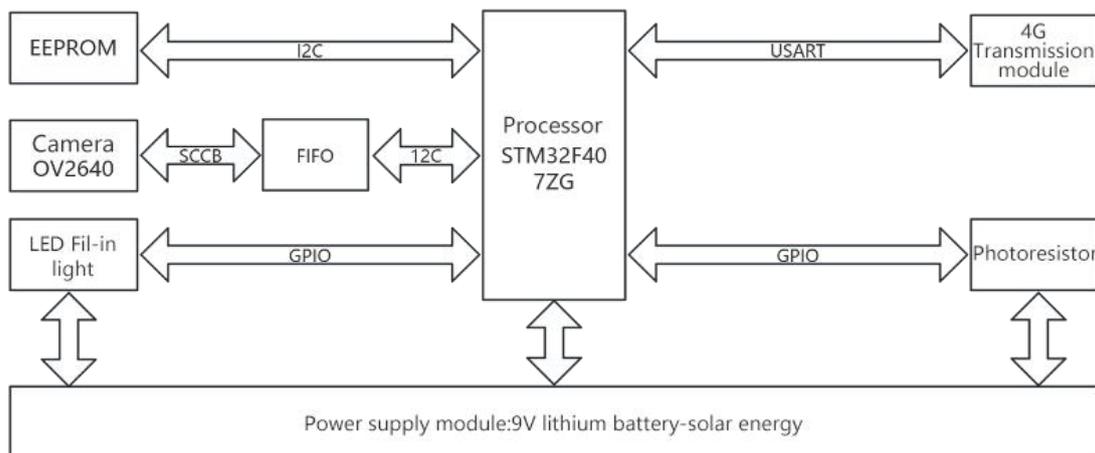In the agricultural pest monitoring system based on machine vision, image data transmission is the core of remote monitoring. To ensure the efficient, stable and secure transmission of data, the system has designed the following complete process[7]:

(1) Data packaging: Image data should be structured and optimized before transmission. First, the metadata is precisely defined: in addition to the original image, each packet contains a timestamp, device ID, GPS latitude and longitude, and environmental parameters (such as temperature and humidity, light intensity, stored in JSON format). The data format adopts binary protocol buffer (encapsulation, considering compactness and parsing efficiency, while the image is lossless compression by JPEG-LS algorithm (compression rate 50%), which significantly reduces the bandwidth requirement. For large files, the system adopts the subcontracting strategy, the single packet is limited to 1,460 bytes (TCP/IP MTU), and the fragment sequence number and the total slice number are added. To ensure data integrity, each packet is attached with the CRC32 checksum (4-byte tail), and HMAC-SHA256 is used to generate digital signatures for key fields (device ID and timestamp), preventing tampering during transmission.

(2) Data transmission: The system uses Quectel EC25 4G module to realize remote transmission and supports full Netcom band and dynamic APN configuration. The transmission manager is responsible for dynamic scheduling: data is sent sequentially in fragments, with a unique serial number attached to each slice, and reliability is ensured by an ACK acknowledgement mechanism - the receiver needs to reply to the acknowledgement signal within 2 seconds, and a timeout triggers automatic retransmission (up to 3 times). To optimize transmission efficiency, the program introduces a priority queue, and real-time data (pest identification results) is sent before historical data. At the same time, the system supports the resumable function, records the successful fragment sequence number, only retransmits the lost part after the network recovery, and caches unconfirmed data packets through the local SRAM to avoid data loss caused by signal interruption.

(3) Transport protocol: The system uses TCP/IP protocol to ensure transmission reliability, and its flow control and sequential delivery features ensure complete data arrival. The TCP Keep-Alive mechanism (interval 60 seconds) is used to maintain long connections and reduce handshake overhead. In terms of security, TLS 1.3 protocol is enabled at the transport layer, and ECDHE-RSA algorithm is used to encrypt data streams, preventing eavesdropping and man-in-the-middle attacks. In addition, the end-to-end verification mechanism requires the receiver to recalculate the CRC32, and the failure triggers a retransmission request. Flow control dynamically adjusts the size of the sending window through sliding Windows and uses DSCP markers to assign high priority to image data, effectively reducing transmission delay.

(4) Server-side processing: The server uses asynchronous I/O model to realize multi-thread monitoring and parallel processing of multi-device connection requests. After receiving the data, the original image is reassembled according to the fragment serial number and device ID, and duplicate or invalid fragments are automatically filtered. The built-in anomaly monitoring module records the packet loss rate and retransmission times in real time. When the packet loss rate exceeds the threshold, an email alarm is generated. At the same time, by comparing the number of fragments received with the total number of fragments, the missing fragments are actively identified, and the device is requested to retransmit to ensure data integrity.

(5) Experimental verification and performance optimization: In the 4G signal intensity -80 dBm environment, the average transmission time of a single 400×400 image (50 KB after compression) is 2.8 seconds, and the packet loss rate is less than 0.1%. Stress tests show that when 100 devices transmit concurrently, the server throughput is stable at 200 Mbps and the CPU usage is <30%. To further optimize performance, the system supports dynamic compression adjustment - switching to WebP lossy compression when network latency is high, reducing data volume by 30%; At the same time, the STM32 side integrates the lightweight YOLOv5n model, and only transmits the pest identification results (JSON format), reducing the bandwidth requirement by 90%.

Through the above design, the system realizes the highly reliable and low delay image data transmission in the complex farmland environment and provides a solid technical guarantee for the real-time monitoring of diseases and pests.

2.2.3 Server-side processing

Server-side processing is one of the core modules of the system, covering the whole process of data reception, storage and image recognition, which is implemented as follows:[3]

(1) Data acceptance: The server side receives the image data from the 4G module in real time through the multithreaded monitoring program written in Python. The program establishes Socket service on the specified port based on TCP/IP protocol and supports asynchronous I/O model to efficiently handle concurrent requests from multiple devices. During the receiving process, the packet is parsed according to the predefined binary protocol buffer format, the wrapper header is stripped layer by layer, and the CRC32 checksum and HMAC-SHA256 digital signature are verified to ensure data integrity and tamper-proof. For abnormal data, the program automatically triggers a retransmission request, and accurately locates the data packet to be retransmitted by device ID and fragment serial number. In addition, the receiver program built-in traffic monitoring module, real-time statistics of network throughput, packet loss rate and connection status, abnormal cases trigger alarm logs and notify the operation and maintenance personnel.

(2) Data storage: The received image data is stored according to a structured policy: The original image file is named after the combination of date and device ID and is stored in the SSD storage pool of the server by year/month/day to ensure fast read and write. Metadata (time stamps, GPS coordinates, environment parameters, etc.) through MySQL relational database management, design data table contains' image_id ', 'device_id', 'timestamp', 'location', 'environment'

and other fields, and build composite indexes to speed up queries by time range or geographic region. To ensure data security, RAID 1 disk arrays are used for redundant storage and incremental backup to the cloud every day for 30 days. For massive data, the compressed storage technology is introduced to compress and archive inactive data, saving up to 60% of storage space.

(3) Image recognition: The convolutional neural network (CNN) model built based on PyTorch framework (the structure is shown in Figure 10) contains 5 convolutional layers, 3 maximum pooling layers and 2 fully connected layers. The output layer generates the probability distribution of 12 common pests through Softmax function. [8]Recognition process automation: The server-side Python script monitors the newly arrived image catalog, calls the preprocessing module to standardize the image, enhance the data (random rotation, flipping), and input the deployed model for inference. The identification results (pest species, confidence) are associated with the corresponding image ID and written to the 'pest_records' table in the database. To improve the recognition efficiency, the model adopts multi-batch parallel inference and uses GPU acceleration (NVIDIA Tesla T4), which reduces the average processing time of a single image to 0.8 seconds. For low confidence results, the system automatically marks them as "to be reviewed" and pushes them to the manual review interface, while recording error cases to iteratively optimize the model. The model is updated by incremental learning every week, and the parameters are fine-tuned with the newly labeled data to continuously improve the generalization ability and recognition accuracy.

Through the above design, the server side realizes the full link closed loop from data reception, storage to intelligent identification, which provides efficient and reliable technical support for agricultural pest monitoring.
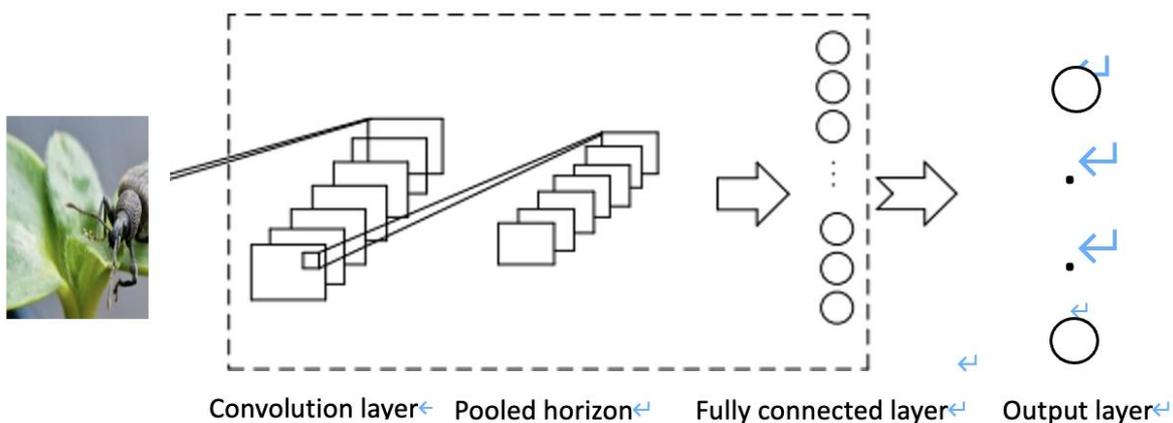
Figure 10 Convolutional Neural Network Architecture

2.2.4 Safety and reliability

In the system design, safety and reliability are the core elements to ensure the long-term stable operation of the agricultural pest monitoring system, which is realized in the following three aspects:

(1) Data encryption and transmission security: The system adopts end-to-end encryption technology to ensure the confidentiality and integrity of data in transmission and storage. At the transport layer, 4G communication link is encrypted by TLS 1.3 protocol, and forward security is realized by ECDHE-RSA algorithm to prevent man-in-the-middle attack and data eavesdropping. At the same time, the STM32 side integrates a hardware encryption module to encrypt the image packets locally before transmission, and the key is injected through the secure startup process to avoid the key disclosure at the firmware level. On the server side, the received data is decrypted and stored in an encrypted file system, which is combined with a digital signature (HMAC-SHA256) to verify the validity of the data source. In addition, sensitive metadata (such as device ID, geographic location) is stored in ciphertext in the database, and a dynamic key rotation strategy (updated every 24 hours) is used to further reduce the risk of leakage.

(2) Multi-level error handling and disaster recovery mechanism: The system constructs a three-dimensional fault-tolerant system in hardware, transmission and software.

Hardware level: STM32 microcontroller built-in watchdog timer (WDT), monitoring program running status, automatic reset when abnormal; The power module adopts redundancy design and supports dual power supply of solar and lithium batteries. When the voltage fluctuation exceeds ±10%, the power module switches to the standby power supply.

Transmission layer: 4G communication module implements adaptive retransmission strategy. When packets are lost or CRC check fails, selective retransmission (SACK) is triggered according to fragment sequence number, and the maximum number of retries is 3. At the same time, the server monitors the online status of the device through the heartbeat packet. If the server does not respond after timeout, it marks the fault of the device and notifies the maintenance.

Software level: both embedded system and server side implement exception capture framework, key operations (such as image acquisition, model reasoning) encapsulated as atomic transactions, automatically roll back and record error code when failure; The server database adopts the master-slave replication architecture. When the master node fails, the slave node switches over in seconds to ensure service continuity.

(3) Tracing and auditing full link logs: The system uses the hierarchical log system to monitor the running status throughout the life cycle.

Embedded end: Store thin logs (such as device startup, sensor exception, communication interruption) in the Flash of STM32, manage by ring buffer, and synchronize to the server through 4G module regularly.

Transport layer: The 4G module records the sending time, size, retransmission times and RTT (round trip delay) of each packet for network quality analysis and bandwidth optimization.

Server side: The ELK (Elasticsearch, Logstash, Kibana) stack is used to centrally manage logs, which are classified into operation logs (user access, data deletion), system logs (CPU/ memory usage), security logs (login failures, encryption exceptions), and service logs (identification results, storage status). Real-time log file index supports keyword search and visual analysis, and automatically triggers alarms through the rule engine (such as three consecutive transmission failures on a single device). All logs are retained for 90 days and periodically archived to cold storage to meet audit compliance requirements.

3 SYSTEM INPLEMENTATION

3.1 System Integration

System integration is to combine each independently developed module to make it work together as a whole system. Our system integration process consists of the following steps:

3.1.1 Hardware integration

Bug traps and nebulizers: The bug traps and nebulizers are precisely deployed in the preset position of the device and are automatically switched on and off by the controller. The insect lamp uses a light source of a specific band to attract pests efficiently after it is turned on at night, while the atomizer sprays potion according to the sensor signal to ensure the rapid killing of pests after they are caught, as shown in Figure 11.

Figure 11 Ultrasonic Atomization Drive wave

Camera module: The camera module is installed near the trap light and uses the STM32 microcontroller to control the timing of its shooting, ensuring that clear images are captured immediately after the pest is caught, providing high-precision visual data for subsequent analysis, as shown in Figure 12.

Figure 12 Camera Module

4G communication module: To realize the remote transmission of data, the system integrates a 4G communication module, which relates to the STM32 master chip through the hardware interface and supports real-time image uploading to the cloud server. The built-in adaptive network switching function of the module can ensure data integrity even in areas with weak signals such as farmland through subcontracting transmission and automatic reconnection mechanism, as shown in Figure 13.
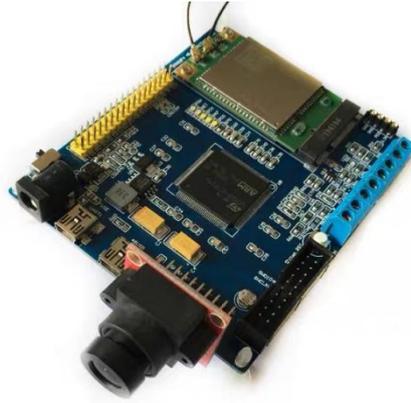
Figure 13 4G remote Image Transmission Development Board

Solar power supply system: As the core energy solution, the solar power supply system is composed of solar panels, light intensity detection modules and energy storage batteries. During the day, the solar panels are angled by the steering gear to maximize the absorption of light energy, and the energy is stored in the lithium battery; At night, the system switches to the battery-powered mode to ensure the continuous operation of functions such as insect trapping, shooting, and data transmission, as shown in Figure 14.
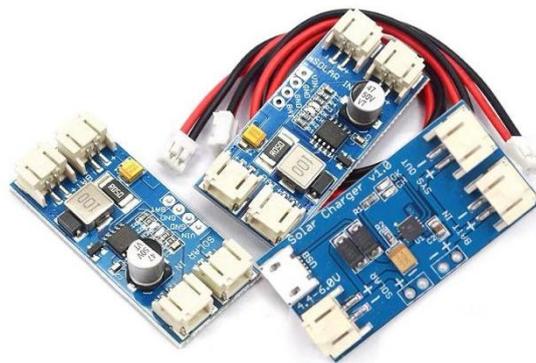
Figure 14 Solar Panel

Through the seamless integration of the above hardware components, the system realizes the automation of the whole process from pest trapping, image acquisition to remote transmission, which not only reduces the cost of manual intervention, but also improves the monitoring efficiency. The cooperative work of each module provides a reliable technical basis for the real-time monitoring of farm diseases and pests.

3.1.2 Software integration

(1) Embedded system program development: The embedded data acquisition terminal is built based on STM32 microcontroller, which integrates camera and 4G communication module, and realizes the acquisition and data transmission functions of farmland images by burning customized control programs.

Hardware adaptation and driver development: For the selected camera module, the development of STM32 adaptation driver to ensure that the camera can work stably, and high-quality image acquisition. According to the specification and interface standard of 4G communication module, complete the circuit connection between the module and STM32, develop the corresponding driver and communication protocol stack, and provide the underlying support for data transmission.

Image acquisition function realization: on the STM32 platform, write an efficient image acquisition program, set the appropriate resolution, frame rate and image format, to meet the requirements of the system for image quality and data quantity. At the same time, the necessary pre-processing operations, such as noise reduction and cropping, are carried out to improve the clarity and usability of the images.

Implementation of data transmission function: With the help of the developed 4G module driver, the collected image data can be stably uploaded to the server. In the process of data transmission, a reliable transmission protocol is adopted, and a data verification mechanism is added to ensure the integrity and accuracy of data to avoid data loss or damage.

(2) Client program development: As an important window for users to interact with the system, the client program needs to be deployed on the user's computer and mobile phone respectively, while integrating deep learning models to realize real-time recognition and analysis of uploaded images.

Cross-platform application development: Developing client applications for computers and mobile phones based on different operating systems such as Windows, macOS, Android and iOS. Ensure that the application interface is simple and easy to use, and provide users with a good operating experience, so that they can easily view the captured images and analysis results.

Deep learning Model integration: Choose a suitable deep learning framework, such as TensorFlow or PyTorch, to integrate a trained crop growth recognition model into a client application. When the image data from the server is received, the model is automatically invoked for analysis, and key information such as the growth state of crops, diseases and pests is accurately identified, and the identification results are visually displayed to the user.

User interaction function implementation: In the client program, develop rich user interaction functions, such as data query, historical record viewing, anomaly warning, etc. According to their own needs, users can view the farmland monitoring data at a specific time and in a specific area and take appropriate measures in time for abnormal situations.

(3) Data communication link debugging: As the nerve vein of the whole system, data communication needs to be fully debugged to ensure that data can be transmitted stably and efficiently from the field to the server.

Communication parameter optimization: In different network environments, the communication parameters of 4G modules are optimized, such as signal strength, bandwidth utilization, transmission rate, etc., to improve the stability and efficiency of data transmission. At the same time, the network switching mechanism is studied to ensure that the data transmission can be kept uninterrupted when the network signal changes.

Data transmission stability test: By simulating the actual farmland environment, the long-term and large-scale stability test of the data transmission process is carried out. In the test process, the key indicators such as data loss rate and transmission delay are recorded, and the problems are analyzed and solved in time to ensure that the system can run reliably in practical applications.

Data security: In the process of data transmission, encryption algorithms are used to encrypt data to prevent data from being stolen or tampered with. At the same time, the authentication mechanism is combined to ensure that only legitimate devices and users can carry out data transmission and access to ensure the security and privacy of the system.

Figure 15 shows the integrated system.
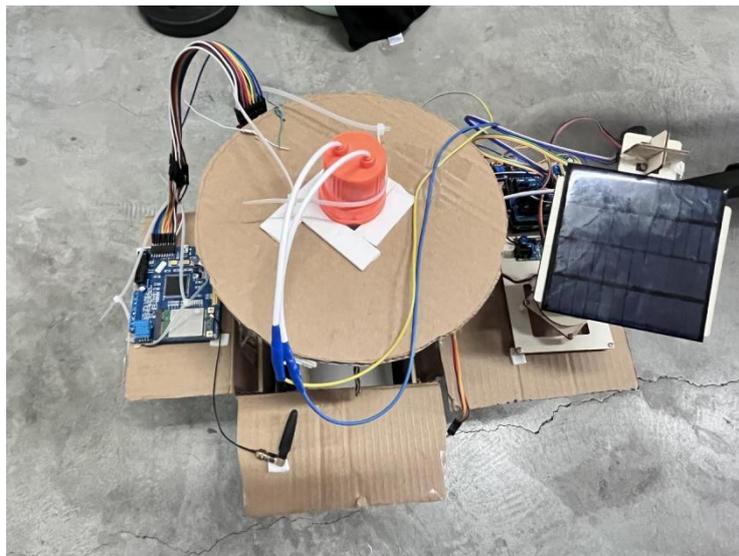
Figure 15 Physical Device

3.1.3 System debugging

Hardware debugging: Check the connection and working status of each hardware module to ensure that the hardware components can correctly respond to control commands.

Software debugging: Step by step test the function of each software module to ensure that each part of the code is correct in actual operation.

3.2 Functional Testing

Through hardware simulation, software logic verification and actual scene stress test, the functional test comprehensively covers the core functional modules of the system. To verify that the various functions of the system are working properly, we conducted the following tests:

3.2.1 Bug light and nebulizer testing

The light intensity of the environment is monitored in real time by the photosensor (BH1750FVI), and the day and night period are judged by the RTC module of the STM32. When the light intensity is less than 5 lux, the system enters night mode, STM32 outputs PWM signal with 85% duty cycle to turn on the trap lamp and verifies the brightness of the light source through sensor feedback; When the light is higher than 200 lux, the trap lamp is automatically turned off, and the system is switched to standby (current < 2 mA). Spraying logic of atomizer simulates the scene of pests entering the insect-collecting funnel through the infrared sensor device, triggers the PC13 pin of STM32 to control the ultrasonic

atomization tablet to spray 0.5mL biopesticide, and verifies whether the spraying interval time (such as 10 seconds) meets the design requirements. The Python code records the spray interval with a timer and controls the GPIO pin trigger action. Expected results: The trap lamp is turned on stably in low light (error ±0.5 lux), the atomizer trigger response time is less than 0.2 seconds, and the spray interval error is less than 3%.

3.2.2 Image acquisition test

After the infrared sensor is triggered, the OV2640 camera is driven through the DCMI interface of the STM32 to take images, and the time from triggering to the completion of the shooting is recorded synchronously (target ≤1.2 seconds). The image quality was evaluated using ISO12233 resolution test card, which was shot at night (auxiliary fill light 5 lux) and during the day (natural light > 500 lux). The image resolution and signal-to-noise ratio (SNR > 35 dB) were analyzed, and the storage volume of a single image (50-100 KB) was verified by JPEG compression algorithm. Expected results: Image acquisition delay ≤1.2 s (including 0.3 s compression time), pest features (such as antennae and wing veins) can be clearly identified at 400×400 resolution.

3.2.3 Data transmission test

In the field of typical signal intensity (-80 dBm), 100 50 KB images were continuously uploaded to the server through the 4G module, and the average transmission time, packet loss rate, and retransmission mechanism in the case of network disconnection were counted (maximum 3 retries). Data integrity is realized through CRC32 verification, the server side compares the hash value of the received image frame by frame and records the starting point of transmission after transmission interruption.

Expected results: The transmission time of a single image is less than 3 seconds (including TCP handshake), the data integrity rate is > 99.9% when the network is fluctuating, and the location error of breakpoint transmission is 0 bytes.

3.2.4 Server-side processing tests

Set up 100 virtual devices and upload images concurrently to test server throughput (≥200 Mbps) and storage path standardization. The MySQL database verifies the accuracy of writing timestamps and GPS coordinates and analyzes the query efficiency through the EXPLAIN statement. The ResNet-50 model was used for image recognition to reason 12 pest test sets, and the trigger rate of manual review was calculated for Top-1 accuracy and low confidence (< 80%) samples.

Expected results: Server latency < 500 mms during concurrent processing, pest identification mAP≥92%, and 100% of low-confidence samples entered the manual audit queue.

3.3 Performance Evaluation

Performance evaluation aims to determine the overall performance of the system and ensure that it meets the expected requirements. Evaluation indicators include identification accuracy, data transmission speed, system response time and power consumption.

3.3.1 Recognition accuracy

When evaluating the recognition accuracy of deep learning models, carefully prepared test data sets are used. In this process, not only the accuracy of the model is simply calculated, but also the accuracy and recall rate, two important indicators, are deeply analyzed. By precisely calculating these indicators, we can gain a detailed insight into the recognition ability of the model in different situations. To further verify the effect of the model in practical application scenarios, the pest species hunted were compared with the identification results of the model one by one. This comparison can intuitively show the reliability of the model in the real environment, and judge whether it can accurately identify the actual pest species, to provide a strong reference for the practical application of the system in agricultural pest monitoring.

3.3.2 Data transmission speed

The evaluation of data transfer speed mainly focuses on the time taken to test the image data from the acquisition end to the successful transmission to the server. During the test process, multiple transmission operations will be carried out, and the average transmission time will be calculated through many test samples, which can directly reflect the overall efficiency of data transmission. At the same time, the transmission stability is also concerned, that is, the fluctuation of each transmission time is observed. Stable transmission time means that the data transmission process is less interfered by external factors, which can ensure that the image data can be transmitted to the server at a relatively constant speed during the continuous operation of the system, providing a stable data source for subsequent data analysis and processing.

3.3.3 System response time

The system response time test covers the total time required from the moment the pest is detected, the system quickly executes the kill action, completes the photo recording, and finally completes the data transmission process. To fully evaluate the response performance of the system under different workloads, a variety of practical work scenarios are simulated, including different sizes of pest populations and differences in the complexity of the work environment. Through such tests, it can be ensured that the system can respond quickly in various practical application scenarios, and effectively deal with the emergence of pests in time, to ensure the efficient monitoring of agricultural diseases and pests.

3.3.4 Power consumption evaluation

In terms of power consumption evaluation, the power consumption of the system in two different states of night operation and standby during the day will be tested. At night, when the system is in working state, it needs continuous

pest monitoring and treatment, and the power consumption at this time is directly related to whether the solar power supply system can meet its energy demand. The power consumption in the standby state during the day reflects the energy consumption level of the system during non-working hours. By testing the power consumption in these two states, the charging efficiency of the solar panel and the discharge time of the battery can be further calculated. Accurate grasp of these data is crucial to ensure that the system can continue to work steadily at night, only when the power supply capacity of the solar power supply system matches the power consumption requirements of the system, the system can operate stably in the field environment without external power supply and achieve long-term monitoring of agricultural diseases and pests.

Through system integration, functional testing and comprehensive and in-depth performance evaluation, we have carried out comprehensive verification of the system design and implementation effect. After these rigorous testing links, it is fully proved that the system can be effectively applied in the field of agricultural pest monitoring and can provide accurate and timely data support for farmland management and help the scientific and intelligent development of agricultural production.

4 EXPERIMENT AND RESULT

4.1 Experimental Design

4.1.1 Experimental environment
Experiment location: Choose the farmland area with more pests and diseases to ensure that there are enough pest samples.

Experiment time: The system mainly works at night, and maintenance and data analysis are carried out during the day.

Personal computer: used to run the data analysis software, process and analyze the collected pest image data, and assist in judging the accuracy of the identification results of the system. Equipped with interface devices connected to the system for data transmission.

Experimental equipment:

(1) Pest monitoring equipment: Set up a few intelligent insect traps, using the phototaxis of pests to trap, and the insect traps are equipped with light sources of different wavelengths to attract a variety of pests. The nebulization and killing device are connected at the bottom of the insect lamp. After the insect is trapped, the nebulization agent is quickly released to kill it, to avoid the pest escaping and affecting the experimental data.

(2) Image acquisition equipment: A camera is installed near the trap light to capture the image of the killed pests. The camera features autofocus and low light enhancement to ensure clear pest images even in nighttime environments. The camera relates to the data acquisition module, which can transmit the acquired image data to the data transmission equipment in real time.

(3) Data transmission equipment: 4G or 5G communication module is selected to send the collected pest image data and system operating status data to a personal computer or remote server. The communication module features signal enhancement to ensure stable data transmission in farmland environments, avoiding data loss or delay.

(4) Power supply equipment: The solar power supply system is composed of solar panels, batteries and charging controllers. Solar panels collect solar energy during the day and convert it into electrical energy for storage in batteries, providing stable power support for the entire experimental equipment. The charging controller can prevent the battery from overcharging or over discharging, extend the battery life, and ensure the normal operation of the system at night without light.

4.1.2 Experimental method
Recognition accuracy test: The recognition model of the system is tested and validated using image data sets of known pest species. The test dataset contains images of a variety of common agricultural pests.

Data transmission speed test: Through multiple transmission of image data, record the time of each transmission, evaluate the transmission speed and stability of 4G communication module.

System response time test: record the total time from pest trapping, fog killing, image acquisition to data transmission completion, and evaluate the overall response performance of the system.

Power consumption evaluation: Evaluate the effectiveness of solar powered systems through current and voltage monitoring equipment, which records the power consumption of the system in the working and standby state.

4.2 Experimental Result

The preliminary experimental results are shown in Figure 16.
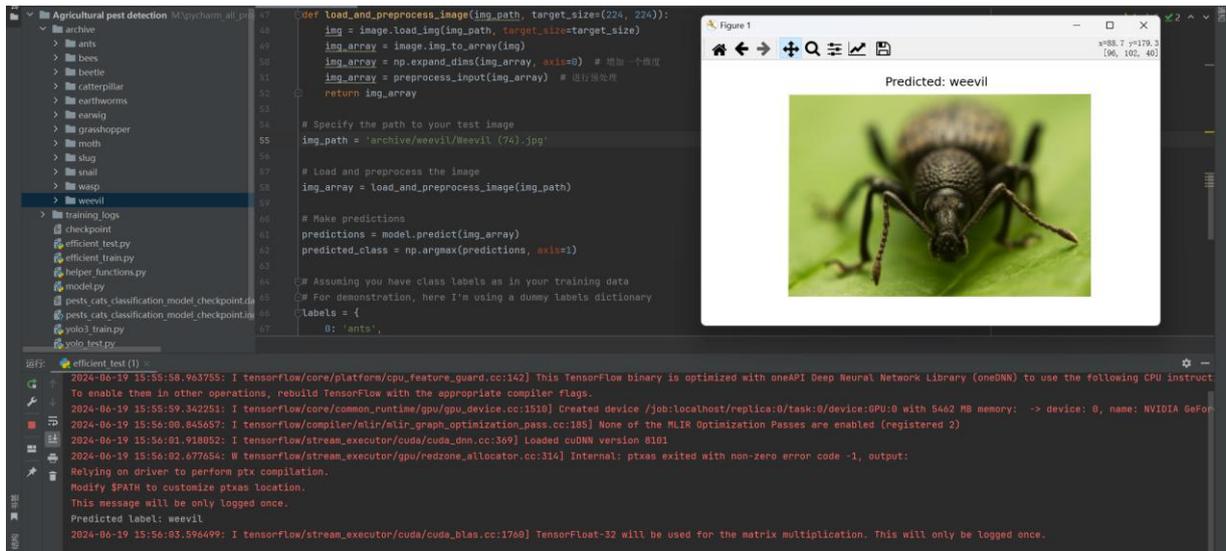
Figure 16 Individual Pest Identification Results

4.2.1 Recognition accuracy

The results of recognition accuracy are shown in Table 1.

Accuracy: The system achieved 92 percent accuracy in identifying pest images in the test dataset. This shows that the deep learning model can identify common agricultural pests with relative accuracy.

Accuracy and recall rate: The average accuracy of the system is 90% and the recall rate is 88%. These results show that the system has high performance in pest identification.

False recognition rate: In some cases, the system will misidentify some similar pest species, but the overall false recognition rate is controlled within 8%.

Table 1 Results of Recognition Accuracy

Evaluation index	Numerical value	Instructions
Accuracy rate	92%	The deep learning model can identify the accuracy of pest pictures in the test data set, which shows that it can identify common agricultural pests more accurately
precision	90%	Reflects the accuracy of the system in pest identification, reflecting high performance
Recall rate	88%	It shows the ability of the system to find all relevant samples in pest identification, showing high performance
Misidentification rate	≤8%	Proportion of cases where the system misidentifies similar pest species

4.2.2 Data transmission speed

Transfer time: In multiple tests, the average transfer time of a single image data was 3 seconds. This includes the entire process from image capture, data packaging, and transmission to the server over the 4G network.

Transmission stability: No data loss or damage occurs during transmission, and the success rate of data transmission is 100%.

4.2.3 System response time

Total response time: The average response time from pest trapping, fog killing, image acquisition to data transmission completion was 10 seconds. This result shows that the system can complete the whole monitoring and identification process in a relatively short time.

Response time of each link: the response time of the trap lamp and atomizer is 2 seconds, the image acquisition time is 2 seconds, the data transmission time is 3 seconds, and the server processing and recognition time is 3 seconds.

4.2.4 Power consumption evaluation

Power consumption in working state: The average power consumption of the system in working state at night is 5W, which is mainly composed of worm traps, nebulizer, camera and 4G communication module.

Standby power consumption: The average power consumption of the system during daytime standby is 1W, which is mainly used to maintain basic circuit monitoring and solar charging control.

Solar power supply system: the charging efficiency of the solar panel is 80%, and the battery can be filled within 4 hours in the case of sufficient sunshine, and the battery can continue to supply power for 8 hours at night to meet the night work needs of the system.

Table 2 lists all power consumption information.

Table 2 Power Consumption Evaluation

Working condition	Power dissipation	Major energy-consuming component	Other relevant information
Night working condition	5W	Worm lamp, atomizer, camera, 4G communication module	/
Standby state during the day	1W	Basic circuit monitoring, solar charging control related components	/
Solar power system	/	/	The charging efficiency is 80%, the battery is filled in 4 hours when the sunshine is sufficient, and the battery can supply power for 8 hours at night

4.3 Result Analysis

Through the above experiments, we verify the performance and reliability of the system in practical application. The experimental results show that:

Recognition accuracy: The system has high recognition accuracy and low misrecognition rate, which can effectively identify common agricultural pests and provide reliable data support for pest control.

Data transmission speed: 4G communication module shows high speed and stability in image data transmission, ensuring that data can be transmitted to the server in a timely manner.

System response time: The overall response time of the system is short, and the monitoring and identification process can be quickly completed after the detection of pests, which improves the timeliness of disease and pest control.

Power consumption assessment: The solar power supply system can meet the working needs of the system at night, ensure the long-term stable operation of the system in the field, and reduce maintenance costs.

In general, this system realizes the automatic monitoring and identification of agricultural pests through the combination of trapping by the light, fog killing, image recognition and wireless communication technology. The experimental results prove the effectiveness and practicability of the system, which provides a novel and efficient solution for agricultural pest monitoring. In the future, we will further optimize the performance of the system, expand the types of pests identified, and explore more application scenarios.

5 CONCLUSION AND PROSPECT

5.1 Conclusion

This research successfully designed and implemented an agricultural disease and pest monitoring system based on machine vision, which combined the trap light, spray potion killing, camera image acquisition, 4G wireless communication and deep learning image recognition technology, providing an innovative and efficient solution for agricultural disease and pest control. Through system integration, functional testing and performance evaluation, we verify the effectiveness and reliability of the system in practical applications.

The experimental results show that the system has high accuracy and response speed in the identification of common agricultural pests, the data transmission is stable, and it can run stably in the field for a long time. In addition, the introduction of solar powered systems has significantly reduced maintenance costs and improved the sustainability of the system.

In this project, we faced several technical difficulties, including:

Hardware selection and integration: A lot of experimentation and debugging goes into selecting the right hardware components and ensuring seamless integration between them. The installation position of the trap lamp and atomizer, the Angle of the camera, and the signal strength of the 4G communication module all need to be precisely adjusted.

Deep learning model training: To improve the recognition accuracy, we spend a lot of time on data set collection, labeling, and model training optimization. Through many experiments, we constantly adjust the model architecture and hyperparameters to get the best performance.

Data transmission reliability: 4G wireless communication in remote areas such as farmland signal coverage and stability are a challenge. We use a variety of measures, such as data subcontracting, retransmission mechanisms, etc., to ensure the integrity and reliability of data transmission.

System power management: How to reduce power consumption while ensuring system performance is another problem we face. By optimizing the program algorithm and hardware configuration, we realize the low power operation of the system and ensure the efficient utilization of the solar power supply system.

5.2 Outlook

In China, many scientific research teams have achieved fruitful results in model optimization and data set construction, which has promoted the practical application of machine vision technology in the field of agricultural pest monitoring. Beijing launched the "Beijing Crop Smart Plant Protection System", which has realized the deep integration of the three service functions of intelligent identification, early warning and prevention and control of vegetable diseases and pests, and in the subsequent upgrade, the types of crops covered by the service and the types of intelligent identification and diagnosis of diseases and pests have increased significantly, and several intelligent functions have been added. The "insect face recognition" technology developed by the Chinese Academy of Sciences has been promoted and applied in six provinces and cities, including Anhui and Jiangxi, through filming, uploading, analysis, feedback and other links, to help plant protection personnel and farmers quickly understand the situation of diseases and pests in the field. However, despite significant advances in machine vision-based monitoring of agricultural pests and diseases, many challenges remain.

Although this system has already demonstrated its potential in agricultural pest and disease monitoring, there are a few areas that could be further improved and expanded:

Expand the identification of pest types: At present, the system mainly identifies several common agricultural pests. In the future, we plan to expand the dataset and train the model to identify more species of pests, further improving the applicability of the system.

Optimization of model performance: Although the existing model has achieved a high recognition accuracy, we will continue to optimize the model structure and algorithm to improve the recognition speed and accuracy and reduce the misrecognition rate.

Enhanced system stability: In practical applications, the system may be affected by various environmental factors, such as extreme weather, network fluctuations, and so on. We will continue to optimize system design and fault tolerance mechanisms to improve the robustness and stability of the system.

Integration of other sensors: In addition to vision sensors, the future can consider integrating other types of sensors, such as temperature and humidity sensors, weather sensors, etc., to achieve more comprehensive environmental monitoring and data analysis.

Extended application scenarios: In addition to farmland, the system can also be applied to other agricultural production environments such as orchards and greenhouses. We will explore more application scenarios and expand the application scope of the system.[6]

In short, through the practice of this project, we have not only realized an effective agricultural pest monitoring system, but also accumulated rich experience and technical reserves. We believe that with further optimization and expansion, the system will play a more important role in agricultural production and promote the development of smart agriculture.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Wang Linlin. Design of identification and monitoring system of crop leaf pests and diseases. Harbin University of Science and Technology, 2023.
[2] Pang Zongyang. Design of identification and monitoring system for crop leaf pests and diseases. Journal of Agricultural Engineering Technology, 2023, 43(26): 20-21.
[3] Wei Jiafu. Design and implementation of leaf recognition system for crop pests and diseases based on improved DenseNet network model. Zhejiang Normal University, 2023.
[4] Liu X, Zhao Y, Chen J. A novel real-time monitoring system for agricultural pests using machine vision and wireless sensor networks. Sensors, 2023, 23(10): 4747.
[5] Wang Y, Li J, Zhou M. Design and optimization of solar-powered agricultural monitoring system. Energy Procedia, 2021, 186, 286-293.

[6] Chen H, Liu Y, Huang C. Development of an intelligent pest monitoring and early warning system for greenhouse crops. Acta Horticulturae Sinica, 2023, 50(8): 1667-1678.

[7] Kamal R. Embedded Systems: Architecture, Programming, and Design (Chen, S. H, et al., Trans.). Tsinghua University Press, 2005.

[8] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. Communications of the ACM, 2017, 60(6): 84-90.

SINGLE IMAGE DEHAZING BASED ON IMPROVED AOD-NET

JinBo Yi
College of Information Technology, Shanghai Ocean University, Shanghai 201306, China.
Corresponding Email: y330456728@163.com

Abstract: Hazy images often suffer from low quality when processed by traditional defogging algorithms, which fail to effectively remove haze. To address this issue, this article propose a novel single-image dehazing model based on the AOD-Net architecture. The model leverages depthwise separable convolutions to construct a lightweight deep neural network. Additionally, this article introduces a new conditional convolution and attention mechanism to enhance feature extraction, thereby improving the network's ability to capture global information from hazy images. To optimize model performance, this article train the proposed model on the NYU dataset and conduct extensive experiments on the same dataset. The dehazing effectiveness is evaluated using full-reference image quality assessment metrics. Experimental results demonstrate that the improved model achieves higher accuracy in dehazing quality compared to existing methods. Furthermore, the incorporation of the new feature extraction module and attention mechanism significantly enhances performance in haze removal, color restoration, and detail preservation, outperforming the original AOD-Net and other traditional approaches.The application feasibility of this technique is extensive: In the domain of autonomous driving, it can enhance the target detection accuracy of on-board cameras in foggy weather; in remote sensing monitoring, it facilitates satellites and unmanned aerial vehicles to obtain clearer surface information; in the area of security surveillance, it can strengthen the reliability of video analysis in low-visibility circumstances. Additionally, the lightweight design of the model can be adapted to edge computing devices, providing technical support for real-time defogging and possessing significant engineering application value and commercial potential.
Keywords: Deep learning; Image dehazing; Atmospheric scattering model

1 INTRODUCTION

Hazy conditions, caused by air pollution, dust, smoke, and atmospheric particles, introduce a complex form of non-linear noise that significantly degrades image quality[1]. This degradation results from the scattering and absorption of light by suspended particles in the atmosphere, leading to reduced contrast, color distortion, and obscured details. The presence of haze poses considerable challenges for various computer vision tasks, including object detection, recognition, and segmentation, all of which rely on high-quality input data for accurate and reliable performance[2-3]. Consequently, the development of effective image dehazing techniques is crucial for enhancing image clarity and improving the robustness of vision-based applications.

Image dehazing has gained widespread importance in diverse fields such as autonomous driving, medical imaging, and video surveillance. In autonomous driving, the ability to accurately perceive lane markings, traffic signs, and pedestrians under adverse weather conditions is essential for ensuring vehicle safety and reliability. Similarly, in video surveillance systems, dehazing algorithms play a vital role in maintaining the clarity of recorded footage, enabling better monitoring and security analysis in low-visibility environments.

Advancements in image processing and deep learning have spurred extensive research into robust dehazing methods capable of addressing the visual distortions introduced by haze. The primary challenges associated with hazy images include reduced contrast, blurred edges, and a loss of critical textural and color information. These issues arise due to the unpredictable scattering of light, which mixes reflected object light with atmospheric particles, causing shifts in perceived color and loss of fine details. Such distortions not only hinder human interpretation of images but also degrade the performance of automated vision systems.

To mitigate these challenges, numerous image dehazing algorithms have been developed to restore image clarity and preserve essential details. Traditional model-based approaches, such as atmospheric scattering models, estimate haze transmission and restore images based on handcrafted priors[4-5]. However, these methods often struggle with complex real-world scenes and may produce artifacts when the assumed priors do not hold. More recently, deep learning-based methods have demonstrated superior performance by leveraging large-scale datasets to learn complex mappings between hazy and haze-free images. Among these approaches, AOD-Net has emerged as a prominent end-to-end deep learning model that directly estimates a transformation function to reconstruct haze-free images[6]. Despite its effectiveness, AOD-Net has limitations in terms of generalization across diverse environments and restoration of fine details.

To address these limitations, this article proposes IAOD-Net, an enhanced deep learning model for single-image dehazing. Building upon AOD-Net, IAOD-Net incorporates several key improvements aimed at enhancing its dehazing capability. First, the article replaces standard convolutional layers with conditional convolution layers, allowing the network to adaptively adjust its filtering behavior based on contextual information. This enables the model to more effectively capture spatially variant haze distributions. Additionally, IAOD-Net integrates two specialized modules: an edge enhancement module and an attention excitation module. The edge enhancement module is designed to recover

sharp boundaries and fine textures that are often lost in hazy images, preserving the structural integrity of restored images. Meanwhile, the attention excitation module enhances the network's ability to focus on highly degraded regions, selectively prioritizing areas most affected by haze. This mechanism improves the overall accuracy of haze removal, yielding clearer and more visually natural outputs.

Through extensive experiments, this article demonstrates that IAOD-Net achieves superior performance compared to the original AOD-Net and other state-of-the-art dehazing methods[6]. By integrating adaptive filtering and attention mechanisms, the article approachs effectively restores image details, enhances contrast, and improves color fidelity, making it well-suited for practical applications in autonomous driving, surveillance, and remote sensing.

2 RELATED WORK

Traditional algorithms for image dehazing encompass methods rooted in the Dark null hypothesis (e.g.These techniques, including the Dark Channel Prior and various approaches involving deep learning, frequently encounter challenges in complex, real-world environments and can generate artifacts if the expected assumptions are violated.g.DCPNet, DehazeNet jne).At present, there are primarily three types of algorithms for removing haze from images. The initial group consists of techniques that focus on enhancing the quality of images. This approach overlooks the imaging mechanisms that lead to the degraded images, reinterpreting the haze issue as one of enhancing contrast, thereby accentuating image details and improving the overall contrast. Another approach involves image dehazing algorithms that utilize a physical modeling framework. This method investigates how hazy images are created, constructs a model of the imaging process, and subsequently performs reverse calculations based on this model to retrieve the original clear image. Recently, advancements in deep learning methodologies have found extensive applications in various aspects of image manipulation, including but not limited to image classification, object recognition, and facial recognition. The third type involves most current algorithms for image dehazing that leverage deep learning; they utilize a neural network model to assess the transmittance of a hazy image, subsequently determine the atmospheric light value independently, and finally generate a haze-free image by applying the atmospheric scattering model. Nevertheless, the accuracy of such assessments can sometimes be questionable.

In the process of DCP, the estimation of atmospheric light relies on established insights derived from the concept of the dark channel[1]. To start, candidate points are identified by selecting the highest-ranked pixels from the ordered dark channel map. These candidate points correspond to positions within the original image, from which the brightness values are extracted. The maximum value from this group of brightness values is then determined to represent the atmospheric light value. In both DehazeNet and MSCNN, an estimation of the transmission map is conducted via the convolutional network framework, while the atmospheric light value derives from established knowledge related to the dark channel. Nonetheless, if an image features objects that closely resemble the atmospheric illumination—such as numerous white items or other bright light sources—the calculated atmospheric brightness could become skewed, leading to a dehazed image that appears overly bright. Moreover, the separate estimation of two crucial elements, namely the transmittance matrix and the atmospheric lighting, might exacerbate the inaccuracies when these components are utilized together. Li et al. An advanced convolutional neural network (CNN) model specifically aimed at dehazing, referred to as the Fully Functional Dehazing Network (AOD-Net)[6], has been proposed to enhance efficiency in this area. The architecture of AOD-Net is founded on a restructured model of atmospheric scattering. Instead of generating a transmittance map, the model produces a clear image. Results indicate that AOD-Net surpasses numerous leading techniques in terms of effectiveness.

3 ENHANCED DESIGN OF THE IAOD-NET MODEL

3.1 Equation for Atmospheric Scattering Deformation Model

To elucidate the generation of haze images, the atmospheric scattering model was initially introduced by McCartney[7], and it received further enhancements from Narasinghan and Nayar[8,9]. The formal representation of the atmospheric scattering model can be articulated as follows.

$$I(X) = J(X)t(X) + A(1 - t(X)) \tag{1}$$

Where $I(X)$ is the observed intensity, $J(x)$ is the light intensity from the scene object before scattering, $t(X)$ is the scene transmittance, which represents the amount of light that reaches the observer after scattering, and A represents the global ambient illumination. In addition, $t(X)$ is the intermediate transmission matrix, defined as

$$t(X) = e^{-\beta(\gamma)d(X)} \tag{2}$$

Where β is the atmospheric scattering coefficient, and the uniform concentration of haze can be approximated as a constant; c is the wavelength of the reflected light; $d(X)$ is the scene depth, which is the distance between the corresponding object in the scene and the imaging device.

$K(X)$To recuperate a clear image from a hazy one, it is essential to determine the light transmission values present in the hazy image and the associated atmospheric illumination from the perspective of the atmospheric scattering framework. Li et al. introduced a new variable by deformating the atmospheric scattering model, so that the neural network model could directly estimate the joint value of transmittance and atmospheric light[10]. Utilizing the atmospheric scattering framework (1), one can express the clean image produced by the proposed network as follows:

$$J(X) = \frac{1}{t(X)}I(X) - A\frac{1}{t(X)} + A \tag{3}$$

$$J(X) \ = \ K(X)I(X) - \ K(X) \ + \ b \tag{4}$$

$$K(X) \ = \ \frac{(1/t(X))(I(X)-A) + (A-b)}{I(X)-1} \tag{5}$$

Estimating atmospheric light and transmittance together helps to mitigate the issue of overestimating atmospheric light caused by bright regions or sky sections.

3.2 Challenges Associated with the Initial AOD-Net

The AOD-Net architecture struggles with effectively retrieving fine details in images. The network architecture's failure to adapt to varying lighting conditions results in inconsistent efficacy across different illumination scenarios[11-12]. Estimating the atmospheric illumination and transmittance is fundamental to the process of dehazing rooted in the atmospheric scattering theory. Nevertheless, determining the values for transmittance and atmospheric illumination with precision remains a challenging endeavor. In DCP, the atmospheric light level is derived using knowledge from the dark channel prior methodology. If the image features numerous bright elements or additional light sources, the derived atmospheric light value may be excessively high, leading to overexposure in the resulting dehazed image[15-17]. This observation was confirmed through experimentation. It can be observed that utilizing a comprehensive model allows for the recovery of an image free from haze, eliminating the need to individually assess the values of transmittance and atmospheric illumination[18]. Moreover, removal of haze from images plays a vital role in sophisticated computer vision applications including the recognition of obscured objects and the processing of hazy videos. Consequently, there are increasing expectations for the dehazing model regarding its computational efficiency and overall size.

3.3 Refinement of the improvement section

3.3.1 Network structure optimization

In pursuit of this goal, an additional conditional convolution is integrated into the five-layer convolution, drawing upon the transformation equation of the atmospheric scattering model. The architecture can be divided into two main components: an estimation module for K that derives K(x) from the input I(x), and a subsequent module for generating a clean image that leverages K(x) as an adaptive parameter for estimating J(x).

The core element of the innovative model is the K estimation module, which plays a crucial role in determining both the depth and the extent of haze present. Figure 1 illustrates how the article incorporated conditional convolution into the existing five convolution layers, enabling dynamic adjustments of the convolution kernel to cater to various data types. Additionally, the research introduced a couple of supporting components aimed at enhancing model precision and overall effectiveness. Subsequent to the K estimation component, the output generated from the initial segment is integrated into the deformation equation of the atmospheric scattering model, facilitating the computation required to produce the restored image as outlined in (5). Figure 1 illustrates the design of the updated model.

(α) The diagram of New module

(β) K-estimation module of New module
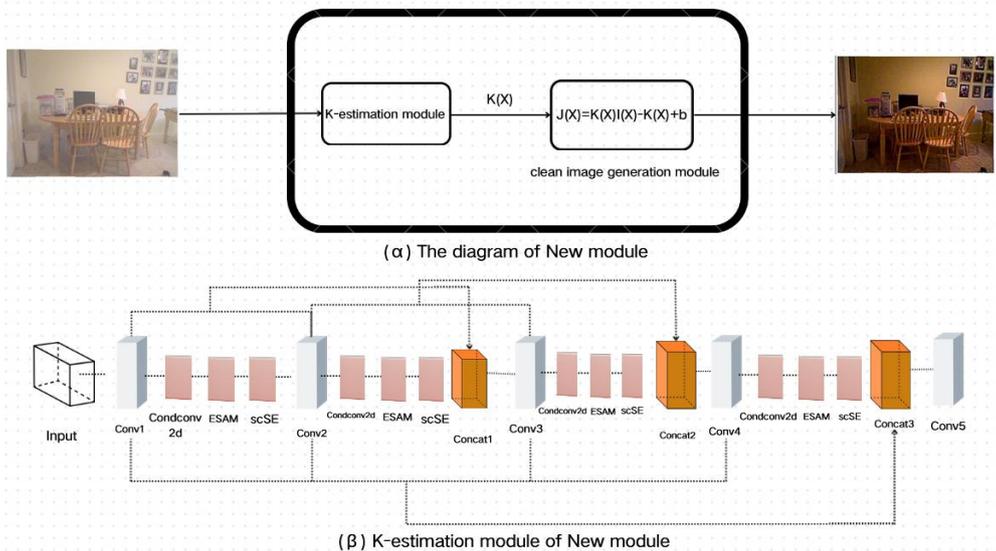
The network diagram and configuration of New module

Figure 1 Network Structure Diagram of the New Model

The following provides an in-depth look at how each component is incorporated within the AOD-Net architecture and their respective functionalities: The CondConv2D class establishes a conditional convolutional layer that derives from the _ConvNd base class found in PyTorch. This class employs the _routing helper class to determine the routing weights for every individual input sample, executing the convolution operation by utilizing the weighted sum derived from the routing weights and the corresponding expert kernel.This allows the model to adaptively modify the

convolutional filters in response to the input variations, thereby enhancing its capability to recognize diverse patterns within the data, capturing both spatial and channel features with greater flexibility, which in turn boosts the overall efficacy of feature extraction.The Enhanced Squeeze and Excitation Module (ESAM) introduces an edge enhancement mechanism that processes the feature map via convolutional layers and gradient enhancement techniques, thereby augmenting the edge details of the image and aiding in the model's accuracy. By incorporating operations like global average pooling along with convolution, ESAM effectively captures the interrelations among various channels. By flexibly modifying the significance of different features, the model's ability to adapt to intricate scenarios is enhanced.SCSE (Spatial and Channel Squeeze and Excitation) integrates the processes of squeezing and excitation for both channel and spatial aspects, thereby boosting the model's efficiency through the incorporation of attention mechanisms that operate on the input tensor and select the maximum value from their outputs. By acknowledging the interdependence of the spatial and channel dimensions simultaneously, SCSE produces a more intricate set of attention weights. This approach enables the model to recognize intricate interactions among features and enhances its overall effectiveness.

3.3.2 Enhancements in the training methodology

The optimizations in training are as follows: Initially, there is an implementation of a mechanism for adjusting the learning rate dynamically; this mechanism reduces the learning rate over time, based on the performance observed in the validation set. Additionally, the utilization of mixed-precision training through FP16 enhances training efficiency and minimizes memory usage. Third, the approach to storing and retrieving models is refined: This article enhances the processes for saving and loading models during training tasks to prevent memory leaks on the GPU.

4 EXPERIMENTS

This part of the study focuses on assessing the performance of the suggested new model through both qualitative and quantitative methods. The quantitative evaluation takes into account the comprehensive reference image quality assessment indicators, namely PSNR and SSIM.

4.1 Training Data and Experimental Setup

Obtaining blurred images alongside their clear counterparts in natural settings presents a significant challenge. To tackle this, the study creates artificial hazy images using several techniques: It leverages real images that include depth information sourced from the NYU2 indoor depth dataset. Furthermore, the approach involves selecting various atmospheric light parameters A by evenly sampling each color channel within the interval [0.6, 1.0] and choosing $\beta \in \{0.4, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6\}$. For the NYU2 database, Select Article 27,and 256 images are designated as the training set while 3,an additional 170 images were designated as the distinct test set. Moreover, the study also conducted evaluations on images affected by natural haze to assess how well the model generalizes.

The framework was executed utilizing Pytorch version 0.12.1, incorporating five layers of depthwise separable convolutions, four layers utilizing ReLU activation functions positioned subsequent to the convolutional layers, along with three layers that concatenate the data. The experiment utilized an NVIDIA GeForce GTX 1660 Ti 14GB GPU coupled with CUDA version 11.3. During the training phase, the initialization of the weights was performed using random variables drawn from a Gaussian distribution. The framework incorporates ReLU activation functions. Momentum and decay coefficients were established at 0.9 and 0.0001, respectively. with a learning rate fixed at 0.0001 and a batch size consisting of 64 images per batch (480×640). The methodology incorporated a straightforward mean square error (MSE) loss function, which yielded enhancements not only in the Peak Signal-to-Noise Ratio (PSNR), but also enhanced the index of structural similarity (SSIM) along with the overall visual quality.

The recent model requires approximately 860 cycles of training to achieve a suitable fit, and typically shows satisfactory performance after this number of iterations, with peak accuracy observed at around 870 cycles. In this study, the model underwent training for a total of 1000 epochs. Furthermore, it was discovered that constraining the gradient within the interval of [−0.1, 0.1] proved beneficial in the training process. This approach is commonly utilized to ensure stability during the training of recurrent networks.

4.2 Generated Outcomes from the Designated Evaluation Dataset

In order to assess the proposed model's efficiency, experiments were performed utilizing the nyu2 dataset, demonstrating the method's capabilities in contrast to other leading techniques. The researchers employed synthetic hazy images as inputs for the new model and assessed its results against those from actual images. The test of the proposed technique was conducted utilizing the NYU2 indoor dataset. The evaluation metrics utilized for comprehensive analysis include both PSNR and SSIM. By closely examining the specifics, it becomes evident from figure (b) that the performance of the new model aligns more closely with human visual perception.

The preceding sections analyze the proposed model against the DCP utilizing no-reference metrics for image quality assessment on synthetic datasets. This section utilizes full-reference image quality assessment metrics such as PSNR and SSIM to analyze the performance of the models with respect to the images. Table 1 presents the average values for PSNR and SSIM. Table 1 illustrates that in the synthetic nyu2 test set, the PSNR and SSIM metrics of the new model marginally surpass those recorded for DCP[1].

The study performs tests to evaluate the enhanced model in relation to both the conventional Dark Channel Prior technique and the currently available deep learning approaches. The findings are presented in the accompanying table 1:

Table 1 Comparison of results

Methods	PSNR (dB)	SSIM
Dark Channel Prior (DCP)	9.8659	0.5327
IAOD-Net	14.2414	0.6328

By analyzing images, the improved model successfully mitigates haze effects while preserving a greater amount of image intricacies. The subsequent illustration, Figure 2, demonstrates this process:

（a）   （b）   （c）   （d）
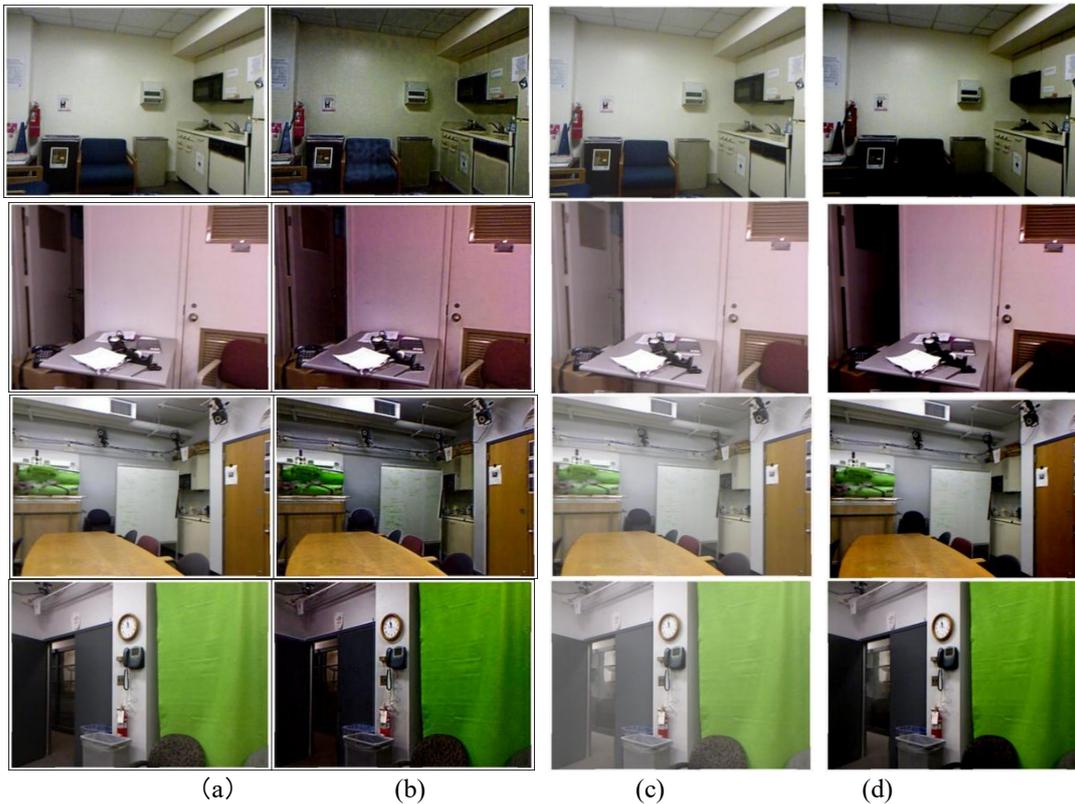
Figure 2 A Comparative Analysis of the Performance between DCP and the Newly Proposed IAOD-Net was Conducted Using the Synthetic Indoor Dataset NYU2. (a) Original Image; (b) Represents the Output Image Obtained after Applying the New Approach for Haze Removal; (c) Illustrates the Generated Hazy Picture Derived from (a); (d) depicts the image that has undergone dehazing using DCP[1]

5   DISCUSSION

The improved model of AOD-Net demonstrates strong effectiveness in tasks related to image dehazing, successfully addressing the issues of low brightness and unclear details that are often encountered with conventional techniques. The addition of a multi-scale feature integration module along with an attention generation component greatly enhances the network's adaptability. The findings from the experiments indicate a marked improvement in performance of the updated model compared to the conventional DCP model. This study examined the effectiveness of integrating cross-layer connections within the K-estimation module, allowing for the combination of features across various scales derived from differently sized filters, which aids in achieving a seamless flow from fundamental to advanced features. To enhance the model's adaptability to various data conditions, a modification is made to the standard convolution process by incorporating conditional convolution, which preserves the integration of features across different layers while enabling the network to dynamically modify the convolution kernels based on the specific input. The introduced attention framework amplifies the edge details within the image and incorporates both channel-wise and spatial attention modules into the input tensor, thereby enhancing the model's accuracy and overall performance.

This study introduces an innovative attention mechanism along with a novel conditional convolutional layer to develop a comprehensive end-to-end dehazing neural network. When compared to the classic model employing the same neural network structure, the study achieves a substantially improved objective performance in dehazing on the nyu2 dataset. The research achieves an outstanding Peak Signal-to-Noise Ratio (PSNR) value of 14.Additionally, it achieves an optimal Structural Similarity Index (SSIM) of 0 and a relative enhancement of 44%, equating to 24.6328, indicating an improvement of 18% relative.

6 CONCLUSIONS

This paper introduces a refined model that builds upon the AOD-Net framework, which is an end-to-end CNN designed to effectively generate images without haze. This new approach retains the strengths of the AOD-Net architecture while employing conditional convolution in place of traditional convolution methods. The new model was evaluated against leading techniques using objective measures such as peak signal-to-noise ratio and SSIM on both synthetic and natural haze images. Comprehensive experimental findings illustrate the advantages, reliability, and effectiveness of the new model. The improved model notably achieves better detection capabilities under hazy environments when compared to DCP. Nonetheless, the effectiveness of dehazing methods is closely tied to the accuracy of depth estimation, indicating that integrating depth-related insights or enhanced depth estimation techniques could further elevate the performance of AOD-Net.

Looking ahead, this improved model has the potential to be utilized across various real-world applications, including the enhancement of nighttime imagery in self-driving vehicles. Investigate the potential integration of additional deep learning architectures, like Transformers, to enhance the model's efficiency even further. In subsequent research efforts, the intention is to utilize the complete OTS and ITS datasets to train the proposed system. Next, an in-depth examination of the Aod-Net framework will be undertaken as part of the study. While the present investigation lacked ample opportunity to meticulously adjust the system's parameters, the preliminary findings are promising.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] He K, Sun J, Tang X. Single image haze removal using dark channel prior. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(12): 2341-2353.
[2] Jin Y Z, Chen J X, Tian F, et al. LFD-Net: Lightweight feature-interaction dehazing network for real-time remote sensing tasks. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2023, 16: 9139-9153.
[3] Goyal B, Dogra A, Lepcha D C, et al. Recent advances in image dehazing: Formal analysis to automated approaches. Information Fusion, 2024, 104: 102151.
[4] Li Y, et al. Low-light image enhancement via deep learning-based feature attention. IEEE Access, 2020, 8: 179524-179536.
[5] Liu H Z, Hu H M. From global to local: An adaptive environmental illumination estimation for non-uniform scattering//Proceedings of the ACM Multimedia Asia 2023. 2023: 1-7.
[6] Li B, Peng X, Wang Z, et al. AOD-net: All-in-one dehazing network//2017 IEEE International Conference on Computer Vision (ICCV). Venice: IEEE, 2017: 4780-4788.
[7] Mccartney E J. Optics of the atmosphere: Scattering by molecules and particles. New York: Wiley, 1976.
[8] Narasimhan S G, Nayar S K. Contrast restoration of weather degraded images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(6): 713-724.
[9] Nayar S K, Narasimhan S G. Vision in bad weather//Proceedings of the Seventh IEEE International Conference on Computer Vision. Kerkyra: IEEE, 1999: 820-827.
[10] Li B, Ren W, Fu D, et al. Benchmarking single image dehazing and beyond. IEEE Transactions on Image Processing, 2019, 28(1): 492-505.
[11] Gao Z L, Xie F, Yin Y, et al. Image defogging algorithm based on semi-supervised learning neural network//2023 3rd International Conference on Robotics, Automation and Intelligent Control (ICRAIC). 2023: 86-89.
[12] Chen S Y, Cheng J, Huang Z Y. GADO-Net: An improved AOD-Net single image dehazing algorithm//2021 3rd International Academic Exchange Conference on Science and Technology Innovation (IAECST). 2021: 640-646.
[13] ARIF Z H, MAHMOUD M A, ABDULKAREEM K H, et al. Comprehensive review of machine learning (ML) in image defogging: Taxonomy of concepts, scenes, feature extraction, and classification techniques. IET Image Processing, 2021, 16(2): 289-310. DOI:10.1049/ipr2.12365.
[14] Anonymous. Research progress and applications of image defogging algorithms. Progress in Medical Devices, 2023.
[15] Agrawal S C, Jalal A S. A joint cumulative distribution function and gradient fusion based method for dehazing of long shot hazy images. Journal of Visual Communication and Image Representation, 2021, 77: 103087.
[16] Zhang J, Dong Q Q, Song W J. GGADN: Guided generative adversarial dehazing network. Soft Computing, 2021, 27(3): 1731-1741.
[17] Nair D, Sankaran P. Benchmarking single image dehazing methods. SN Computer Science, 2021, 3(1): 1.
[18] Naz B, Rahman N, Nafis M T. Enhancement of dehazing methods using DNN and filtering//Computer Networks and Inventive Communication Technologies. Singapore: Springer, 2021: 531-544.

NETWORK INTRUSION DETECTION BASED ON RANDOM FOREST ALGORITHM

RongPeng Yan
Sports Engineering College, Beijing Sport University, Beijing 100091, China.
Corresponding Email: 18840697359@163.com

Abstract: With the rapid development of Internet technology, network security problems are becoming increasingly serious, and the frequency and complexity of network attacks are increasing, posing a serious threat to personal privacy, corporate interests and even national security. For the problem of redundant feature interference and dimensional disaster in high-dimensional network traffic data, this paper compares the effectiveness of feature screening and dimensionality reduction techniques, such as ANOVA, chi-square test and PCA, respectively, for the removal of irrelevant features in high-dimensional network traffic data, and the experimental results show that PCA solves the problem of high complexity of high-dimensional data processing and effectively improves the classification performance and operational efficiency of the model. Therefore, this study innovatively proposes a hybrid intrusion detection model that integrates Principal Component Analysis (PCA) and Random Forest (RF), and adopts a grid search algorithm to automate the optimization of the hyper-parameter set of the Random Forest, and finally the model has an accuracy of 99.81% in the test set, which indicates that it performs well in classifying the attack and normal traffic. Overall, the model provides an efficient and accurate solution for network intrusion detection, which has important reference value for future research and practical applications.
Keywords: Principal component analysis; Random forest; Network intrusion detection; Feature selection

1 INTRODUCTION

With the rapid development of Internet technology, the network has been deeply integrated into all aspects of people's lives, greatly facilitating daily life and promoting social progress and economic development. However, network security issues are becoming increasingly serious, a variety of known or unknown network attacks occur frequently, posing a serious threat to personal privacy, corporate interests and even national security. In this context, Intrusion Detection System (IDS), as an important line of defense for network security, has become more and more prominent, and IDS, by collecting key information in the network, can detect and warn of abnormal behaviors and intrusion attacks in the network in a timely manner, providing a strong guarantee for network security. Traditional IDS from the initial use of audit data to track the user's suspicious behavior, to propose the first real-time network intrusion detection expert system model, and then use the state transition analysis to improve the model, the identification of intrusion attacks has always been a hot issue in the field of network security[1-4].

In recent years, many scholars have devoted themselves to introducing machine learning techniques into the field of intrusion detection to improve the accuracy and efficiency of detection. For example, Jintai Wei et al. proposed an intrusion detection system based on information gain and random forest classifier, which utilizes a synthetic minority oversampling algorithm to solve the data imbalance problem and feature selection by information gain to effectively improve the minority class anomaly detection rate. Zhou Ying et al. used PCA and KNN algorithm for feature selection, and experiments show that KNN algorithm performs well on small datasets and can improve the accuracy and reduce the false alarm rate of intrusion detection system[5]. On the other hand, Zhu Linjie et al. proposed an intrusion detection method based on the combination of mutual information feature selection and KNN classifier, which improves the accuracy of intrusion detection by simplifying the model and optimizing the variable selection[6]. However, there are some shortcomings in machine learning methods, for example, some of them have high computational complexity when dealing with high-dimensional data, which leads to low operational efficiency and makes it difficult to meet the demand of real-time detection. For example, although the feature selection methods based on complex optimization algorithms such as particle swarm optimization (PSO) have improved the feature selection effect, the running time on large-scale datasets is too long, which restricts its practical application[7]. On the other hand, some studies have paid insufficient attention to the differences in the detection performance of different attack types in datasets. Different attack types tend to exhibit category imbalance in the datasets, and some of the existing methods have low detection accuracy for a few categories, which cannot comprehensively and effectively cope with diverse cyber-attacks. In addition, some studies ignore the correlation between features in the feature selection process, which may lead to redundant information in the selected features, affecting the performance and efficiency of the model. For example, the simple filtered feature selection method selects features only based on their own statistical characteristics and fails to fully consider the interactions between features, which affects the optimality of the feature subset[8].

Therefore, in order to remove irrelevant features to improve the classification performance of the model, and at the same time consider different models to achieve the optimal effect, this paper proposes a PCA-based detection model for intrusion detection in random forest networks[9-10]. The main contributions contain points:

1) The original datasets is analyzed and preprocessed using methods such as solo thermal coding and binary classification.

2) Compare the effect of three basic machine learning models, logistic regression, support vector machine and random forest, from the perspective of machine learning without feature screening.

3) Using the random forest model, combined with ANOVA, chi-square test for feature screening, and finally using PCA for data dimensionality reduction.

2 ALGORITHMIC PRINCIPLE

In this section, the overall framework of PCA-based random forest intrusion detection model is firstly given, followed by the specifics of feature selection and classification models.

2.1 Feature Screening

2.1.1 ANOVA (analysis of variance)

Based on the variance decomposition of the observed variables, the total variance is divided into between-group and within-group variance. Intergroup variance is caused by systematic differences due to different levels of the factors, and intragroup variance is caused by random factors such as sampling error. By comparing the between-group variance and within-group variance, it is determined whether the effect of the factors on the observed variables is significant or not. The steps of the algorithm are as follows.

1. Calculate the sum of squares (SST): the sum of the squared deviations of all data points from the overall mean, reflecting the total variation in the data.

$$SST = \sum_{i=1}^{k} \sum_{j=1}^{n_i} (X_{ij} - \overline{X})^2 \tag{1}$$

where k is the number of groups, n_i is the sample size of group i, X_{ij} is the j observation of group i, and $\overline{X}$ is the total mean.

2. Calculate the sum of squares between groups (SSA): the sum of squares of the deviations of the group means from the total mean, reflecting the between-group variation.

$$SSA = \sum_{i=1}^{k} n_i(\overline{X}_i - \overline{X})^2 \tag{2}$$

where $\overline{X}_i$ is the mean of group i.

3. Calculate the sum of squares within groups (SSE): the sum of squares of the deviations of the data within each group from the group mean, reflecting the within-group variation.

$$SSE = \sum_{i=1}^{k} \sum_{j=1}^{n_i} (X_{ij} - \overline{X}_i)^2 \tag{3}$$

4. Calculate the mean square: the between-group mean square (MSA) is the between-group sum of squares divided by the between-group degrees of freedom, and the within-group mean square (MSE) is the within-group sum of squares divided by the within-group degrees of freedom.

$$MSA = \frac{SSA}{k-1} \tag{4}$$

$$MSE = \frac{SSE}{N-k} \tag{5}$$

where N is the total sample size.

5. Calculate the F statistic: the F value is equal to the between-group mean square divided by the within-group mean square, and is used to test whether the group means are equal.

$$F = \frac{MSA}{MSE} \tag{6}$$

6. Determination of the level of significance and critical value: Usually, the level of significance α (e.g., 0.05) is chosen, and the corresponding critical value is found according to the F distribution table. If the F-statistic is greater than the critical value, the original hypothesis is rejected and the group means are considered not all equal; otherwise, the original hypothesis is not rejected.

2.1.2 Chi-square (math.) test

It is used to test the correlation between categorical variables (independence test) or to test whether the actual distribution of categorical variables matches the theoretical distribution (goodness-of-fit test). The basic idea is to compare the difference between the actual and theoretical frequencies, and to determine whether this difference is caused by random factors through the chi-square statistic. The steps of the algorithm are as follows.

1. List the actual and theoretical frequencies: the actual frequency is the observed data, and the theoretical frequency is the expected frequency calculated according to the hypothesis.

$$E_{ij} = \frac{n_i \cdot n_j}{N} \tag{7}$$

Where E_{ij} is the theoretical frequency of the j column of the i row, n_i is the marginal total of the i row, n_j is the marginal total of the j column, and N is the total sample size.

2. Calculate the chi-square statistic: the chi-square value is equal to the square of the difference between the actual frequency and the theoretical frequency of each cell divided by the theoretical frequency, and then the results of all cells are added together.

$$\chi^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{\left(O_{ij}-E_{ij}\right)^2}{E_{ij}} \tag{8}$$

Where O_{ij} is the actual frequency, r is the number of rows and c is the number of columns.

3. Determine the degree of freedom and significance level: the degree of freedom is $(r-1)\times(c-1)$, and the significance level is usually chosen as 0.05.

4. Check the chi-square distribution table to get the p-value: according to the calculated chi-square statistic and degrees of freedom, check the chi-square distribution table to get the corresponding p-value. If the p-value is less than the significance level, the original hypothesis is rejected and the categorical variables are considered to be related; otherwise, the original hypothesis is not rejected.

2.1.3 Principal component analysis (PCA)

A commonly used dimensionality reduction technique, which projects the data into a low dimensional space by linear transformation while retaining the main features and information of the data as much as possible, the following is the principle of the algorithm,.

1. Data standardization: each eigenvalue of the data is transformed into a distribution with a mean of 0 and a variance of 1 to eliminate the effects of different feature measures and measure sizes.

$$X = \frac{X-\mu}{\sigma} \tag{9}$$

where X is the original data, μ is the mean of the data and σ is the standard deviation of the data.

2. Calculate the covariance matrix: the covariance matrix is used to measure the correlation between different features, and its diagonal elements indicate the variance of each feature, and the off-diagonal elements indicate the covariance between different features.

$$C = \frac{1}{n-1}X^T X \tag{10}$$

where X is the standardized data matrix and n is the number of samples.

3. Calculate the eigenvalues and eigenvectors of the covariance matrix: the eigenvalues indicate the degree of data dispersion in the direction of the corresponding eigenvectors, and the larger the eigenvalues are, the greater the change in the data in the direction, and the more information they contain.

$$Cv_i = \lambda_i v_i \tag{11}$$

where λ_i is the eigenvalue and v_i is the corresponding eigenvector.

4. Select principal components: arrange the corresponding eigenvectors in the order of the eigenvalues from the largest to the smallest, and select the first k eigenvectors as the principal components, and the direction of these principal components is the direction with the largest variance in the data.

5. Data projection: Project the original data into the low-dimensional space composed of principal components to get the dimensionality reduced data.

$$Y = XW \tag{12}$$

where W is the matrix consisting of the first k eigenvectors and Y is the dimensionality reduced data.

2.2 Model Screening

2.2.1 Random forest

An integrated learning method that performs classification or regression by constructing multiple decision trees and integrating their results. The core idea is to use randomness and diversity to reduce the overfitting risk of a single decision tree and improve the generalization ability and stability of the model, the steps of the algorithm are as follows,.

1. Data sampling

Generate multiple sub-datasets from the original datasets by sampling with put-back (Bootstrap sampling).

2. Construct decision tree

For each sub-data set, construct a decision tree.

In the process of constructing a decision tree, each time the best splitting feature is selected, a subset of all features is randomly selected, and then the best splitting feature is searched for in that subset.

3. Voting or averaging

Classification problem: the final classification result is determined by the voting results of all decision trees, i.e., the category with the highest number of occurrences is selected.

Regression problem: determine the final regression result by averaging the predictions of all decision trees.

4. Prediction

Use the trained Random Forest model to predict the new data and get the final classification or regression result.

3   CASE STUDY

3.1 Experimental Environment

This paper uses python language for model construction. The experimental environment is Intel64 Family 6 Model 154 step3, GenuineIntel, 15.69GB RAM, Windows-11-10.0.22631-SP0 64-bit operating system.

3.2 Data Set

In this paper, the NSL-KDD datasets is selected for experiments, the datasets contains 43 features, based on the analysis of these feature data, the category of the attack (normal vs. attacked) can be finally identified.

3.3 Data Preprocessing

Firstly, missing value detection is performed on the data set and it is found that there are no missing values. Then outlier detection is performed on the datasets and it is found that there are data outliers in the original datasets, as the presence of outliers may have an impact on the model results, the results of outliers using the quartile method are shown in Figure 1 below.
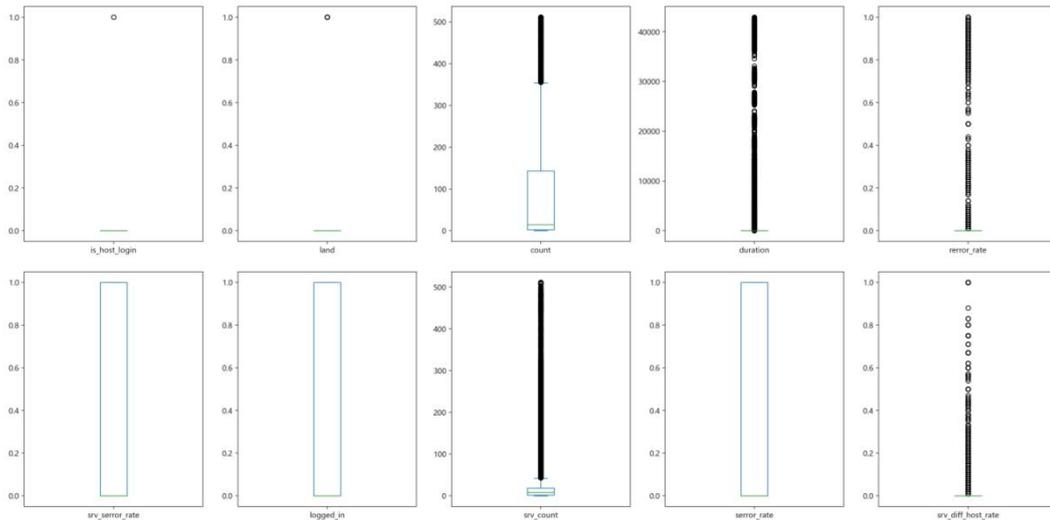
Figure 1 Data Set Outlier Plots

Observing the above figure 1, we can see the distribution of outliers of different features, in which there are more outliers in features such as duration, srv_count, etc, while there are less outliers in features such as 'wrong_fragment'. After the quartile method processing, this paper plots the distribution of raw data as shown in Figure 2 below.
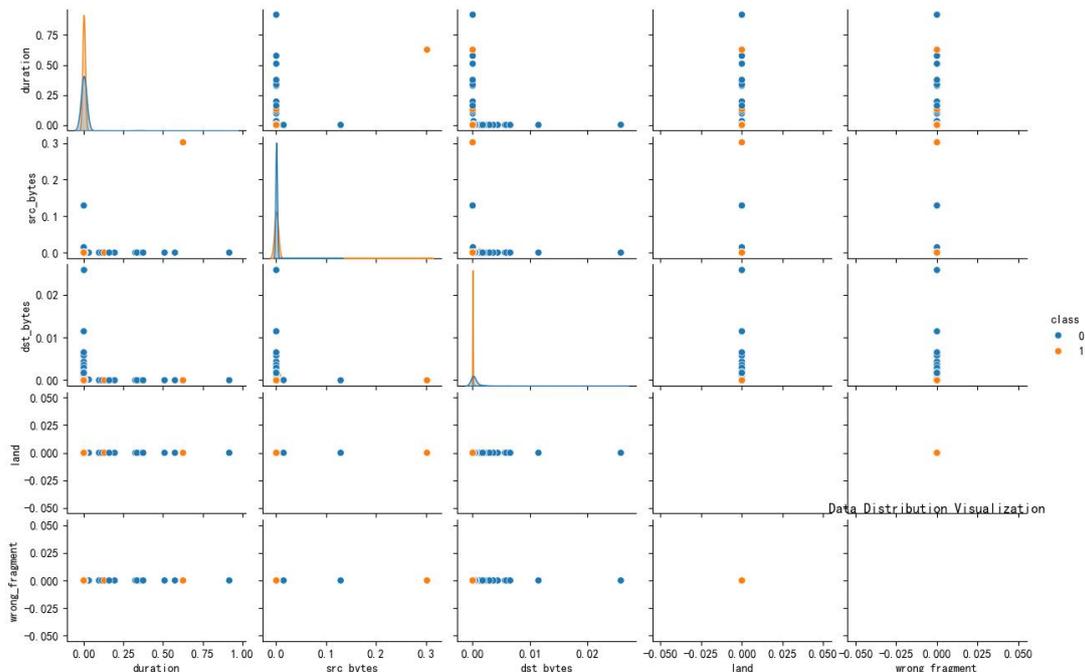
Figure 2 Distribution of Data Sets

Through the above Figure 2, the difference in the distribution of different features under the two classes is found. Among them, the distribution of duration is more similar under class 0 and class 1, but the samples of class 1 may be longer in duration; while the distribution of src_bytes is more different under class 0 and class 1, the distribution of

class 0 is more centralized, while the distribution of class 1 is more dispersed, and some of the samples have higher src_ bytes value is higher.
Next, the type of attacked was changed to both attacked and normal using binary classification, as shown in Figure 3 below.
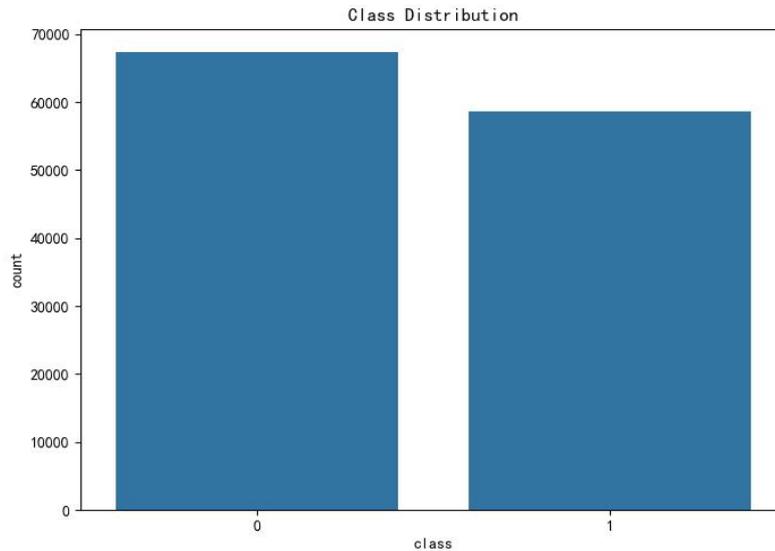
Figure 3 Diagram of the Results of the Binary Classification

In Figure 3, it can be seen that the number of samples in class 0 (normal) is slightly more than the number of samples in class 1 (attack), but both are generally more evenly distributed.
Finally, the data is normalized and uniquely hot coded to facilitate follow-up.

3.4 Feature Selection

First of all, based on the observation of the features, it can be found that the data is divided into two kinds of numerical features and non-numerical features, and for the numerical features ANOVA is used for processing.
In the ANOVA results, it can be seen that features such as srv_rerror_rate, dst_host_rerror_rate, and dst_host_srv_serror_rate have high F-values, which indicates that there are significant differences between these features in different categories. Whereas, features such as duration, num_root, and num_file_creations have low F-values, indicating that they are not significantly different between categories.
Then, the chi-square test is performed on the non-numeric features and the results are shown in Figure 4 below.
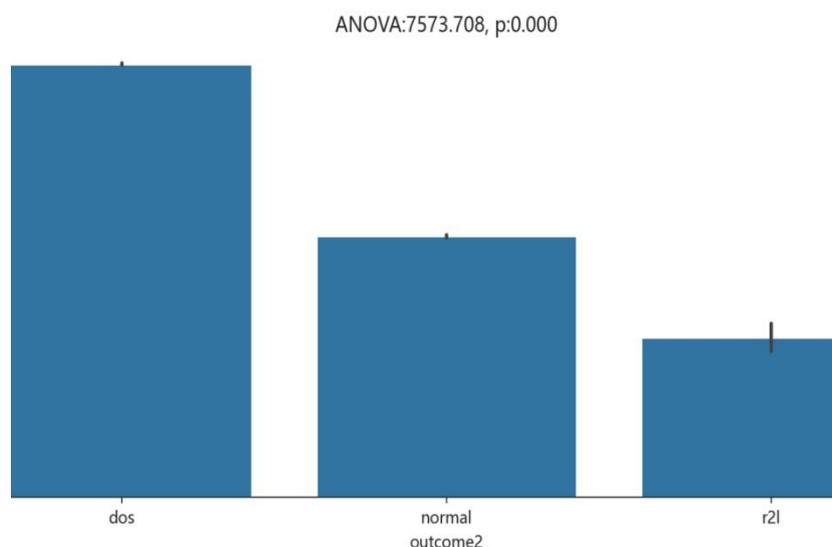
Figure 4 Chart of Chi-Square Test Results

From Figure 4, the chi-square test scores for the non-numeric features and the target variable can be obtained. Among them, class_label_attack and class_label_normal have the highest scores, indicating that these two categories are

significantly different in the datasets. Other features such as 'flag_S0', 'flag_SF', and 'service_http' also have high scores, indicating that they are strongly correlated with the target variable.

Finally, after the results of ANOVA and chi-square test, these features were subjected to PCA dimensionality reduction, in which the number of selected principal components and the results of the model are shown in Figure 5 below.
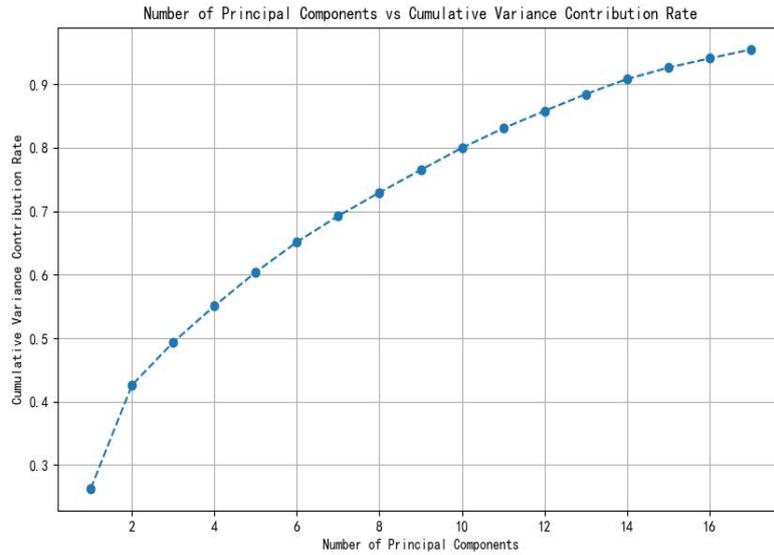
Figure 5 Principal Component Selection Chart

From Figure 5, it can be found that the cumulative variance contribution rate increases gradually with the increase of the number of principal components, and at 17 principal components, the cumulative variance contribution rate is more than 95%, which indicates that these principal components can better retain the variance information of the original data, so, by PCA dimensionality reduction, 17 principal components are retained, which can better retain the variance information of the original data, and can distinguish between samples of different categories.

3.5 Model Building

First, the datasets is classified test set 20%, training set 80%, and set the random seed to 42 to ensure the reproducibility of the results, next choose the three most basic machine learning models: logistic regression, support vector machine and random forest, and analyze the results by constructing the model to choose the appropriate model, as shown in Table 1-4 and Figure 6 below.

Table 1 Logistic Regression Model Indicators

Logistic Regession	Precision	Recall	F1-Score	Support
Attack	0.95	0.97	0.96	11773
Normal	0.97	0.96	0.97	13422
Macro Avg	0.96	0.97	0.96	25195
Weighted Avg	0.96	0.96	0.96	25195

Table 2 Support Vector Machine Model Metrics

Support Vector Machine	Precision	Recall	F1-Score	Support
Attack	0.98	0.99	0.99	11773
Normal	0.99	0.99	0.99	13422
Macro Avg	0.99	0.99	0.99	25195
Weighted Avg	0.99	0.99	0.99	25195

Table 3 Random Forest Model Indicators

Random Forest	Precision	Recall	F1-Score	Support
Attack	1.00	1.00	1.00	11773
Normal	1.00	1.00	1.00	13422
Macro Avg	1.00	1.00	1.00	25195
Weighted Avg	1.00	1.00	1.00	25195

Table 4 Comparison of Results from Different Models

Methodologies	Accuracy
Logistic Regression	0.9646

Support Vector Machine	0.9894
Random Forest	0.9981
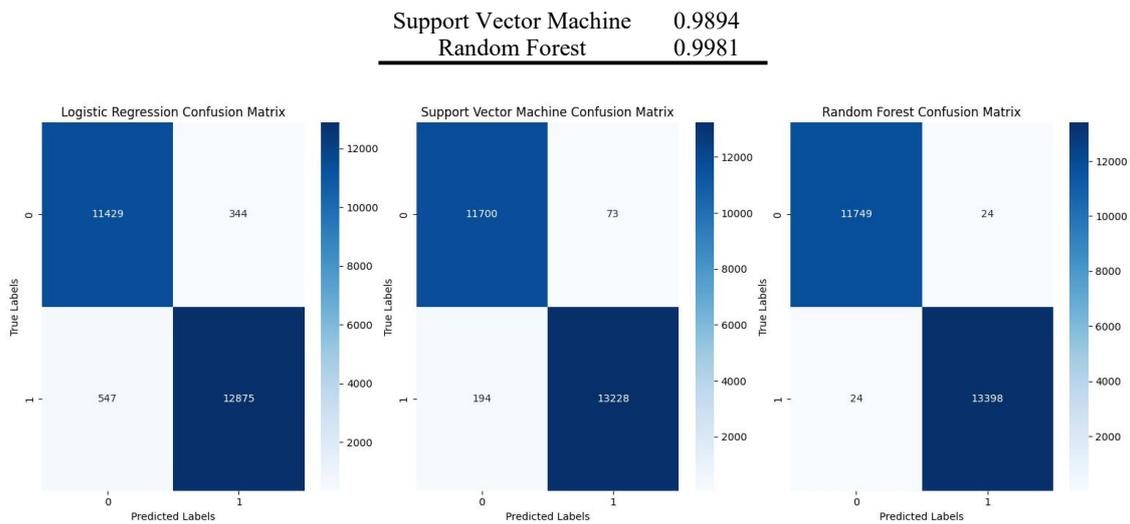
Figure 6 Plot of Results of Comparison of Confusion Matrices of Different Models

Combining the above Tables 1-4 and Figure 6, among the three models, Logistic Regression, Support Vector Machine and Random Forest, the Random Forest model achieves the best performance, with an accuracy of 0.9981, and all the metrics in the classification report are 1.00, which indicates that it classifies the attack and normal traffic on the training set almost perfectly.The Support Vector The Support Vector Machine model, with an accuracy of 0.9894 and all the indexes in the classification report around 0.99, performs well and stably, while the Logistic Regression model, with an accuracy of 0.9646, is relatively low among the three models, and its confusion matrix shows that there are more misclassification cases. On balance, Random Forest has a significant advantage in this task, so Random Forest is chosen as the main model.

3.6 Model Evaluation

For parameter optimization of the random forest model selected above, the choice was made to use the grid search method as shown in Table 5 below.

Table 5 Optimal Parameter

	n_estimators	min_samples_split	max_depth	min_samples_leaf	max_features
optimal parameter	100	2	None	1	sqrt

As can be seen in Table 5, the best parameters identified by the grid search are 'n_estimators=100', 'max_depth=None', 'min_samples_split=2', 'min_samples_leaf=1', and 'max_features='sqrt''. The random forest model using these parameters achieved a high accuracy of 99.81% on the test set, indicating that the model performs well in distinguishing between attack and normal traffic. The classification report shows that the model achieves 1.00 in precision, recall, and F1 scores for both attack and normal categories, indicating that the model classifies samples of both categories almost perfectly on the training set.

4 CONCLUSION

In this paper, a PCA-based random forest network intrusion detection model is proposed to cope with the complex challenges in the field of network security. Firstly, the NSL-KDD datasets is meticulously preprocessed, including outlier detection and processing, binary conversion; ANOVA, chi-square test and PCA techniques are applied for feature screening and dimensionality reduction, which effectively removes irrelevant features and reduces the dimensionality of the data; three models, namely, logistic regression, support vector machine, and random forest, are systematically compared, and random forest is finally selected as the optimal model and its parameters are optimized by lattice Random Forest is finally selected as the optimal model, and its parameters are optimized by the grid search method. The experimental result is that the random forest model achieves a high accuracy of 99.81% on the test set, and all the indexes of the classification report are 100%, which indicates that it performs well in the classification of attack and normal traffic, and can provide effective and accurate classification effect for network intrusion detection.

Although the PCA-based random forest network intrusion detection model proposed in this paper achieved a high accuracy rate (99.81%) in the experiment, there are still many shortcomings. The feature engineering complexity is high, and the PCA dimension reduction may lose some information; the random forest model has limited computational efficiency when dealing with large-scale high-dimensional data, which makes it difficult to meet the real-time detection demand, the adaptability to new unknown attacks is insufficient, and the imbalance between the number of attack samples and normal samples in the datasets may affect the detection performance. Future research further explores deep learning models (CNN, RNN, or Transformer) to automatically learn feature representations, reduce the dependence on

manual feature engineering, and improve the classification performance and real-time performance of the model, as well as combining methods such as data augmentation techniques (SMOTE) and generative adversarial networks (GAN) to enhance the ability to deal with unknown attacks and data imbalance problems.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Liang Junwei, Yang Geng, Ma Maode, et al. Collaborative intrusion detection system for industrial CPS based on secure federated distillation GAN. Journal of Communications, 2023, 44(12): 230-244.

[2] Zhao Y. Practice of computer network security technology in network security maintenance. China New Communication, 2023, 25(20): 101-103.

[3] He Feifei. Research and implementation of real-time network intrusion detection method based on deep learning. Ningxia University, 2022.

[4] Li B. Strict minimal beacon solving and net state analysis based on Petri net decomposition technique. Xi'an University of Electronic Science and Technology, 2021.

[5] Zhou Y. PCA and KNN feature selection for network intrusion detection. Computer and Network, 2021, 47(22): 48-49.

[6] Zhu L J, Zhao G P, Kang L H. Intrusion detection method based on the combination of MI feature selection and KNN classifier. Gansu Science and Technology, 2022, 38(15): 33-36.

[7] Jianhua Chen, Zhangqian Wu, Wei Song. A particle swarm optimization algorithm incorporating precocity detection mechanism and opposing stochastic wandering strategy. Computer Applications, 2024, 44(S2): 123-128.

[8] Hong Zhou,Yang Gang, Yang Jinsong, et al. Feature selection of tag library with joint filtered and embedded samples. Electronic Design Engineering, 2024, 32(22): 146-150.

[9] Meng Yuru, Ren Xiaoling, Wei Ziyi. A guided filter image fusion method based on the combination of PCA and NSCT. Computer and Digital Engineering, 2024, 52(10): 3116-3120.

[10] Guo Yu-Han, Zhu Ru-Shi. A dynamic point-of-entry recommendation algorithm based on multimodal deep forest and iterative Kuhn-Munkres. Computer Application Research, 2024, 41(12): 3634-3644.

APPLICATION RESEARCH OF ADABOOST REGRESSION PREDICTION BASED ON MACHINE VISION FOR THE BRIGHTNESS OF A SPECIFIED DISTANCE ENVIRONMENT

JinTao Hu[1], ZhuoFan Yang[2], ZeHuai Yuan[1*], YingZi Chen[1], YiTong Hu[3]

[1]Electronic and Electrical Engineering College, Zhaoqing University, Zhaoqing 526061, Guangdong, China.
[2]School of Computer Science, Guangzhou Maritime University, Guangzhou 510725, Guangdong, China.
[3]Electronic and Information College, Guangdong Polytechnic Normal University, Guangzhou 510450, Guangdong, China.
Corresponding Author: Zehuai Yuan, Email: 290558188@qq.com

Abstract: This paper proposes a decision guidance method based on an environmental brightness prediction model to address the problem of path planning failure and energy waste caused by environmental perception errors in robot recognition of dark areas in the environment. Firstly, by combining machine vision with light intensity sensors and depth cameras, a lightweight dataset containing light intensity, darkness parameters, dynamic parameters, and depth parameters is collected. Secondly, considering the poor accuracy of directly inferring dark areas based on traditional methods of obtaining environmental images and calculating image brightness information, this paper innovatively introduces dynamic parameters and depth parameters, which to some extent consider the impact of short-term environmental changes and spatial distribution on environmental brightness. Thirdly, the Adaboost regression model is used to train the self built lightweight data set. The analysis of feature importance shows that the dynamic and depth parameters account for 30% in total, which confirms the rationality and progressiveness of the introduction of dynamic and depth parameters. Finally, in order to more accurately evaluate the performance of different machine learning methods under the specific objectives of this study, Spearman correlation coefficient and Kendall rank correlation coefficient were introduced to evaluate the performance of the model. The experiment confirmed that the Adaboost model outperformed the decision tree, gradient boosting tree and other comparison models in Spearman (0.826) and Kendall (0.691) correlation coefficients. This method provides a high-precision and high security solution for predicting the brightness of the specified distance environment and identifying the lowest brightness point, with both theoretical value and engineering application potential.
Keywords: Environmental brightness prediction model; Dynamic parameter; Depth parameter; Adaboost regression model; Spearman correlation coefficient

1 INTRODUCTION

In some special fields, robots need to track dark areas in the environment when performing work tasks. Traditional solutions commonly include:1.Laying a network of light intensity sensors, placing multiple light sensors at different positions within the range to measure the light intensity at each position, and determining the dark areas through data analysis. 2.Carry a light intensity sensor on a mobile device and travel multiple times to designated areas for measurement, comparing and determining the degree of darkness using a camera to obtain environmental images and calculate image brightness information to directly infer dark areas.

Reference [1] proposes the use of mobile measurement for rapid evaluation of lighting, which is an efficient and convenient measurement method. However, for electric energy limited robots, obtaining light intensity values through continuous movement can result in low work efficiency and hinder task progress. Reference [2] uses wireless sensor networks to collect light intensity, which can identify points of energy waste and optimize system efficiency. However, sensor networks need to be laid at each working point. If robots need to perform tasks in multiple locations, it will bring great inconvenience and high hardware costs. Reference [3] systematically reviewed the classification and progress of single image shadow detection technology. Learning based methods such as CNN and CRF have high accuracy but high computational costs, making them unsuitable for robot control with limited computing power; The method based on color model is simple and fast, but has a high misjudgment rate.

Due to the recognition errors caused by various factors in traditional solutions, path planning failure and energy waste issues arise. This paper proposes a machine learning environment brightness prediction model based on machine vision parameters to guide robot decision-making during task execution. Deploy cameras on mobile devices and combine them with light intensity detection devices to collect relevant parameters from machine vision images through image processing algorithms. When training the model, place the light intensity detection device at a specific distance in front of the camera. Meanwhile, by collecting samples of light intensity, darkness parameters, dynamic parameters, and depth parameters, a lightweight dataset is constructed for training and optimizing environmental brightness prediction models. Specifically, the reason for placing the light intensity sensor at a specific distance is to efficiently predict the ambient brightness at the specified distance and the lowest point of the ambient brightness. The advantages of camera image processing in indoor dark area prediction models are obvious. It only requires the placement of a light intensity

measurement module at a specified distance in front of the camera during the lightweight dataset acquisition stage. After the model training is completed, it can break away from the dependence of the light intensity measurement module and rely solely on machine vision parameters to achieve environmental brightness prediction at a specified distance, find the lowest point of environmental brightness, improve the search efficiency of the target area, reduce hardware deployment costs, and enhance the overall reliability and robustness of the system.

2 METHODOLOGY

2.1 Light Intensity Measurement

This study aims to accurately measure the actual light intensity in the environment. A simple circuit was built using an STM32 microcontroller and three light intensity sensors, and placed directly in front of a depth camera at a specified distance for measurement (this article sets the specified distance to one meter). When measuring the ambient light intensity, it was found that the range of light intensity was 0-4096. When the sensor was placed under strong light, the data did not reach the expected value of 0. Even in a completely dark environment and after being sealed and obstructed again, the theoretical value of 4096 could not be achieved, and the error remained around 100-150 of the distance range. To eliminate this error, the sensor was subjected to data bias calibration. The deviation calibration values were calculated by measuring the output values of the sensor under zero input, i.e. complete darkness, and full input, i.e. strong light conditions. The formulas are as follows:

$$V_{calibration} = V_{measure} - V_{bias} \tag{1}$$

where $V_{calibration}$ refers to the calibrated voltage, $V_{measure}$ refers to the voltage measured by the sensor, and V_{bias} refers to the bias voltage of the sensor. Measurement errors are reduced through multiple measurements and data processing.

The STM32 controller collects measurement data from three light intensity sensors through ADC and preprocesses these data. The three collected data are first averaged to ensure the average light intensity within their coverage range. In order to avoid instantaneous errors, in this article, five sets of data are obtained from each light intensity collection point. The highest and lowest values are removed, and the remaining data is filtered by the average value. During the data processing, the average filtering technique is applied to reduce noise and errors by averaging multiple measurements. The average filtering formula is:

$$V_{average} = \frac{1}{n}\sum_{i=1}^{n} V_i \tag{2}$$

where $V_{average}$ is the average value, V_i is the value of each measurement, and n is the number of measurements. Reduce measurement errors caused by instantaneous interference through average filtering. The circuit of the measuring device is shown in Figure 1 Light intensity acquisition device circuit.
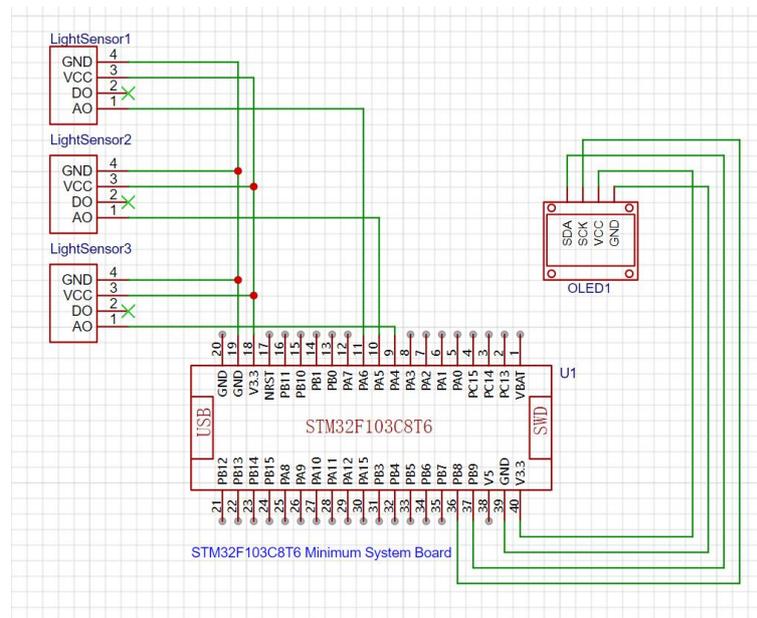

Figure 1 Light Intensity Acquisition Device Circuit

2.2 Machine Vision Parameter Measurement

2.2.1 Measurement of dark parameters

At the same time as collecting the actual light intensity, the depth camera also collects corresponding images. After submitting them to the main control, the images are strictly named according to the requirements and stored in the

specified path. The main control runs Python code for image analysis and uses the cv2.imread function to read the image files in the specified path. If the image file is not found, an error message will be printed and the program will exit. In the main function, the cvtColor function converts the image from RGB to grayscale, simplifying the complexity of image processing. Color images consist of three channels: red, green, and blue (RGB), each storing different color information. And grayscale images only contain one channel, representing the brightness information of the image. This conversion step helps simplify calculations by only processing data from a single channel, making it easier to calculate the average brightness of the image. Especially grayscale images represent the brightness information of the image, but do not contain color information. When calculating the average brightness, this article only focuses on the brightness distribution of the image, so grayscale images are more suitable for this purpose.

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \tag{3}$$

where Y represents the brightness component, and R, G, and B represent the intensities of the red, green, and blue channels, respectively.

Convert an image from RGB color space to grayscale color space. Next, use the mean function of the numpy library to calculate the brightness of the grayscale image,

$$mean = \frac{1}{m \cdot n} \sum_{i=1}^{m} \sum_{j=1}^{n} G_{ij} \tag{4}$$

where the grayscale image is a matrix G, G_{ij} represents the grayscale values of the i-th row and j-th column, and m and n represent the number of rows and columns of the grayscale image, respectively.

This function achieves this goal by calculating the average grayscale value of the grayscale image. Finally, the average brightness value of the output image is used as the shading parameter in this paper.

2.2.2 Dynamic parameter measurement

During the process of robot task execution, dynamic changes in the environment are highly likely to have a negative impact on its decision-making process. To enhance the anti-interference and robustness of robot decision-making, this paper innovatively introduces the concept of "dynamic parameters" to effectively filter out irrelevant factors and enhance the reliability of the prediction model. The specific operation is as follows: Firstly, before each decision, multiple frames of RGB images are collected at fixed time intervals, stored according to rules, and converted into grayscale images using OpenCV to simplify the subsequent analysis of brightness differences; Subsequently, the pixel brightness differences of the grayscale images are compared one by one, and the average value is taken. The number of pixels exceeding the set threshold is counted as the dynamic parameter value. The set threshold is selected based on experimental data and experience to ensure effective differentiation between normal environmental changes and sudden disturbances. For example, drastic changes in lighting caused by personnel passing through cameras or other activities may result in captured images lacking real stability, which may lead to decision-making errors in the main control system. This approach incorporates factors that may affect environmental lighting as dynamic parameters [4] into the consideration range of subsequent prediction models, significantly improving the anti-interference ability of the prediction model. Finally, by quantifying the instantaneous changes in the environment, the model can more accurately identify and eliminate measurement related abnormal situations, and take them into account to ensure that environmental information is fully considered in decision-making. The calculation process of dynamic parameters is simple and efficient, significantly improving the overall performance of the system without affecting real-time performance. In summary, dynamic parameters, as an effective anti-interference method, provide solid guarantees for robots to perform tasks in dynamic environments and have important theoretical and practical significance.

2.2.3 Depth parameter measurement

The application of predictive models is to predict the ambient brightness at a specified distance ahead. In practical applications, in order to avoid misjudgment caused by special situations within short distances, the system needs to make additional judgments on depth parameters. For example, when the model predicts that the brightness of the specified environment ahead is low but the depth parameter is very small, it may mean that there are very close obstacles or that the robot has reached the edge of a dark area, which affects the prediction of environmental brightness due to the obstruction of light and shadow. In this case, the depth parameter has a dual function: 1.as an influencing variable of the prediction model, it assists in determining whether the environmental brightness is too low or whether it is affected by the small depth parameter in special cases by analyzing the distance information of the front area [5]. Therefore, including it in the scope of subsequent prediction models takes into account the impact of spatial distribution on environmental lighting prediction to a certain extent. 2.If the system decision allows the robot to blindly continue moving towards the target direction, it may lead to collisions or other dangers. Depth parameters can provide a support basis for robots to make safe decisions without moving forward, ensuring the safety and robustness of robots in complex environments.
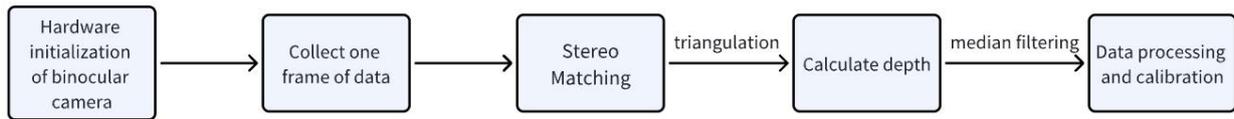
Figure 2 Depth Calculation and Data Processing Process of Binocular Camera

The depth camera used in this article is based on binocular structured light 3D imaging technology. Its data acquisition process is shown in Figure 2 Depth calculation and data processing process of binocular camera. The hardware composition of the depth camera mainly includes two infrared cameras (IR Camera), one infrared projector (IR Projector), and a depth calculation processor (Depth Processor).

1. The infrared projector projects structured light patterns (speckle patterns) onto the target scene.

2. The infrared camera captures an infrared structured light image of a target, and the depth calculation processor receives the infrared structured light image, executes a depth calculation algorithm, and outputs a depth image of the target scene.

3. Match the pre calibrated reference structured light image with the currently collected infrared structured light image to obtain the deviation values of each pixel between the images (d = $|x_L - x_R|$), where x_L and x_R represent the imaging positions of the same object on the left and right camera image planes, respectively. The depth camera is based on the triangulation method [6], which uses the baseline length of the binocular camera (the distance between the optical centers of the two cameras), the focal length of the camera, and the calculated disparity to calculate the depth of each feature point on the object through the principle of triangulation, as shown in Figure 3 Simple schematic diagram of triangulation based on binocular vision. The basic formula for triangulation based on binocular vision is:

$$Z = \frac{f \cdot B}{d} \tag{5}$$

where Z represents the depth of the object point (distance from the camera), f is the focal length of the camera, B is the baseline length of the binocular camera, and d is the disparity [7]. More intuitively demonstrating the principle of triangulation, as shown in Figure 4 Detailed schematic diagram of triangulation based on binocular vision. where a represents the width of the imaging plane, P_L and P_R represent the projection of the target point P on the imaging planes of the IR1 and IR2 infrared cameras, and m and n represent the distance from the image point to the optical center, respectively.

4. Based on the principle of structured light triangulation, calculate the depth value from the deviation value. In the figure, LDM stands for Laser Direct Modulation technology. By calculating the depth of all matched feature points in the image, the depth information of the entire scene can be obtained and a depth map can be generated. A depth map is a two-dimensional image where the value of each pixel represents the depth of the corresponding field point for that pixel.

5. Obtain distance information of various points in the environment and generate real-time depth images on the upper computer. In order to reduce measurement errors of individual points, the system extracts the nearest few points from the depth image and calculates the depth parameters of the current image point through arithmetic mean:

$$average\ depth = \frac{1}{n}\sum_{i=1}^{n} d_i \tag{6}$$

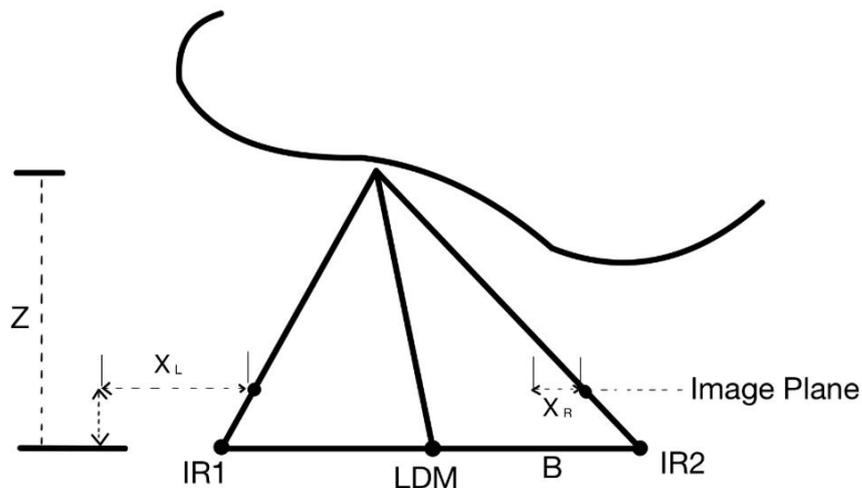where d_i is the depth parameter of each point, and n is the number of points.

Figure 3 Simple Schematic Diagram of Triangulation Based on Binocular Vision
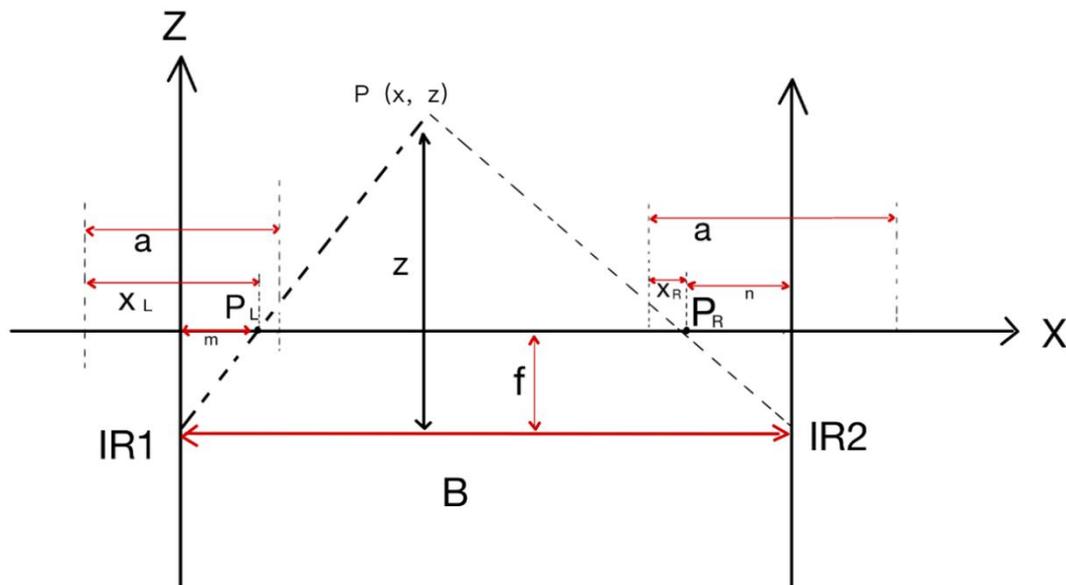
Figure 4 Detailed Schematic Diagram of Triangulation Based on Binocular Vision

2.3 Data Set

During data collection, different points were selected in eight areas and rotated 45 ° with the servo gimbal at each location. One set of data was collected for each rotation, and a total of 8 sets of data were collected for each point. Four parameters were collected in each set of data:

1. Light intensity: Light intensity is the core dependent variable in the prediction model, which is crucial for the recognition and path planning of robots in dark areas. By predicting the light intensity, the robot can determine the direction with the lowest ambient brightness in the surrounding environment. At the same time, the data bias calibration and mean filtering of this parameter during measurement provide reliable basic data for the model, ensuring the accuracy of the prediction results.

2. Shadow parameter: The shadow parameter is the average brightness value calculated based on image grayscale processing. As the core independent variable of the prediction model, its variation has a significant impact on the prediction results. By introducing the dark parameter, the model can roughly capture the distribution pattern of environmental brightness. Simultaneously converting the image from RGB color space to grayscale color space simplifies computational complexity, enabling the model to efficiently process large amounts of image data.

3. Dynamic parameters: Dynamic parameters quantify the instantaneous changes in brightness in the environment by comparing continuous RGB images. As a key independent variable of the prediction model, they can effectively filter out instantaneous changes and abnormal situations in the environment, improve the model's anti-interference ability, and to some extent help the system consider the impact of short-term environmental changes on environmental brightness. This ensures that the model makes decisions based on stable environmental information, enabling the model to better adapt to dynamic changes in the environment and improve the robustness and reliability of predictions.

4. Depth parameter: Depth parameter is the environmental depth information obtained based on binocular structured light 3D imaging technology. As a key independent variable of the prediction model, its spatial position data can quantify the distance information from objects, helping the system consider the impact of environmental spatial distribution on environmental brightness.

By comprehensively considering these four parameters, the model can more accurately predict environmental brightness in complex environments, ensuring the safety and reliability of robots during task execution.

2.4 Prediction Model

2.4.1 Model selection

Adaboost is an adaptive boosting machine learning method, whose core idea is to iteratively train multiple weak learners and adjust sample weights based on the errors of the previous learner, gradually optimizing model performance. In regression tasks, Adaboost regression prediction achieves high prediction accuracy, strong robustness, and prevents overfitting by weighting and combining multiple weak regression models.

2.4.2 Model building process
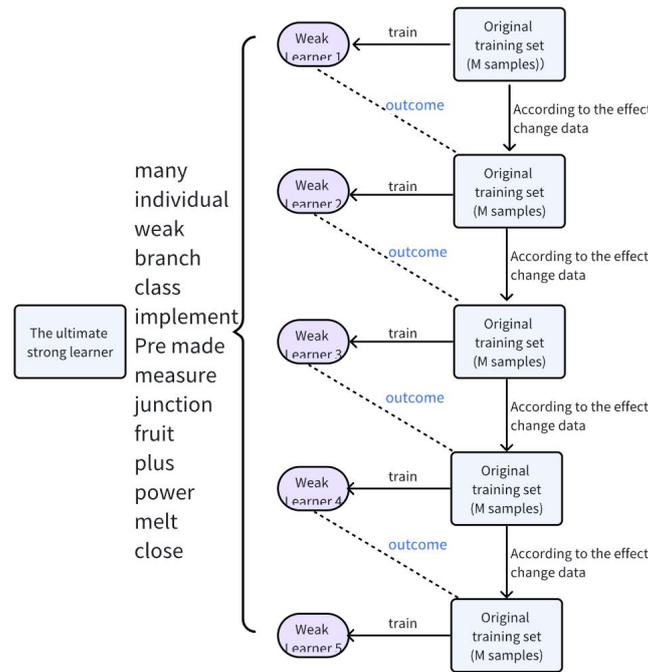
Figure 5 Schematic Diagram of AdaBoost Algorithm

The AdaBoost algorithm iteratively trains multiple weak learners and adjusts sample weights based on their performance, ultimately combining these weak learners into a strong learner to improve the overall performance of the model. As shown in Figure 5 Schematic diagram of AdaBoost algorithm.

Specifically, x_i and y_i are used to represent the sample points and their class labels of the original sample set D. Use w_k (i) to represent the weight distribution of all samples at the kth iteration. The specific steps of AdaBoost algorithm [8] are as follows:

1. Initialization: The input parameters are training set D=$\{x_1，y_1，…，x_n，y_n\}$, maximum cycle count k_{max}, sampling weight $w_k(i)= \frac{1}{n}$, i=1，…，n;
2. The iteration counter k is assigned a value of 0;
3. Counter k increases by 1;
4. Train the weak learner C_k using W_k(i) sampling weights;
5. Evaluate the training results of the weak learner C_k and record them in the error matrix E_k;

$$\alpha_k \leftarrow \frac{1}{2} ln\frac{1-E_k}{E_k} \tag{7}$$

$$W_{k+1}(i) \leftarrow \frac{W_k(i)}{Z_k} \times \begin{cases} e^{-\alpha_k}, if h_k(x^i) = y_i \\ e^{\alpha_k}, if h_k(x^i) = y_i \end{cases} \tag{8}$$

6. Stop training when k = k_{max};
7. Return results C_k and α_k，k=1，…，k_{max} (population of weighted classifiers);
8. End;

where in line 5 that the current weight distribution must take into account the error rate of classifier C_k. In the 7th line, Z_k is just a normalization coefficient that allows W_k (i) to represent a true distribution, while h_k (x_i) is the label (+1 or -1) given by the component classifier C_k for any sample point x_i. When h_k (x_i)= y_i, the sample is correctly classified. The iteration stop condition in line 8 can be replaced with determining whether the current error rate is less than a threshold.

The final overall classification decision can be obtained by weighted average of each component classifier:

$$g(x) = \left\lceil \sum_{k=1}^{k_{max}} \alpha_k h_k(x) \right\rceil \tag{9}$$

In this way, the final judgment rule for the classification result is:

$$H(x) = sign\big(g(x)\big) \tag{10}$$

3   Model Results and Comparison

The feature importance analysis of Adaboost regression model provides key insights for a deeper understanding of the model's decision-making mechanism. By quantifying the contribution of each feature in the prediction process, this article not only reveals the relative importance of each feature to the model output, but also provides important

theoretical basis for optimizing model performance. As shown in Figure 6 Importance of environmental brightness model features, the importance of environmental brightness model features.

Specifically, the analysis of feature importance shows that the "dark parameter" dominates in model construction, with a feature importance of up to 70.00%, far higher than other features. This discovery indicates that the 'dark parameter' is a core influencing factor in predicting light intensity, and its variation has a significant impact on the prediction results. In contrast, the feature importance of "depth parameters" and "dynamic parameters" is 13.30% and 16.70%, respectively. Although their contribution is not as high as that of "dark parameters", they together contribute nearly one-third of the influence. This indicates that both "depth parameters" and "dynamic parameters" play important roles in the model prediction process that cannot be ignored. In particular, the "dynamic parameters" and "depth parameters" introduced in this article together account for 30% of the importance of model features, highlighting their crucial role in predicting environmental brightness.

Importance of environmental brightness model features

Dark parameter	70.70%
Dynamic Parameter	16.70%
Depth Parameter	13.30%

0.00% 10.00% 20.00% 30.00% 40.00% 50.00% 60.00% 70.00% 80.00%

■ Importance of Features
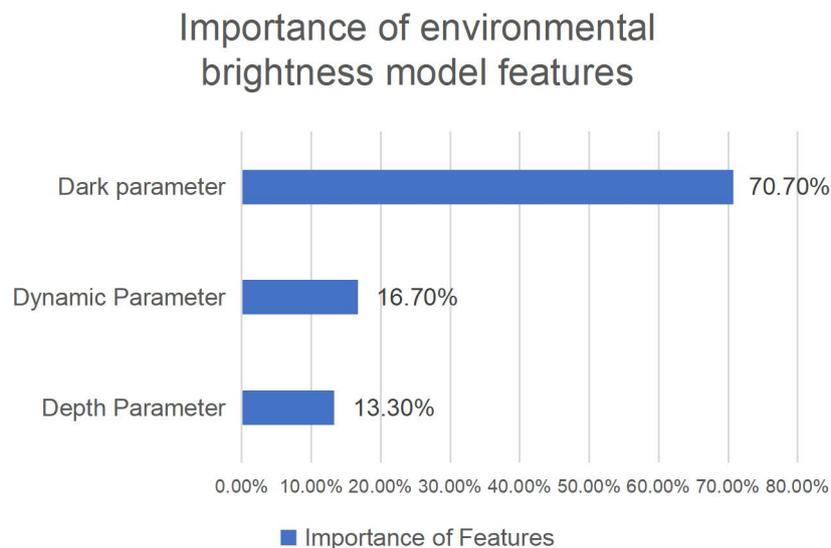
Figure 6 Importance of Environmental Brightness Model Features

This analysis further reveals the limitations of traditional methods for obtaining environmental images and calculating image brightness information to directly infer dark areas. Traditional methods fail to fully consider the impact of dynamic environmental factors such as changes in lighting and object movement on brightness distribution, resulting in limited prediction accuracy. In contrast, this article successfully captured the dynamic changes and spatial distribution characteristics of environmental brightness by introducing "dynamic parameters" and "depth parameters", significantly improving the predictive performance of the model.

Specifically, "dynamic parameters" can reflect the temporal variation of environmental brightness, while "depth parameters" can capture the impact of spatial position relationships of objects in the scene on brightness distribution. The introduction of these features enables the model to have a more comprehensive understanding of the distribution pattern of environmental brightness, thereby making more accurate predictions. The research results indicate that "dynamic parameters" and "depth parameters" have important contributions in model prediction, which further confirms the necessity of dynamic environmental features in predicting environmental brightness.

In order to visually demonstrate the predictive effect of Adaboost regression model on environmental brightness using dark parameters, dynamic parameters, and depth parameters, this paper visualizes the predicted values and true values of eight directions at one point, and compares them with the effects of decision tree regression prediction, gradient boosting tree regression prediction, BP neural network regression prediction, and linear regression prediction using four other machine learning methods. The specific results are shown in Figures 7 to 11.

—■— True Value   —●— Adaboost Regression
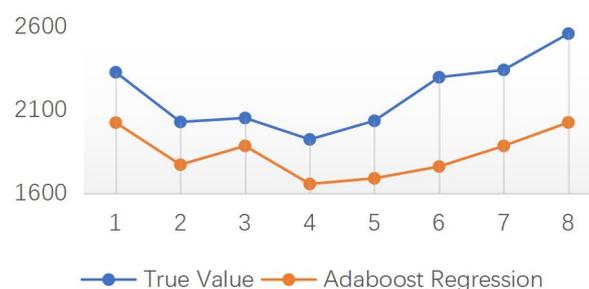
Figure 7 Adaboost Regression
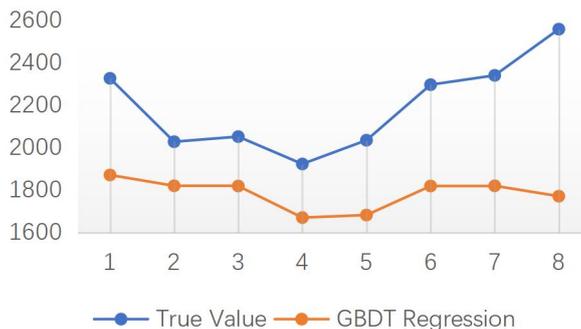
Figure 8 GBDT Regression

Figure 9 Decision Tree Regression
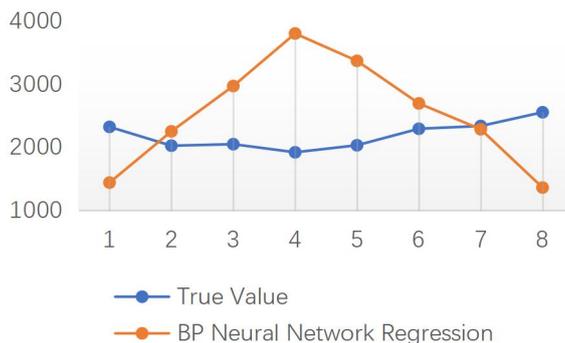
Figure 10 BP Neural Network Regression

Figure 11 Linear Regressio

According to the above visualization analysis, it can be seen that due to the pursuit of lightweight dataset samples, the performance measurement parameters of commonly used traditional machine learning regression prediction models are poor. Specifically, traditional performance evaluation metrics such as mean square error (MSE), mean absolute error (MAE), etc. mainly focus on the numerical error between predicted values and true values, but these metrics fail to fully reflect the performance of the model under the specific objectives of this paper.

The core goal of this article is to accurately identify the lowest point of environmental brightness in the sample through a predictive model, but traditional indicators have obvious limitations in evaluating whether the model can achieve this goal. Traditional indicators such as MSE and MAE mainly focus on the numerical differences between predicted values and true values, while ignoring the relative order and distribution characteristics of predicted values. For this article, accurately identifying the lowest point of environmental brightness not only requires the numerical accuracy of the predicted values, but also requires the model to correctly sort the brightness values of the samples. However, traditional indicators cannot evaluate the monotonicity and ranking ability of the model, which are crucial in this article. For example, if the model can correctly sort the brightness values of the samples, even if there is a certain error between the predicted values and the true values, the model can still effectively identify the lowest point of environmental brightness. In addition, traditional indicators have not been optimized for the specific goal of this article, which is to identify the lowest point of environmental brightness, resulting in a deviation between their evaluation results and the actual application effect of the model. Therefore, relying solely on indicators such as MSE or MAE cannot comprehensively evaluate the monotonicity and ranking ability of the model, nor can it directly measure the effectiveness of the model in identifying the lowest point of environmental brightness. This article introduces Spearman correlation coefficient and Kendall rank correlation coefficient for performance evaluation under specific objectives.

Spearman correlation coefficient is the Pearson correlation coefficient calculated by converting the raw data into rank. Measure the monotonic relationship between predicted values and true values. Sensitive to non-linear relationships, it can capture correlations even if the predicted values are not linearly related to the true values.

$$\rho = \frac{Cov(R(x), R(y))}{\sigma_{R(x)} \sigma_{R(y)}} \tag{11}$$

where R (x) and R (y) are the ranks of x and y, Cov is the covariance, and σ is the standard deviation.

The Kendall rank correlation coefficient measures the consistency of the rank of two variables. More emphasis is placed on consistency than Spearman correlation coefficient. In the case of a small sample size, Kendall rank correlation coefficient is usually relatively stable.

$$\tau = \frac{C - D}{\frac{1}{2} n(n - 1)} \tag{12}$$

where C is the number of harmonious pairs, D is the number of discordant pairs, and n is the sample size.

Spearman and Kendall rank correlation coefficients are both ranking based indicators [9] that can measure the monotonic relationship between predicted values and true values, even if they are not linear.

By using Spearman and Kendall rank correlation coefficients simultaneously, the model can be evaluated from different perspectives:

1. Comprehensive evaluation of monotonicity and consistency: Spearman correlation coefficient focuses more on the monotonicity between predicted values and true values, while Kendall rank correlation coefficient focuses more on consistency. By combining these two indicators, the ranking ability of the model can be evaluated more comprehensively.

2. Robustness assessment of outliers: Kendall is more robust to outliers, while Spearman is more sensitive to outliers. By comparing two indicators, it is possible to determine whether there are outliers in the data that affect the model.

Through Spearman correlation analysis and Kendall's tau-b correlation analysis, the true values were compared with the predicted values of various machine learning models (including Adaboost regression, decision tree regression, gradient boosting tree regression, BP neural network regression, and linear regression). The results showed that the Adaboost regression prediction model was significantly better than other models in both correlation indicators [10], as shown in Figure 12 Spearman correlation analysis thermal matrix and Figure 13 Kendall's tau-b correlation analysis thermal matrix.

Specifically, the Spearman correlation coefficient between Adaboost regression predictions and true values is 0.826, and Kendall's tau-b correlation coefficient is 0.691, both of which are higher than other models, demonstrating significant advantages in sorting ability and monotonicity. The Spearman correlation coefficient measures the monotonic relationship between predicted and true values, while Kendall's tau-b correlation coefficient further verifies the consistency of the ranking. Compared to Adaboost regression model, decision tree regression, gradient boosting tree regression, BP neural network regression, and linear regression perform worse in both indicators, indicating their limitations in ranking and monotonicity. Therefore, the Adaboost regression prediction model performs the best in the performance evaluation under the specific objectives of this article, and can more accurately reflect the distribution characteristics of the true values, providing important basis for the prediction task of the lowest point of environmental brightness.

Figure 12 Spearman Correlation Analysis Thermal Matrix

Figure 13 Kendall's tau-b Correlation Analysis Thermal Matrix

4 CONCLUSION

This article proposes a decision guidance method based on an environmental brightness prediction model to address the problems of path planning failure and energy waste in robot recognition of dark areas in the environment. By combining machine vision image processing technology, light intensity sensors, and depth cameras, a multi parameter lightweight dataset (including light intensity, shading parameters, dynamic parameters, and depth parameters) is collected, and the measured light intensity values are subjected to data bias calibration and average filtering to ensure data accuracy. This article innovatively introduces dynamic parameters and depth parameters to consider the impact of short-term environmental changes and spatial distribution on environmental brightness, effectively avoiding misjudgment and collision risks.

1. Multi parameter fusion: Traditional methods rely solely on environmental images to calculate brightness information, which results in significant errors. This article introduces dark parameters, dynamic parameters, and depth parameters to capture the characteristics of environmental brightness from multiple dimensions, significantly improving the accuracy and robustness of the prediction model.

2. Data processing optimization: In response to measurement errors in light intensity sensors, data bias calibration and average filtering techniques have been proposed to effectively reduce measurement errors and ensure data reliability.

3. Application of Adaboost regression model: Adopting the Adaboost regression model and utilizing its adaptive enhancement mechanism to optimize prediction performance. The analysis of feature importance shows that the contribution rate of dark parameters reaches 70%, and the dynamic and depth parameters account for 30%, which verifies the rationality and progressiveness of the introduction of these parameters.

In order to more accurately evaluate the performance of different machine learning methods under the specific objectives of this article, Spearman correlation coefficient and Kendall rank correlation coefficient were introduced as evaluation metrics. The experimental results show that the Adaboost model outperforms comparative models such as decision trees and gradient boosting trees in terms of Spearman (0.826) and Kendall (0.691) correlation coefficients, verifying its significant advantage in predicting the lowest point of environmental brightness [11]. This indicates that the Adaboost regression model can not only accurately capture the distribution characteristics of environmental brightness, but also effectively identify the lowest point of environmental brightness, providing reliable decision support for robot navigation in complex environments.

In summary, this article provides an innovative and efficient solution for predicting environmental brightness, which provides important technical support for autonomous navigation and decision-making of robots in complex environments. Future research will continue to revolve around this direction, continuously optimizing and improving relevant technologies and methods.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

REFERENCES

[1] Tomczuk P, Chrzanowicz M, Jaskowski P, et al. Evaluation of street lighting efficiency using a mobile measurement system. Energies, 2021, 14(13): 3872.

[2] Delaney D T, O'Hare G M P, Ruzzelli A G. Evaluation of energy-efficiency in lighting systems using sensor networks//Proceedings of the first ACM workshop on embedded sensing systems for energy-efficiency in buildings. 2009: 61-66.

[3] Murali S, Govindan V K, Kalady S. A survey on shadow detection techniques in a single image. Information technology and control, 2018, 47(1): 75-92.

[4] Narayan A, Tuci E, Labrosse F, et al. A dynamic colour perception system for autonomous robot navigation on unmarked roads. Neurocomputing, 2018, 275: 2251-2263.

[5] Miclea V C, Nedevschi S. Monocular depth estimation with improved long-range accuracy for UAV environment perception. IEEE Transactions on Geoscience and Remote Sensing, 2021, 60: 1-15.

[6] Gai J, Xiang L, Tang L. Using a depth camera for crop row detection and mapping for under-canopy navigation of agricultural robotic vehicle. Computers and Electronics in Agriculture, 2021, 188: 106301.

[7] Zhang X, Chen L, Zhang F, et al. Research on the accuracy and speed of three-dimensional reconstruction of liver surface based on binocular structured light. IEEE Access, 2021, 9: 87592-87610.

[8] Yaqin Q I N, Qiugu L I, Pengyan Z, et al. Research on risk perception tendency of drivers based on multi-class Adaboost algorithm. China Safety Science Journal, 2022, 32(4): 141.

[9] Arsov N, Dukovski M, Evkoski B, et al. A measure of similarity in textual data using Spearman's rank correlation coefficient. arXiv preprint arXiv:1911.11750, 2019.

[10] Maydanchi M, Ziaei A, Basiri M, et al. Comparative Study of decision tree, adaboost, random forest, Naïve Bayes, KNN, and perceptron for heart disease prediction//SoutheastCon 2023. IEEE, 2023: 204-208.

[11] Ramakrishna M T, Venkatesan V K, Izonin I, et al. Homogeneous adaboost ensemble machine learning algorithms with reduced entropy on balanced data. Entropy, 2023, 25(2): 245.

WIRELESS SENSOR NETWORKS BASED ON IMPROVED DUNG BEETLE ALGORITHM

QiYuan Shen[1*], YunLong Xia[2]

[1]*College of Electronics and Information (Micro-Nano Technology College), Qingdao University, Qingdao 266071, Shandong, China.*
[2]*College of Electrical Engineering and Information Technology, Lanzhou University of Technology, Lanzhou 730050, Gansu, China.*
Corresponding Author: QiYuan Shen, Email: sqy_qdu@163.com

Abstract: To address the limitations of traditional Dung Beetle Optimization (DBO) in the wireless sensor network coverage problem, specifically like low convergence speed and susceptibility to local optima, this paper proposes an optimization scheme based on the Improved DBO algorithm (IDBO). The algorithm combines three key strategies: first, a Logistic chaos initialization strategy is used to generate a more optimal initial solution; second, a Levy flight strategy is introduced to enhance the global search capability; and finally, the search process is further optimized by using a dynamic nonlinear convergence factor to adaptively adjust the search step size. With the above improvements, the algorithm is significantly improved in global search performance. Experiments on the CEC2005 benchmark suite show that IDBO outperforms similar algorithms and approaches the global optimum. Finally, at last, the proposed IDBO algorithm is applied to the wireless sensor network coverage optimization problem for simulation experiments. The simulation results show that the improved IDBO algorithm improves the coverage of the network nodes by 4.9% compared to the basic DBO algorithm and enhances the overall performance of the network with good practicality, stability and robustness.
Keywords: Dung beetle optimization; Wireless sensor network coverage; Logistic chaotic initialization; Levy flight strategy; Dynamic nonlinear convergence factor

1 INTRODUCTION

In recent years, with the continuous progress of social science and technology, the Internet, and the continuous development of artificial intelligence technology, the Internet of Things (IoT) has become one of the popular research fields nowadays [1]. Wireless Sensor Networks (WSNs), as the core carrier of IoT technology, plays an irreplaceable role in the fields of environmental monitoring, smart city management, industrial automation, and disaster warning [2]. Its core objective is to realize efficient coverage and real-time monitoring of the target area through cooperative sensing and data transmission of distributed nodes [3]. However, the layout strategy of sensor nodes directly determines the coverage quality, energy efficiency and deployment cost of the network [4]. Especially in complex scenarios, how to optimize node layout through scientific methods and break through the limitations of traditional algorithms has become the focus and difficulty of WSNs research [5].

Currently, WSNs coverage optimization problems usually involve nonlinear constraints, multi-objective trade-offs and large-scale search spaces, and traditional deterministic algorithms (e.g., gradient descent, exhaustive search, etc.) can hardly meet the practical needs because they are prone to fall into the local optimal solutions and have low computational efficiency [6]. In recent years, meta-heuristic algorithms (e.g., genetic algorithm [7], particle swarm optimization [8], etc.) have gradually become an effective tool for solving such complex optimization problems by virtue of their global search capability and adaptability.

Coverage optimization for wireless sensor networks aims to maximize the coverage of the monitored area by rationally deploying nodes while taking into account energy consumption, network lifetime and cost. In the last five years, meta-heuristic algorithms have been widely used in academia and industry, for example, literature proposes a node localization optimization method based on the bat algorithm for solving the localization error problem due to environmental noise in WSNs [9]. The algorithm significantly improves the localization accuracy by dynamically adjusting the frequency and pulse firing rate, and reduces the average localization error by about 25% in complex environments Literature designs a dynamic cluster head selection strategy based on a differential evolutionary algorithm [10], which extends the network lifetime by optimizing the energy and distance parameters of the candidate nodes. Simulation results show that this algorithm improves the network survival time by about 30% compared to the traditional LEACH protocol Literature applies Teaching-Learning-Based Optimization Algorithm (TLBO) to the problem of energy balancing in WSNs to optimize the node duty cycle by simulating the "teacher-student" interaction mechanism [11]. Experiments show that the network energy distribution uniformity is improved by 40% and the overall lifetime is extended by 20%. Literature proposes the multi-objective Sparrow Search Algorithm (MSSA) to solve the trade-off problem between coverage voids and energy consumption in WSNs [12]. By introducing the Pareto optimal solution set, the coverage reaches 96% at 1000 node size while energy consumption is reduced by 15%. Literature combines the Sine Cosine Algorithm (SCA) with Q-learning and proposes an adaptive routing protocol (SCA-QL) to

optimize the data transmission paths in WSNs [13]. The scheme reduces end-to-end delay by 15% and improves link reliability in dynamic environments.

Dung Beetle Optimizer (DBO) is a hyper-inspired algorithm proposed by Prof. Bo Shen's team in 2023, inspired by the bionics of five different behaviors: ball-rolling, dancing, reproduction, foraging, and stealing. With its simplicity, small number of parameters, and powerful search and exploration capabilities, DBO is widely used in optimization tasks, including high-dimensional feature selection and data clustering. Despite its advantages, DBO has the disadvantages of slow convergence and convergence to a local optimum. Based on this drawback, this study introduces an improved DBO algorithm that combines the mechanisms of Logistic Chaos Mapping, Nonlinear Convergence Factor with levy flight strategy for improvement.

Logistic chaotic mapping, as a classical chaotic system with traversal, randomness and sensitivity to initial conditions, improves the diversity of initial solutions by about 20-30% in complex multi-peak function optimization, which greatly enhances the population diversity. In addition, the nonlinear convergence factor can dynamically adjust the exploration step size according to the iteration progress, with a large step size approaching the optimal solution quickly at the initial stage and a small step size finely adjusted at the later stage, thus accelerating the convergence speed, enhancing the global search capability and robustness, and ultimately improving the quality of the solution. levy flight strategy further enhances the population diversity, and the algorithm is capable of exploring a wider solution space through the long-step jumps, which reduces the local optimal trapping risk. Empirical results from the CEC2005 benchmark suite, using 23 test functions, verify that the IDBO algorithm greatly improves global optimization performance. This leads to a significant acceleration of convergence and improved accuracy. The experimental data show that the algorithm proposed in this study overcomes the above five algorithms in terms of computational complexity and node resource constraints, also makes up for the lack of dynamic topology adaptation, achieves multi-objective optimization capability and reduces the complexity of parameter tuning as compared to the five algorithms: GWO, COA, WOA, HHO and the original DBO algorithm. Therefore, the main advantage of IDBO algorithm applied in wireless sensor networks (WSN) is that it shows higher stability, accuracy and efficiency in finding the optimal solution, which highlights its superior overall performance.

2 BASIC DUNG BEETLE ALGORITHM

The Dung Beetle Optimization Algorithm classifies dung beetles into four species based on the division of labor in the dung beetle population: ball dung beetles, breeding dung beetles, little dung beetles, and thief dung beetles. 20%, 20%, 25%, and 35% of the dung beetles in the population are of the four species, respectively. Different types of dung beetles update their positions according to different position updating methods, and the position represents the solution to the problem.

2.1 Ball Dung Beetle

Dung beetles in nature will move in a straight line using natural light as a navigation when rolling a dung ball, the intensity of the light source will affect the path of the dung beetles' movement, at this time their position is updated as shown in (1). However, the natural environment is full of unknowns, when encountering impassable places, at this time they will redefine the target of their actions through unique behaviors, the dancing formula is shown in (2):

$$\begin{cases} x_i(t+1) = x_i(t) + \alpha \times k \times x_i(t-1) + b \times \Delta x \\ \Delta x = |x_i(t) - X^w| \end{cases} \tag{1}$$

$$x_i(t+1) = x_i(t) + \tan(\theta)|x_i(t) - x_i(t-1)| \tag{2}$$

Here, t denotes the current iteration number; $x_i(t)$ represents the position of the i-th dung beetle at the t-th iteration; $k \in (0,0.2]$, indicates the deflection coefficient; $b \in (0,1)$; Δx quantifies light intensity, where X^W denotes the globally worst position; α is probabilistically assigned a value of 1 or -1.; When $\alpha = 1$, the direction remains unchanged, When $\alpha = -1$, it deviates from the original direction; $\theta \in [0, \pi]$, represents the deflection angle.

2.2 Breeding Dung Beetles

Dung beetles will roll balls of dung to a safe location to create a safe and secluded environment for female dung beetles to lay their eggs. This safe area is defined below:

$$\begin{cases} Lb^* = max(X^* \times (1-R), Lb) \\ Ub^* = min(X^* \times (1-R), Ub) \end{cases} \tag{3}$$

where, X^* denotes the current local optimal solution; the inertia weight $R = 1 - t/T_{max}$, where T_{max} is the maximum iteration count. Lb and Ub denote the lower and upper bounds of the optimization problem, respectively, while Lb^* and Ub^* define the dynamically adjusted boundaries for the breeding region. As illustrated in Equation (3), dynamically modifying these region boundaries plays a crucial role in preventing the algorithm from prematurely converging to local optima. Accordingly, the position of individuals is updated adaptively based on the following formulation:

$$B_i(t+1) = X^* + b_1 \times (B_i(t) - Lb^*) + b_2 \times (B_i(t) - Ub^*) \tag{4}$$

In Equation (4), $B_i(t+1)$ represents the position of the t-th breeding ball at the i-th interation, b_1 and b_2 are random vectors of size $1 \times d$, where d is the problem dimension.

2.3 Small Dung Beetles

Newly hatched dung beetles search for food, and the optimal foraging area is defined as:
$$Lb^b = max\ (X^b \times (1-R), Lb) \tag{5}$$
$$Ub^b = min\ (X^b \times (1-R), Ub)$$
In Equation (5), X^b is the global optimal position; Lb^b and Ub^b denote the lower and upper bounds of the foraging region, respectively.
The position update rule for small dung beetles is defined as:
$$x_i(t+1) = x_i(t) + C_1 \times (x_i(t) - Lb^b) + C_2 \times (x_i(t) - Ub^b) \tag{6}$$
In Equation (6), C_1 is a random number following a normal distribution, and C_2 is a random vector within the interval (0,1).

2.4 Thieving Dung Beetles

Thieving dung beetles like to get something for nothing, and their location is updated in the following way:
$$x_i(t+1) = X^b + S \times g \times (|x_i(t) - X^*| + |x_i(t) - X^b|) \tag{7}$$
Here，g is a random vector following a normal distribution with a size of $1 \times d$；and S is a constant.

3 IMPROVED DUNG BEETLE ALGORITHM

3.1 Population Initialization via Logistic Chaotic Mapping

The standard DBO algorithm uses a function to initialize the population, and the population generated by this method has a large randomness and will not cover the whole space uniformly, which reduces the algorithm's search efficiency. Therefore, in this paper, we adopt the method of initializing the population with chaotic mapping to improve the randomness of the algorithm's search, so that it can better traverse the whole search space.
Logistic chaotic mapping is publicized as follows:
$$x_{n+1} = \mu \cdot x_n \cdot (1 - x_n) \tag{8}$$
In Equation (8), $x_n \in [0,1]$ denotes the system state at step n，and μ is a control parameter typically within [0,4]，When $\mu \in (3.56995,4]$, the system exhibits chaotic behavior.

3.2 Nonlinear Convergence Factors

In dung beetles in reproductive behavior, where, as the iterative process of adaptation decreases. With the gradual decrease, the influence of the initial boundary individual will become smaller and smaller, and in the late iteration, the individual will be oriented to the optimal position obtained from its own position in the previous iteration for local optimization, although this strategy improves the algorithm's optimization ability to a certain extent, however, from the above analysis, it can be seen that the method only takes into account the positive influence of the dung beetle's optimal individual on its movement, and ignores the original boundary brought about by the Positive influence, i.e., when the optimal position of the dung beetle's individual is near the original boundary, the strategy may still cause the individual to miss the better adapted position, therefore, the present improvement proposes an improved weight formula, and uses a kind of random number to perturb the weights, so that the initial boundary can be selected at a later stage. The details are as follows.
$$R = 1 - \text{rand} \times \sqrt{\frac{t}{M}} \tag{9}$$
In Equation (9)，t denotes the current iteration count，，M is the maximum iteration count，and rand is a uniformly distributed random number within [0, 1], This mechanism introduces randomness to avoid local optima and enhance global search capability.

3.3 Levy Flight Strategy

In the original algorithm, the original formula corresponding to the rolling ball moves in the form of a straight line, which is not only unable to traverse the whole globe efficiently, but also does not have a deep search for the localization. Therefore the rolling ball phase formula of the dung beetle algorithm is improved.
The Levy flight strategy is a widely used technique to enhance the global search capability of optimization algorithms. It allows for dynamic adaptation to the search space and helps maintain search diversity, thereby improving the algorithm's ability to explore globally. In the context of Dung Beetle Optimization (DBO), Levy flight is typically

integrated into the position update mechanism to strengthen the global exploration performance of dung beetles. The improved position update formula incorporating Levy flight is expressed as follows:

$$x_i(t+1) = X^b + S \times g \times \left(|x_i(t) - X^*| + \text{Levy}(\beta) \cdot |x_i(t) - X^b|\right) \tag{10}$$

In Equation (10), $X_i(t)$ represents the position of the $i - th$ individual at iteration t, and X_{best} is the best-known position. α is a scaling factor controlling step size. Levy flight is characterized by step lengths following the Levy distribution with parameter β, enabling long-distance jumps in the search space to improve global.

After simulating the high-dimensional multi-peak functions such as Ackley and Griewank, it is found that the global optimal discovery probability of the DBO is significantly improved and the convergence speed is accelerated after the introduction of Levy flights. By integrating the content from the previous section with the DBO algorithm, the pseudo-code of the IDBO algorithm is presented as follows:

Algorithm 1 IDBO

Require: The maximum iterations T_{max}, the size of the particle's population N.
1: Utilize Logistic chaos mapping for population initialization $i \leftarrow 1, 2, ...,N$ and define its relevant parameters
2: **while** ($t \leq T_{max}$) **do**
3: **for** $i \leftarrow 1$ to N **do**
5: Synthesize novel individual positions utilizing (13) and (14), determine its fitness value f^b for the dung beetle population
4:     **if** $i ==$ ball-rolling dung beetle **then**
5:       $\lambda = \text{rand}(1)$;
6:     **if** $\lambda < 0.9$ **then**
8:     Update the ball-rolling dung beetle's position by using(1);
9:     **else**
10:      Update the ball-rolling dung beetle's position by using(2);
11:     **end if**
12:   **end if**
13:   Revise the position of the brood ball according to (4);
14:   Enforce the constraints on the new position using the upper and lower bounds defined by (3) and (9);
15:   Modify the position of the subordinate dung beetle in accordance with (5) and (9);
16:   Update the position of the small dung beetle according to (6).;
17:   Adjust the position of the thief dung beetle using (10)
18:   Update the global best position X^b and the worst position X^w
19:   **if** the newly generated position is better than before **then**
20:     Update it;
21:   **end if**
22: $t = t + 1$;
23: **end while**
24: **return** X^b and f^b

---

4 EXPERIMENT AND COMPREHENSIVE ANALYSIS

The simulations were conducted on a 64-bit Windows 11 operating system. All analyses were executed using MATLAB 2023b on a computing platform powered by an AMD Ryzen 7 4800H processor (2.30 GHz) and equipped with 16 GB of RAM.

4.1 Test Functions and Parameter Settings

The CEC 2005 [14] benchmark suite was utilized to assess the performance of the proposed IDBO algorithm. This suite encompasses four distinct categories of functions: unimodal, multimodal, hybrid, and composite, each designed to test specific aspects of an algorithm's capabilities. By incorporating this variety, the benchmark framework offers a thorough and systematic evaluation of the algorithm's performance and generalizability. These diverse test functions serve as a robust experimental platform for analyzing the scalability and effectiveness of optimization algorithms, allowing for a more precise assessment of IDBO's adaptability across various problem types..

4.2 Comparison with other Algorithms and Parameter Settings

The performance of the IDBO algorithm was systematically compared against 6 well-known algorithms. This includes the Particle swarm optimization (PSO) [15], Genetic algorithm (GA) [16], Grey wolf optimizer (GWO) [17], Whale Optimization Algorithm [18] (WOA), Arctic puffin optimization [19] (APO), original Dung Beetle optimization algorithm (DBO) [20].
Table 1 summarizes the parameter settings for the seven different MH algorithms. For each algorithm, 30 independent simulation runs were performed, each limited to 500 iterations with a population size of 30. The performance outcomes

of each algorithm were documented, including the mean (denoted as Ave) and standard deviation (Std). To enable a clear comparison across algorithms, the best results obtained under each test condition are highlighted in bold within the table. This visual emphasis effectively identifies the top-performing algorithms for each specific scenario.

Table 1 Parameter Configurations for Competing Algorithms

Algorithms	Parameter	Value
PSO	C_1, C_2	2, 2
GA	Cross, mutation	0.2, 0.1
GWO	α	[0,2]
WOA	a, a2, b	[0,2], [- 1,-2], 1
APO	F, C	0.5, 0.5
DBO	RDB, EDB, FDB, SDB	6, 6, 7, 11
IDBO	RDB, EDB, FDB, SDB	6, 6, 7, 11

4.3 Quantitative Evaluation

In this section, the performance of the IDBO algorithm is rigorously evaluated using the CEC2005 benchmark suite. To ensure consistency, experimental parameters were standardized: the population size was set to 30, the maximum number of iterations to 500, and each algorithm was executed over 30 independent runs. Table 2 presents a detailed comparison of the results obtained by all competing algorithms, reporting both the average (Ave) and standard deviation (Std) values. The advantages of IDBO are underscored through comprehensive statistical analysis. The first row of the performance summary displays the Friedman mean ranks for each algorithm, while the second row provides the final rankings based on the Friedman test. To highlight the most effective results, the best-performing values are bolded in the table. Additionally, Figure 1 illustrates the convergence curves of the algorithms, providing a visual representation of their optimization efficiency and iterative progress toward the global optimum. The results collectively demonstrate the robustness and adaptability of the IDBO algorithm across a range of optimization scenarios.

The detailed experimental outcomes are presented in Table 2, with the top-performing results among the seven algorithms highlighted in bold. Notably, IDBO consistently achieved the best or near-best optimization results across all test functions and never ranked last in any case. From a statistical perspective, IDBO secured the highest rank, while DBO followed with a Friedman test mean of 2.54. This not only underscores the inherent strength of the original DBO algorithm but also emphasizes the necessity for further algorithmic enhancement. These findings clearly demonstrate that the proposed IDBO outperforms several state-of-the-art algorithms within its category. Figure 1 displays the convergence curves across various dimensions, revealing that IDBO achieves the fastest convergence rate, superior solution accuracy, and exhibits the capability to escape local optima during the iteration process. In summary, the experimental results strongly confirm the effectiveness and superiority of IDBO as a robust and high-performance optimization algorithm.
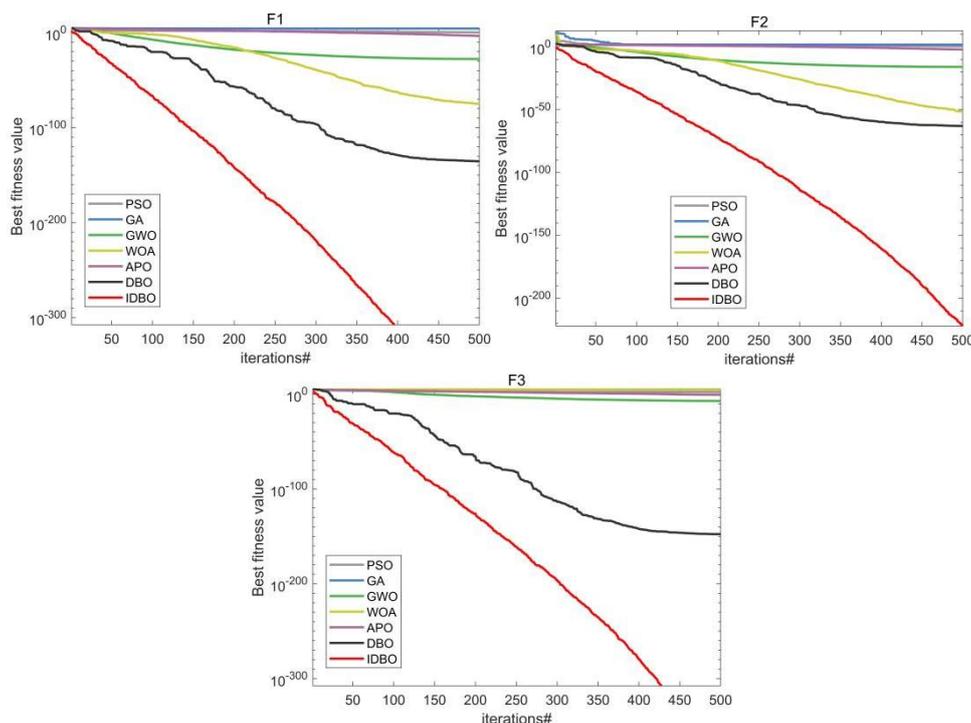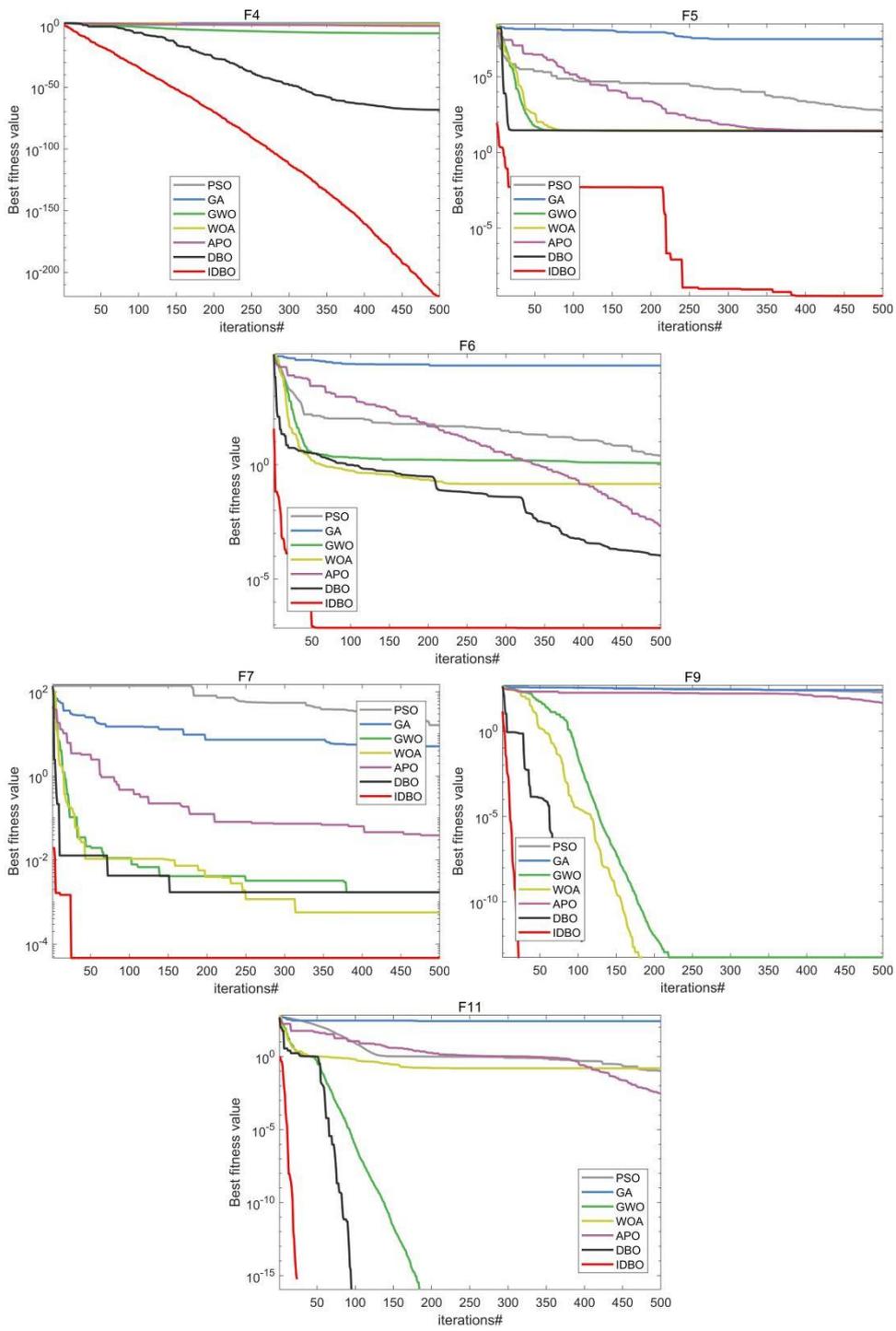
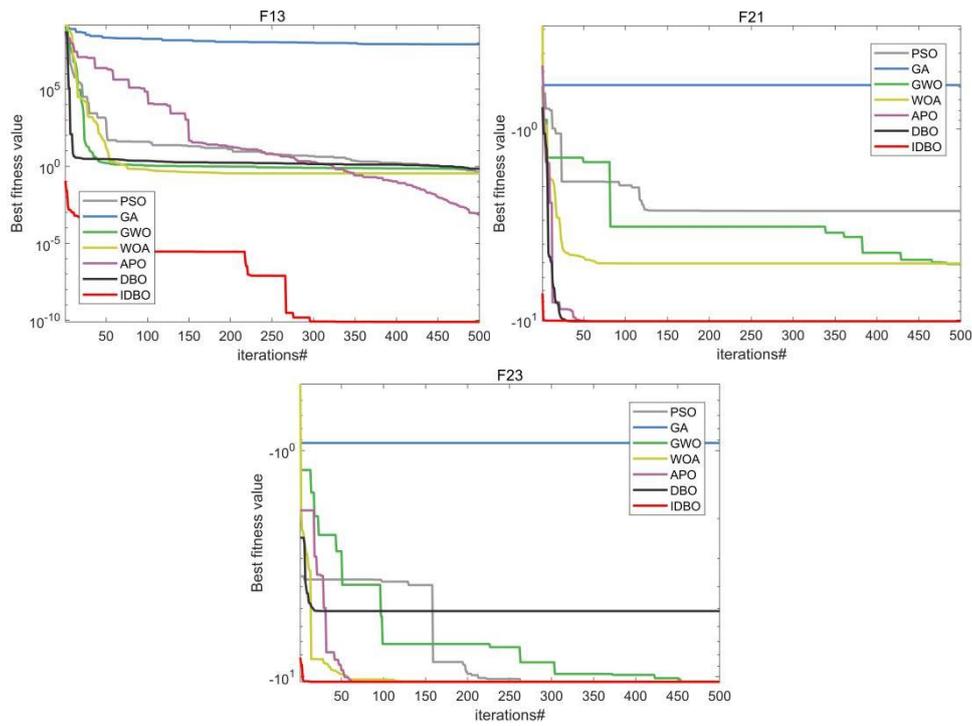**Figure 1** Convergence Curves of Different Algorithms

Table 2 Comparison of Results on CEC 2017 (Dim=30)

ID	Metric	PSO	GA	GWO	WOA	APO	DBO	IDBO
CEC2005-F1	Ave	2.1899E+00	2.0311E+04	1.1309E-27	2.7988E-71	8.0822E-04	3.2967E-111	0.0000E+00
	Std	1.1531E+00	6.7213E+03	1.5597E-27	1.5276E-70	8.7209E-04	1.8056E-110	0.0000E+00
CEC2005-F2	Ave	4.3051E+00	5.5969E+01	8.8141E-17	8.2682E-51	9.3412E-03	8.4919E-56	9.3684E-213
	Std	1.3590E+00	1.0813E+01	6.3964E-17	2.9380E-50	4.3593E-03	4.1934E-55	0.0000E+00
CEC2005-F3	Ave	1.7406E+02	5.4615E+04	9.5897E-06	4.2889E+04	2.1652E-01	1.1754E-23	0.0000E+00
	Std	4.2307E+01	1.4129E+04	1.9828E-05	1.5195E+04	1.4867E-01	6.4381E-23	0.0000E+00
CEC2005-F4	Ave	1.9450E+00	7.2056E+01	8.2342E-07	5.5313E+01	7.2780E-01	2.2579E-58	2.2095E-210
	Std	2.7640E-01	8.1057E+00	7.9958E-07	3.0214E+01	2.2914E-01	1.0107E-57	0.0000E+00
CEC2005-F5	Ave	1.1232E+03	2.1224E+07	2.7063E+01	2.7830E+01	2.4330E+01	2.5715E+01	3.6246E-04
	Std	7.1871E+02	1.7317E+07	6.8524E-01	4.6210E-01	9.0538E+00	1.8181E-01	6.7064E-04
CEC2005-F6	Ave	2.1401E+00	2.2294E+04	6.8072E-01	4.3627E-01	2.1814E-03	6.0633E-04	8.0633E-07
	Std	9.9516E-01	7.4702E+03	3.1907E-01	2.9702E-01	1.5357E-03	9.2715E-04	1.6071E-06
CEC2005-F7	Ave	2.2886E+01	1.1649E+01	1.9375E-03	3.4420E-03	2.9600E-02	1.3904E-03	3.5056E-05
	Std	1.9494E+01	8.3489E+00	1.3652E-03	3.6722E-03	1.1486E-02	1.1834E-03	2.9257E-05
CEC2005-F8	Ave	-6.3628E+03	-2.1164E+03	-6.3408E+03	-1.0706E+04	-6.7605E+03	-8.4969E+03	-1.2569E+04
	Std	1.1316E+03	5.0442E+02	7.4970E+02	1.5905E+03	1.2927E+03	1.8384E+03	8.4815E-02
CEC2005-F9	Ave	1.6674E+02	2.6023E+02	3.9434E+00	0.0000E+00	9.7766E+01	0.0000E+00	0.0000E+00
	Std	2.9415E+01	4.8325E+01	4.4291E+00	0.0000E+00	4.0324E+01	0.0000E+00	0.0000E+00
CEC2005-F10	Ave	2.6533E+00	1.9770E+01	1.0205E-13	4.2337E-15	7.3861E-03	4.4409E-16	4.4409E-16
	Std	4.2985E-01	5.3143E-01	1.8202E-14	2.0723E-15	3.3874E-03	0.0000E+00	0.0000E+00
CEC2005-F11	Ave	1.3633E-01	2.1844E+02	1.9540E-03	9.3348E-03	8.7823E-03	0.0000E+00	0.0000E+00
	Std	4.8881E-02	8.2964E+01	6.0843E-03	3.5852E-02	1.1186E-02	0.0000E+00	0.0000E+00
CEC2005-F12	Ave	7.1152E-02	2.4170E+07	4.9462E-02	2.5163E-02	1.3885E-02	3.1131E-04	5.5409E-08
	Std	1.1945E-01	3.2642E+07	2.5965E-02	3.0658E-02	3.5847E-02	1.6001E-03	9.4659E-08
CEC2005-F13	Ave	5.4968E-01	8.7242E+07	6.6725E-01	5.0125E-01	4.7669E-03	7.7574E-01	1.1949E-06
	Std	2.0763E-01	8.0554E+07	2.3682E-01	2.1816E-01	5.3068E-03	3.9018E-01	2.7725E-06
CEC2005-F14	Ave	3.3314E+00	1.1788E+00	4.0321E+00	2.8335E+00	9.9800E-01	1.0311E+00	9.9800E-01
	Std	2.5520E+00	5.2228E-01	4.1917E+00	2.9226E+00	0.0000E+00	1.8148E-01	3.3173E-10
CEC2005-F15	Ave	8.8995E-04	1.1941E-02	3.7659E-03	7.1993E-04	3.0749E-04	8.5180E-04	3.2056E-04
	Std	1.0151E-04	1.3163E-02	7.5513E-03	3.6179E-04	1.4057E-19	4.4954E-04	1.5452E-05
CEC2005-F16	Ave	-1.0316E+00	-9.4964E-01	-1.0316E+00	-1.0316E+00	-1.0316E+00	-1.0316E+00	-1.0316E+00
	Std	4.7012E-16	1.1049E-01	1.3815E-08	5.5790E-10	6.7752E-16	5.7578E-16	5.6453E-11
CEC2005-F17	Ave	3.9789E-01	7.0195E+01	3.9793E-01	3.9789E-01	3.9789E-01	3.9789E-01	3.9789E-01
	Std	0.0000E+00	7.9426E+00	1.5275E-04	9.1911E-06	0.0000E+00	3.2434E-16	1.8070E-07
CEC2005-F18	Ave	3.0000E+00	7.5344E+00	3.0000E+00	3.0000E+00	3.0000E+00	3.9000E+00	3.0065E+01
	Std	4.5944E-15	1.0296E+01	5.7361E-05	1.1427E-04	1.3143E-15	4.9295E+00	3.2483E-01
CEC2005-F19	Ave	-3.8628E+00	-3.3348E+00	-3.8609E+00	-3.8570E+00	-3.8628E+00	-3.8612E+00	-3.8037E+00
	Std	2.1615E-15	3.3167E-01	3.2182E-03	6.6394E-03	2.7101E-15	3.2065E-03	5.2125E-02

ID	Metric	PSO	GA	GWO	WOA	APO	DBO	IDBO
CEC2005-F20	Ave	-3.2863E+00	-1.5596E+00	-3.2609E+00	-3.2451E+00	-3.3180E+00	-3.2394E+00	-3.0982E+00
	Std	5.5415E-02	5.4189E-01	7.3342E-02	9.5655E-02	2.1707E-02	8.3431E-02	1.2513E-01
CEC2005-F21	Ave	-6.9735E+00	-7.7663E-01	-9.3971E+00	-8.3577E+00	-1.0153E+01	-7.4455E+00	-1.0153E+01
	Std	3.1472E+00	6.4577E-01	1.9938E+00	2.5854E+00	7.0670E-15	2.5750E+00	5.2776E-04
CEC2005-F22	Ave	-9.0125E+00	-1.2235E+00	-1.0147E+01	-8.1392E+00	-1.0403E+01	-8.9888E+00	-1.0403E+01
	Std	2.6169E+00	5.5349E-01	1.3940E+00	2.8277E+00	9.3299E-16	2.3810E+00	3.2399E-04
CEC2005-F23	Ave	-9.5982E+00	-1.2363E+00	-1.0356E+01	-6.6610E+00	-1.0536E+01	-9.3877E+00	-1.0536E+01
	Std	2.1451E+00	5.4666E-01	9.7875E-01	3.5576E+00	2.0600E-15	2.3627E+00	1.3439E-03
Friedman average		4.62	6.79	4.25	4.26	2.93	2.61	2.54
Overall Rank		6	7	4	5	3	2	1

5 MATHEMATICAL MODELING OF WIRELESS SENSOR NETWORK COVERAGE

5.1 Perceptual Models

In order to reduce the complexity of the perception model, often such problems will be simplified to a binary model, also known as Boolean perception model [21]. In the two-dimensional plane, the sensing range of a sensor node is a circular area formed by the sensor node as the center and the radius as a circle, the probability of being sensed within this range is 1, and the area beyond this range is considered unperceivable with a probability of 0. The mathematical model is:

$$P(r, S_i) = \begin{cases} 1 & d(r, S_i) \leq R \\ 0 & else \end{cases} \tag{11}$$

In Equation (1), $P(r, s_i)$ denotes the probability that sensor s_i detects point r and $d(r, s_i)$ is the Euclidean distance between s_i and r.

5.2 Coverage

Most of the coverage strategy studies for wireless sensor networks are based on solving for area coverage based on area grids. The area coverage of a sensor network is a commonly used quantitative metric, which is the basis for comparing the advantages and disadvantages of optimization algorithms.

It is assumed that there are sensors in the sensor network, and its set is denoted as:

$$S = \{s_i\}, i = 1, \cdots, n \tag{12}$$

In Equation (2): S denotes the sensor network; s_i is the $i-th$ sensor node; and n represents the total number of nodes.

Area coverage is defined as the ratio of the effective coverage area of the target area by all the sensor nodes to the total area of the target area, and is expressed as [22]:

$$\eta = \frac{\sum_{i=0}^{n} \Phi_{s_i}}{A} \tag{13}$$

In Equation (3): η is the area coverage rate; Φ_{s_i} is the sensing area of the i-th sensor, and A is the total area of the target region.

A point in the target area may be sensed by different nodes in the network, then the probability of the point being sensed by the sensor network is:

$$P(r, S) = 1 - \prod_{i=1,\cdots,n} [1 - P(r, s_i)] \tag{14}$$

Assume there are m points in the target area. The total coverage rate of the sensor network over all detected points is defined as:

$$C_{(s)} = \frac{\sum_{r=1}^{n} P(r, S)}{A} \tag{15}$$

In Equation (5): $C(s)$ denotes the total coverage rate of sensor network S over all detected points in the target area.

The optimization objective of this study is to deploy a fixed number of sensor nodes in the target area, and to maximize the coverage rate $C(s)$ of the WSN by applying intelligent algorithms to optimize node layout.

5.3 Coverage Optimization based on IDBO Algorithm

In wireless sensor networks, the region is assumed to be a two-dimensional plane. Satisfaction:
(1) All nodes use Boolean sensing model.
(2) In the wireless sensor network coverage, all sensor nodes are isomorphic with the same communication radius Rc and perception radius Rs. To ensure the connectivity of the whole network, this mathematical relationship is set as Rc ≥ 2Rs.

5.3.1 Experimental simulation and result comparison

In order to verify the superiority of the proposed improved Sparrow algorithm in wireless sensor network applications, this paper uses MATLAB2023b to simulate the proposed algorithm IDBO, and compares and analyzes it with the WSNs deployed by the other six algorithms under the same conditions. The experimental parameters are shown in Table 3, and the number of iterations is set to 200.

Table 3 Experimental Parameters

Sensing field	Nodes N	Communication radius R	Perceived radius r
50 m*50 m	30	10 m	5 m

The distribution of optimized nodes at 200 iterations is shown in Fig 2. As can be seen from Fig 2, compared with the other six high-performance algorithms, the coverage of WSN optimized by the improved IDBO is more comprehensive and nearly achieves full coverage; at the same time, from the coverage iteration curve in Fig 3, it can be seen that IDBO is significantly better than the other algorithms, and its convergence speed is faster and more stable, and it achieves the highest coverage rate of 81.84% within 200 iterations, which demonstrates a very high optimization efficiency. efficiency. In contrast, DBO, WOA and PSO are the next best algorithms, which have higher convergence coverage, but still do not surpass IDBO. In the graph, DBO also shows strong optimization ability, with faster convergence and 77.96% coverage, which indicates that its performance in the WSN coverage problem is more stable and efficient. WOA is also competitive, with faster convergence but falls into the local optimum too early, and reaches the maximum coverage of 81.84% within 200 iterations. prematurely falls into the local optimum, and finally reaches a coverage rate of 75.60%. In contrast, the performance of GA, POA and GWO is significantly weaker, which not only have slow convergence speed, but also have coverage rates of 68.64%, 67.68% and 73.96%, respectively, indicating that they fail to improve the coverage rate sufficiently in the optimization process, and their ability to adapt to complex problems is poor. On the other hand, IDBO almost always breaks through the coverage peak quickly, and still keeps the rising trend even in the late iteration, which shows that it has a strong ability to jump out of the local optimum. The optimization results for 500 iterations are shown in Table 4. By analyzing the performance of each algorithm, IDBO shows excellent optimization ability, with an optimal coverage of 81.84%, which ranks first among all algorithms and significantly outperforms other competing algorithms.
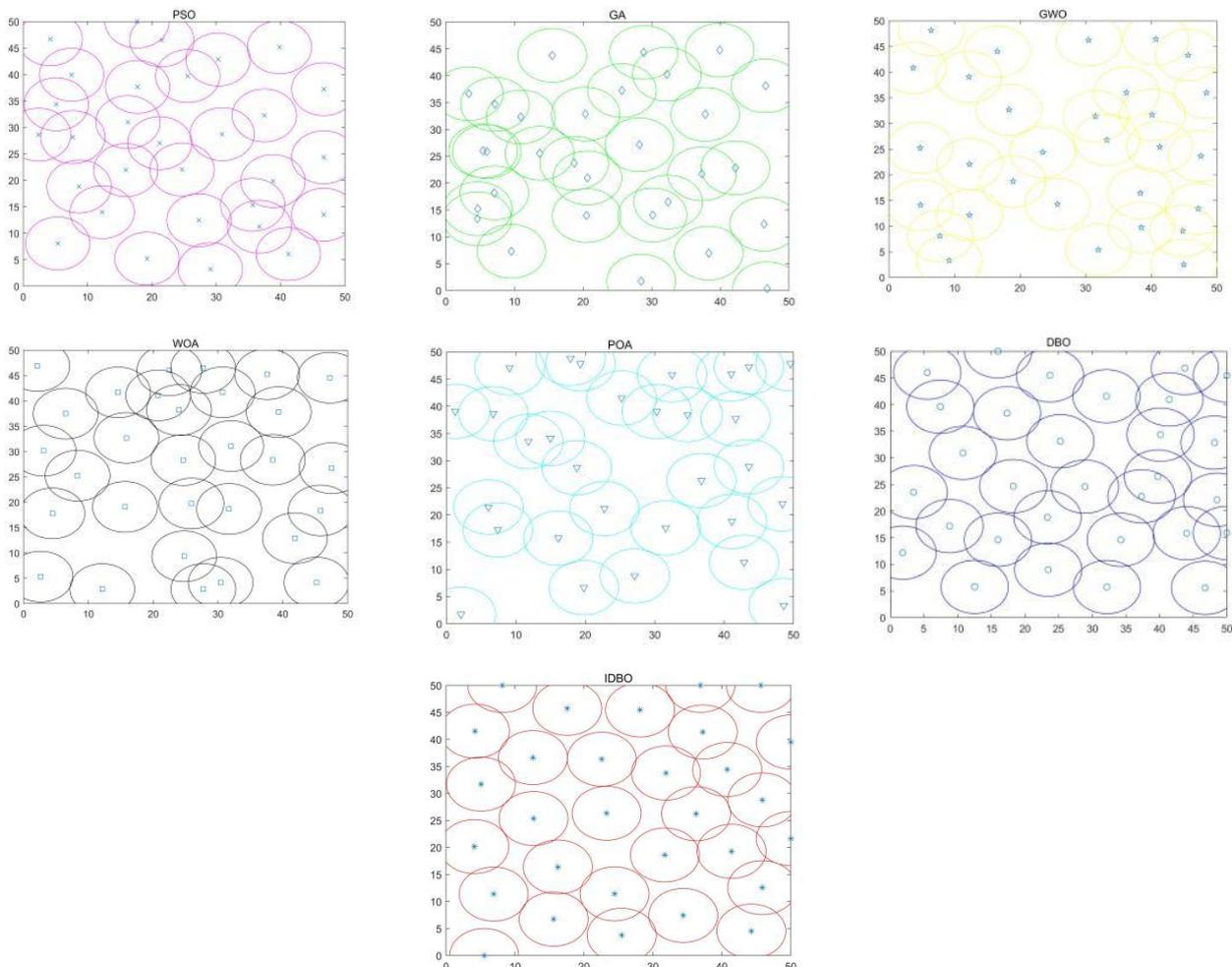
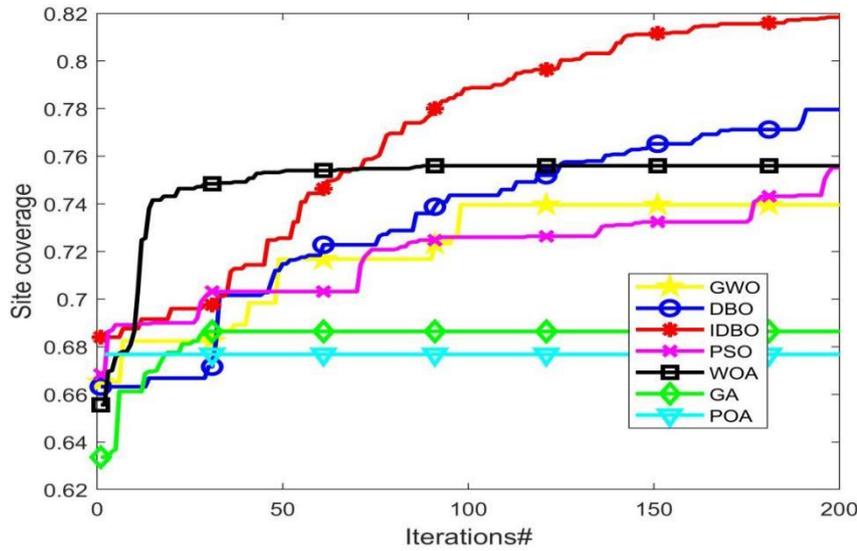**Figure 2** Optimization Node Distribution Diagram (Iteration 200 Times)

Figure 3 Coverage Optimization Line Chart (Iteration 500 Times)

Table 4 Coverage Results (iteration 500 times)

Algorithm	PSO	GA	GWO	WOA
fraction of coverage /%	75.52%	68.64%	73.96%	75.60%
Algorithm	POA	DBO	IDBO	
fraction of coverage /%	67.68%	77.96%	81.84%	

6 CONCLUSION

To tackle the challenges arising from random node deployment in the wireless sensor network (WSN) coverage problem, this study introduces an enhanced dung beetle optimization algorithm (IDBO). The algorithm combines three key strategies: first, a Logistic Chaos initialization strategy is used to generate a better initial solution; second, a Levy flight strategy is introduced to enhance the global search capability; and finally, a dynamic nonlinear convergence factor adaptively adjusts the search step size, thus further optimizing the search process. These improvements make IDBO not only have faster convergence speed, but also improve its robustness and adaptability in complex search space, effectively avoiding the local optimal solution problem.

Experimental results on the CEC2005 benchmark test set show that IDBO exhibits excellent optimization search performance in most of the test functions, with the first Friedman ranking, fully demonstrating its potential as a high-performance optimizer. To further verify the effectiveness of the proposed algorithm, IDBO is applied to solve the wireless sensor network (WSN) coverage problem. Experimental results indicate that, in comparison to the traditional DBO and other advanced algorithms such as GA, PSO, GWO, WOA, and POA, IDBO achieves notable improvements in reducing node redundancy, expanding coverage area, and enhancing coverage efficiency. Moreover, it demonstrates superior adaptability and faster convergence, highlighting its robustness in addressing complex WSN deployment scenarios. In addition, IDBO has better coverage efficiency performance compared to other intelligent optimization algorithms. Therefore, IDBO has excellent optimization performance in the wireless sensor network coverage problem, with the dual advantages of reducing network coverage cost and improving coverage rate. Future research will focus on further improving the convergence speed of the algorithm while ensuring high coverage.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Li X, Li B, Li Y. Per-connection vs. service subscription: A study on service charging model of IoT platform empowering manufacturers to upgrade product intelligence. Nankai Management Review, 2024: 1–27.
[2] Zhang J. Research on data aggregation algorithm for wireless sensor networks based on node heterogeneity. Hebei: Hebei University, 2024.

[3] Chen G, Wan W. Trust assessment and node selection strategy based on distributed cloud node task collaboration. Guangdong Communication Technology, 2023, 43(12): 45–50.

[4] Zhu J. Research on node layout, localization, and mobile node path planning problems in WSN. Shenyang: Northeastern University, 2010.

[5] Huang Q, Lai C. Research on communication route planning algorithm for WSNs based on fuzzy affiliation optimization algorithm. Automation Technology and Application, 2024: 1–8.

[6] Song J, Hu Y M, Luo Y B. Comparative analysis of swarm intelligent optimization algorithms based on WSN network coverage optimization problem. Journal of Dali University, 2024, 9(12): 65–73.

[7] Dong H, Pan F. Constant tension fuzzy control of lithium battery winding based on genetic algorithm. Automation and Instrumentation, 2025, 40(2): 43–48.

[8] Wei S, Jia H, Cao J, et al. Prediction of heat transfer performance of buried pipe based on particle swarm optimization algorithm. District Heating, 2025(1): 107–116.

[9] Wang J, et al. An improved bat algorithm for node localization in wireless sensor networks. Applied Soft Computing, 2020, 89: 106156.

[10] Li X, et al. Energy-efficient cluster head selection in WSNs using differential evolution. Engineering Applications of Artificial Intelligence, 2021, 97: 104041.

[11] Kumar S, Singh P. TLBO-based energy balancing in wireless sensor networks for IoT applications. Ad Hoc Networks, 2022, 135: 102952.

[12] Chen H, et al. Multi-objective sparrow search algorithm for coverage and energy optimization in WSNs. IEEE Sensors Journal, 2023, 23(5): 5123–5135.

[13] Zhang L, et al. SCA-based Q-learning routing protocol for energy-efficient wireless sensor networks. Computer Networks, 2021, 194: 108133.

[14] Yao X, Liu Y, Lin G. Evolutionary programming made faster. IEEE Transactions on Evolutionary Computation, 1999, 3(2): 82–102.

[15] Kennedy J. Particle swarm optimization// Proceedings of the IEEE International Conference on Neural Networks (Perth, Australia). IEEE, 1995, 4: 1942–1948.

[16] Goldberg D, Holland J. Genetic algorithms and machine learning. Machine Learning, 1988, 3(2): 95–99.

[17] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer. Advances in Engineering Software, 2014, 69: 46–61.

[18] Mirjalili S, Lewis A. The whale optimization algorithm. Advances in Engineering Software, 2016, 95: 51–67.

[19] Wang W, Tian W, Xu D, et al. Arctic puffin optimization: A bio-inspired metaheuristic algorithm for solving engineering design optimization. Advances in Engineering Software, 2024, 195: 103694.

[20] Xue J, Shen B. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization. The Journal of Supercomputing, 2023, 79: 7305–7336.

[21] Liu J S, Li W X, Li Y. LWMEO: An efficient equilibrium optimizer for complex functions and engineering design problems. Expert Systems with Applications, 2022, 198: 116828.

[22] Wang H, Zhang X, Lu H. Sensor coverage optimization strategy based on geometric coverage algorithm. Computer Applications Research, 2017, 34(8): 2478–2482.

A PHASE COMPENSATION METHOD UNDER PICKET-FENCE EFFECT

HaoRan Du

Xi'an Electronic Engineering Research Institute, Xi'an 710100, Shaanxi, China.
Corresponding Author: HaoRan Du, Email: duhaoran0214@nudt.edu.cn

Abstract: The fence effect induces significant phase deviation in fast Fourier transform (FFT) spectrum analysis, which adversely impacts the accurate measurement of signal parameters. In this paper, we propose a phase compensation method based on modeling the characteristics of spectral leakage. This approach constructs a quantitative compensation model for frequency offset and phase error by analyzing the phase distribution of the main spectral lines. By integrating interpolation correction and optimizing the window function, our method effectively mitigates the phase distortion caused by the fence effect. Simulation experiments demonstrate that the phase error can be reduced to within 0.5° compared to traditional methods, making it suitable for high-precision signal processing applications.
Keywords: Picket-fence effect; Phase correction; Spectrum analysis; Fast fourier transforms

1 INTRODUCTION

In engineering signal processing, the Fast Fourier Transform (FFT) is widely utilized in spectrum analysis due to its high efficiency[1,2]. However, the frequency components of the actual signal often do not align precisely with the discrete frequency points of the FFT, leading to spectral energy leakage and phase distortion, commonly referred to as the Picket-Fence Effect[3]. This effect is particularly pronounced in high-precision measurement scenarios. For instance, in power system fault localization, phase errors can result in deviations in the calculation of traveling wave arrival times, which subsequently impacts the accuracy of fault point localization. Similarly, in rotating machinery fault diagnosis, phase distortion of the bearing damage feature frequency can diminish the accuracy of fault classification.

In terms of the fence effect, traditional amplitude correction methods are well-established. The energy center of gravity method[4,5] corrects frequency estimation through weighted averaging; however, it exhibits poor noise immunity. The bispectral interpolation method[6,7] estimates frequency deviation by utilizing the ratio of the amplitudes of neighboring spectral lines, but it is sensitive to asymmetric leakage. Meanwhile, the spectral refinement method[8,9] enhances resolution through complex modulation, albeit at the cost of real-time performance. Existing studies predominantly concentrate on amplitude correction, while phase correction continues to encounter several significant challenges:

1)Asymmetry of spectral leakage: the window function side-flap attenuation property leads to a nonlinear shift of the main-flap phase distribution, and the traditional linear compensation model fails;

2)Noise sensitivity problem: phase estimation is significantly affected by random perturbations in high noise environments;

3) Insufficient adaptability to dynamic scenarios: In the analysis of non-stationary signals, such as new energy grid connections and high-speed bearing monitoring, it is difficult for the existing algorithms to achieve fast convergence.

In this paper, we propose a phase compensation algorithm based on a two-parameter iteration method to address the aforementioned challenges. First, we establish an analytical model of the phase distribution of Hanning window spectrum leakage to derive the nonlinear relationship between frequency bias and phase error. Second, we design a joint frequency-phase iteration mechanism that balances computational efficiency and noise immunity by integrating an adaptive window function selection strategy. Finally, we conduct simulation experiments that demonstrate the phase error can be reduced to less than 0.5° compared to traditional methods, meeting the requirements of high-precision applications such as smart grids. This study offers a comprehensive analysis of phase error in complex noise environments and provides new theoretical insights and technical support for real-time phase measurement in such conditions.

2 MODELING THE PICKET-FENCE EFFECT

Let the signal be a single-frequency complex exponential signal after adding a window:

$$x(n) = Aw(n)\exp\{j(2\pi f_0 n + \phi_0)\}, \quad n = 0,1,\dots,N-1 \tag{1}$$

Where $w(n)$ is the window function, whose spectral dominant energy distribution influences the phase deviation, and N denotes the number of sampling points.

When an FFT digital filter is employed to perform a fast Fourier transform for extracting the spectral components of a signal, the repetition frequency of the signal does not always align with the resonant frequency of the filter. Figure 1 illustrates the amplitude and phase frequency response of the FFT filter, alongside the theoretical signal spectral component G_d, which corresponds to the frequency f_0. The theoretical signal spectral component does not equal the

resonant frequency k_m of the filter; it is the difference between the two that introduces the additional phase shift $\Delta\phi$. The phase estimate of the peak frequency point k_m of the main flap after the FFT is:

$$\hat{\phi}(k_m) = \phi_0 + \Delta\phi(\delta, w) \tag{2}$$

where δ by the expression shown in equation (3) represents the normalized frequency offset and $\Delta\phi$ is the phase error term associated with the window function.

$$\delta = \frac{f_0 - k_m}{\Delta f} = \frac{f_0 - k_m}{f_s} \cdot N \tag{3}$$

where f_s denotes the sampling frequency.
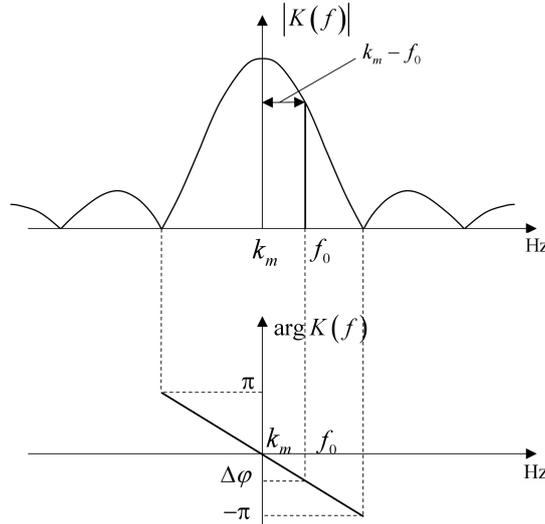
Figure 1 Amplitude-Frequency Response and Signal Spectral Components of FFT Filters

By expanding the spectral leakage function using Taylor series, the nonlinear relationship between the phase error and the frequency offset is derived as follows:

$$\Delta\phi \approx -\pi\delta\left(1 - \frac{1}{N}\right) + \frac{\pi^3\delta^3}{3N^2} \cdot C_w \tag{4}$$

where C_w is the phase correction coefficient of the window function (e.g., Hanning window $C_w = 1.5$, rectangular window $C_w = 1$). Based on Equations (2) and (4), the compensation equation can be established:

$$\phi_{\text{comp}} = \hat{\phi}(k_m) - \Delta\phi(\delta, w) \tag{5}$$

3 DESIGN OF PHASE COMPENSATION METHOD

Through the above analysis, this paper proposes a phase compensation method to cope with the fence effect, and the specific algorithm research flow is shown in Figure 2.

Figure 2 Phase Compensation Algorithm Flowchart

The specific steps of the phase compensation method are outlined below:

Step 1: Spectrum coarse estimation: obtain the primary flap peak frequency k_m and the initial phase $\hat{\phi}(k_m)$ using FFT.

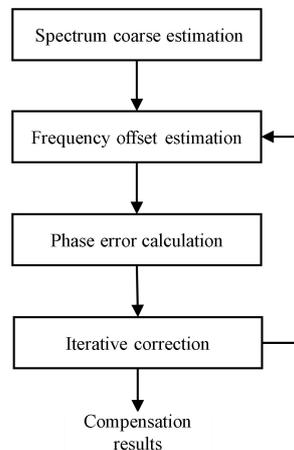Step 2: Frequency offset estimation: calculate the normalized offset δ using bispectral interpolation.

Step 3: Phase error calculation: substitute the compensation model based on the type of window function and solve for $\Delta\phi$.

Step 4: Iterative correction: Update the frequency estimation using the compensated phase and repeat steps 2 and 3 until convergence is achieved.

4 SIMULATION RESULTS

In order to verify the effectiveness and robustness of the proposed algorithm, the following simulation data, generated using the signal model presented in Equation (1), is utilized for algorithm validation. The specific signal parameters are set as follows: $f_0 = 50.5\text{Hz}, \phi_0 = 60^\circ, f_s = 1024\text{Hz}, N = 1024$.

To verify the effectiveness of the proposed method, we compared the absolute phase errors (in degrees) after phase compensation using our approach, the conventional single-spectral line method, and the bispectral line interpolation method, all under a signal-to-noise ratio (SNR) of 40dB. The simulation results are presented in Table 1.

Table 1 Comparison Results of Phase Compensation Effect of Different Methods

Method	Phase Error (°)	Frequency Error(Hz)
Single-spectral line method	4.2	0.15
Bispectral line interpolation method	1.8	0.04
Method of this paper	0.3	0.01

The data in Table 1 indicate that the compensation effect of the method proposed in this paper outperforms both the traditional mono-spectral line method and the bispectral line interpolation method under high SNR conditions, and the simulation results align with theoretical expectations.

In order to verify the robustness of the method, the SNR of the input signal is varied to compare the root mean square error (RMSE) of the method presented in this paper with that of traditional single-spectrum and bispectral interpolation methods after phase compensation at different SNR. The simulation results are illustrated in Figure 3.
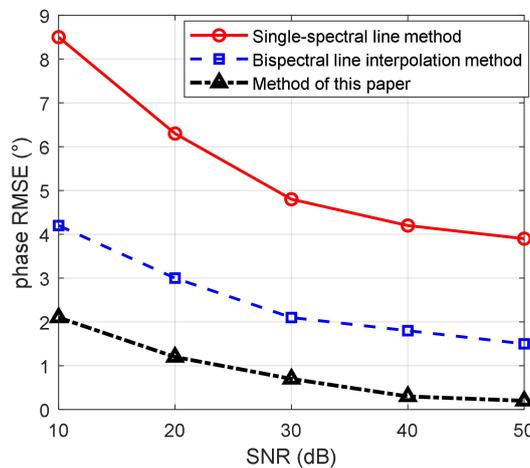
Figure 3 Comparison Results of Phase RMSE at Different SNR are Plotted

Figure 3 illustrates that the traditional single-spectral line method exhibits the largest error due to the failure to account for the nonlinear phase shift caused by spectral leakage. As the SNR increases, the RMSE decreases gradually. The bispectral line interpolation method mitigates the effects of leakage through frequency interpolation, resulting in a significant reduction in RMSE; however, a residual error remains. The method proposed in this paper integrates a phase compensation model with iterative correction, achieving an error of 2.1° at a low SNR of 10dB, which decreases to approximately 0.2° at a high SNR of 50dB, demonstrating superior stability. This approach maintains high accuracy even at low SNR levels (SNR ≤ 20dB), reducing the error by 50% to 70% compared to traditional methods. At high SNR levels (SNR ≥ 30dB), the error stabilizes within 0.5°, meeting the requirements for high-precision measurement.

5 CONCLUSIONS

In this paper, we propose an adaptive phase compensation method based on spectral leakage modeling to address the phase distortion problem caused by the fence effect in the discrete Fourier transform. By analyzing the characteristics of spectral leakage, we construct a nonlinear correction model for frequency offset and phase error, and we design a two-parameter iterative algorithm to achieve joint frequency-phase estimation. The simulation and experimental results demonstrate that this method maintains high accuracy even under low signal-to-noise ratio conditions. This study reveals the quantitative relationship between spectral leakage asymmetry and phase error, overcoming the limitations of traditional linear compensation models. Additionally, we propose a dynamic selection strategy for the window function that balances the requirements for high resolution and noise immunity, making it applicable to complex working conditions. This research provides reliable technical support for power system harmonic analysis, radar signal processing, and other applications. Future research directions include phase coupling compensation for multi-frequency signals, real-time hardware implementation of algorithms, and the expansion of interdisciplinary applications.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] A Duncan, R D Sedgewick, R D Coalson. Fast Fourier transform simulation techniques for Coulomb gases. Condensed Matter, 2005, 72(8).

[2] E J Raynolds, R L Mullin. Applications of conformal computing techniques to problems in computational physics: The fast Fourier transform. Computer Physics Communications, 2005, 170(1): 1–10.

[3] W J Cooley, J W Tukey, et al. An algorithm for the machine calculation of complex Fourier series. Mathematics of Computation, 1965, 19(90): 297–301.

[4] P Chen, J C Wang, X Y Wu. Analysis and application study of improved FFT algorithm for energy center of gravity method. Mechanical Science and Technology, 2018, 37(12): 1883–1889.

[5] Y F Zhang, B Lang, D X Zhang. Calculation of dielectric loss based on energy center of gravity method and dynamic plus window harmonic analysis. Electrical Test and Instrumentation, 2016(3): 18–22.

[6] Y W Cao, K F Zhang, Z H Jiang, et al. A robust discrete spectrum correction method. Journal of Electronics and Information, 2005(9): 1353–1356.

[7] J C Xu, Q Xu, L Q Chai. Application of spatial interpolation in power spectral density calculation. Intense Laser and Particle Beam, 2010(7): 1520–1524.

[8] K Ding, C H Pan, W H Li. Comparison of spectral analysis between ZFFT and Chirp-Z transform with refined band selection. Vibration and Shock, 2006, 25(6): 9–13.

[9] J M Liu, H Q Ying. Fourier transform method for continuous refinement analysis of FFT spectra. Journal of Vibration Engineering, 1995, 8(2): 162–167.

PERFORMANCE-SCORING DRIVEN MODEL SCALING AND SCHEDULING FOR EDGE VIDEO ANALYTICS SERVING

XingTao Xu[1], JiZhe Zhang[2], HaiTao Zhang[1*]

[1]*School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing 100876, China.*
[2]*Kembl Petroleum Technology Co., Ltd., Beijing 100029, China.*
Corresponding Author: Haitao Zhang, Email: zht@bupt.edu.cn

Abstract: The growing demand for real-time, multi-task video analytics at the edge has encountered challenges in resource-constrained environments, including redundant computations across tasks and poor adaptability to dynamic workloads. In this paper, we propose a performance scoring-driven framework for model scaling and scheduling in edge video analytics. The framework consists of two core modules: (1) the model performance scoring module evaluates the model performance from four dimensions—video complexity, task correlation, model performance, and system resource utilization. (2) The model scaling and scheduling module then calculates a comprehensive score based on these four evaluation metrics. Aiming at maximizing the comprehensive score, this module employs the particle swarm optimization algorithm for model scheduling and system resource allocation, and selects the optimal combination of detection models based on the current model and system states. Experimental results demonstrate that our framework outperforms state-of-the-art baselines, achieving superior performance under dynamic edge workloads.
Keywords: Model scaling and scheduling; Edge computing; Scoring metrics

1 INTRODUCTION

In recent years, the advancement of video surveillance technology and the increasing number of monitoring cameras in public spaces have made video analytics a prominent research field [1-2]. In practical applications, to reduce transmission latency and enhance data security, video analytics models are increasingly being deployed on edge devices rather than cloud servers [3-4]. However, it brings challenges such as constrained computational and storage resources and unstable network conditions, thus leading to degraded accuracy or compromised real-time performance in edge-based video analytics. Existing works primarily optimize accuracy-latency trade-off or resource allocation in edge video analytics through model scaling and scheduling, aiming to improve system quality of service (QoS). Nevertheless, these methods underperform in the multi-task video analytics scenarios due to the following reasons:

1.1 Task Dependencies Induce Redundant Computations

In such edge scenarios, multiple video analytics tasks exhibit dependencies or correlations [5]. For instance, bounding box generated from object detection may subsequently be used for recognition or object tracking. Current methods overlook these correlations, causing repeated computations across associated tasks. As a result, this redundancy wastes edge computational and storage resources, thereby reducing accuracy, increasing latency, and significantly degrading multi-task processing efficiency and system performance.

1.2 Simplified Scheduling Objectives Lack Overall Model Evaluation

Existing edge scheduling decisions are typically driven by the goal of maximizing resource utilization or optimizing the accuracy-latency trade-off. These single-objective strategies have inherent limitations. They prioritize single metric (accuracy [8-9], latency [10-11] or resource utilization [6-7]) during model scaling and scheduling, thereby failing to consider other critical metrics and give a comprehensive evaluation of the model performance and system environment. In real scenarios, sacrificing accuracy for high resource utilization or incurring delays in time-sensitive contexts could severely undermine the entire video analytics system. Furthermore, excessive pursuit of resource utilization may destabilize systems under high workloads, degrading task accuracy and failing to meet the demands of complex and dynamic video analytics scenarios.

To address these issues, we propose a performance scoring-driven model scaling and scheduling method tailored for edge video analytics serving. Our method specifically targets at two key challenges in multi-task edge video analytics serving: redundant computations and simplified scheduling objectives. For the incomplete evaluation challenge in existing approaches, we design a model performance scoring mechanism. In addition to resource utilization, the scoring mechanism incorporates input video complexity, task characteristics, and model performance metrics to generate a comprehensive evaluation score for driving model scheduling decisions. We further propose a dynamic model scaling and scheduling algorithm which employs the particle swarm optimization (PSO) algorithm [12, 13] to optimize resource allocation and model selection process.

The main contributions of our work are as follows:

(1) A model performance scoring mechanism tailored to guide scaling and scheduling decisions. To catch the key requirements and task characteristics of edge video analytics serving, we design a holistic scoring metric that integrates

(2) video complexity, task dependencies, model accuracy, and resource utilization. This metric serves as the foundation for collaborative optimization in edge environments.

(3) An elastic model scaling and scheduling method using particle swarm optimization. Using the comprehensive scores as the optimization objective, we develop a dynamic model scaling and scheduling strategy that optimizes resource allocation and model combinations through PSO. This method achieves a balance between real-time responsiveness, computational efficiency, and system stability under varying edge video analytics workloads.

(4) Implementation and validation of an edge video analytics system. We implement and evaluate our proposed method on real-world edge devices and several video analytics scenarios, demonstrating its effectiveness.

2 MOTIVATIONS

This section first introduces the technical background and characteristics of edge video analytics. Based on this, we identify the challenges of existing model scaling and scheduling methods.

2.1 Characteristics of edge video analytics

Unlike cloud-centric processing, edge video analytics serving exhibits the following characteristics:
(1) Limited Resources
Common edge devices operate under limited computational and storage resources, and this hardware disparity compels algorithm designs to balance model accuracy against resource efficiency.
(2) Latency-sensitive requirements
Applications such as autonomous driving and real-time surveillance demand millisecond-level responsiveness.
(3) Multi-task concurrency and dependency
In edge computing scenarios, multiple correlated video analytics tasks are often executed concurrently on the same edge node, sharing intermediate results or competing for limited resources.
(4) Dynamic environmental complexity
Real-time fluctuations exist in video stream quality (e.g., resolution, lighting), scene semantics (e.g., sudden changes in crowd density), and network conditions.
These inherent traits collectively define the core challenges in edge video analytics, driving research efforts toward cross-task collaboration, resource scheduling, and model optimization in dynamic edge environments.

2.2 Challenges on model scaling and scheduling

Model scaling and scheduling are pivotal for balancing computational efficiency and service quality in edge video analytics. Although prior works have improved resource utilization in static scenarios, they fall short in addressing the following challenges in edge video analytics.

Challenge 1: Redundant computations are caused by task correlations. Existing edge schedulers treat video analytics tasks (e.g., object detection, tracking) as isolated processes, ignoring their inherent dependencies. For instance, a fall detection task and a loitering detection task may process the same video frame independently, extracting overlapping features (e.g., human body bounding boxes). Existing approaches fail to reuse intermediate results across tasks, resulting in unsustainable overheads in multi-task deployments. A holistic scheduler must exploit task correlations to eliminate redundant computations while preserving accuracy.

Challenge 2: Single-objective scheduling has limitations. Most scheduling policies [5, 7, 10, 19] prioritize a single optimization metric. However, video analytics serving demands multi-dimensional optimization. As shown in Figure 1, high-accuracy models may violate latency constraints, whereas lightweight models often compromise accuracy in complex video scenarios. This underscores the need for a unified scoring metric that evaluates model performance across model accuracy, latency, and environmental adaptability.
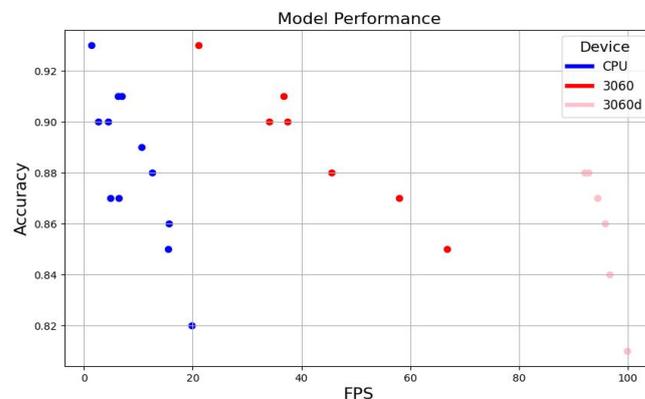

Figure 1 Accuracy-Latency Trade-off on Different Devices

These challenges collectively highlight the limitations of conventional model scaling and scheduling methods in handling task dependencies, multi-dimensional trade-offs, and environmental conditions.

2.3 Problem Formulation

In edge scenarios, quality of service (QoS) is primarily reflected in three aspects: system resource utilization, video analytics accuracy, and processing latency. Given the high real-time requirements in edge scenarios, our objective is to minimize the system latency, while ensuring system overall accuracy and resource utilization. Its formal definition is as follows:

$$min\ L_{TA} \tag{1}$$

$$\text{s.t.}\ Acc_i \geq \tau_i$$
$$U_{CPU}, U_{GPU} \geq U_{min} \tag{2}$$
$$M_u < M_t$$

Latency L_{TA} refers to the execution time from the input of a video frame to the output of the result. The resource utilization constraint is as follows: the CPU utilization U_{CPU} and GPU utilization U_{GPU} of system should exceed specified minimum thresholds U_{min}. The memory used M_u must be less than the system memory limit M_t. Additionally, we set different accuracy thresholds τ_i for different video analytic tasks, and the accuracy must be greater than the corresponding threshold.

To solve the problem above, we adopt a model performance scoring mechanism to evaluate the performance of different models under various constraints. Then, we conduct model scaling and scheduling based on these scores. The whole framework will be introduced in the next section.

3 FRAMEWORK OVERVIEW

We propose a dynamic performance scoring-driven model scaling and scheduling method. The overall framework architecture is illustrated in Figure 2.

As inputs, the framework receives (1) raw video frames from edge cameras or unprocessed video streams, (2) characteristics of video analytics tasks, including video analytics objectives (e.g., fall detection, face recognition), priority levels, and correlation requirements between tasks, (3) model performance profiles, which involves several key metrics: accuracy, latency and memory demands of video analytics models, and (4) real-time resource snapshot, including available GPU memory, CPU utilization, and network bandwidth.

Our framework consists of three core modules: the frame processing module, the model performance scoring module, and the model scaling and scheduling module. The frame processing module first extracts and filters key frames from the input video streams for subsequent analytics. Then it calculates the video complexity based on the number of target objects. The model performance scoring module dynamically evaluates each candidate model using the input video analytics task information (e.g., priority, QoS requirements), model information (e.g., model architecture, accuracy and latency profiles), and real-time system resource status (e.g., GPU memory availability). By aggregating multi-dimensional evaluations, this module calculates the comprehensive metric $TScore$. Finally, the model scaling and scheduling module makes scheduling decisions based on $TScore$. Leveraging the Particle Swarm Optimization (PSO) algorithm, this module identifies the optimal model combinations that maximize the $TScore$, thereby ensuring efficient model performance under constraints in Eq.2.

The core of our framework is the model performance scoring mechanism and the model scaling and scheduling method based on the PSO algorithm, which will be detailed in Section 4 and Section 5.
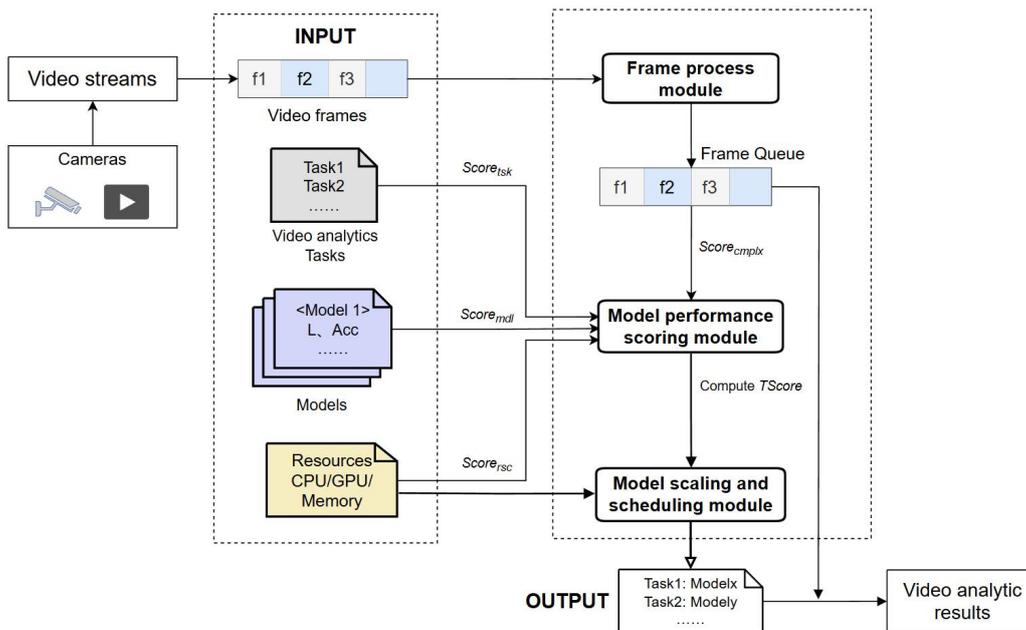

Figure 2 The Overview of Our Framework

4 MODEL PERFORMANCE SCORING

According to the optimization objectives in Section 2.3, in the edge video analytics system, the performance indicators of latency, accuracy, and system resource utilization are influenced by multiple factors. Therefore, in this section, we introduce a model performance scoring metric for edge video analytics, which integrates four critical factors: video complexity, which reflects the number of targets to be detected in the input frames and the degree of changes in the video background, task correlation, which is calculated based on the categories of video analytics tasks in the system and the feature similarity among different tasks, model performance, including the accuracy and latency of different model variants on different edge devices, and resource utilization, referring to the estimated usage of system resources (CPU, GPU, and memory) on the edge device.

4.1 Video Complexity Score

For each task, the quality of the input will indirectly affect the performance of models and thus influence the final analytics result. Frames with higher complexity require models with better performance for video analytics. Taking this impact into account, we propose a scoring metric $Score_{cmplx}$ to measure the complexity of videos, which takes into consideration factors such as the number of targets and the complexity of environment in the video. We calculate the video complexity $Complex_{obj}(vd)$ based on the number of targets in the video frame and the historical maximum number of targets. Difference function $DIFF(f_n)$ is used to calculate the difference between the current frame f_n and the previous frame f_{n-1}.

Based on the video complexity and the difference between frames. The video complexity score can be calculated as:

$$Score_{cmplx}(vd) = DIFF(f_n) \cdot \big(1 - Complex_{obj}(vd)\big) \tag{3}$$

4.2 Task Correlation Score

Usually, there exists a certain correlation in the analytical logic of video analysis tasks. For instance, both target detection and target tracking require the use of target bounding boxes in their respective process. Thus, we use a scoring metric to evaluate the correlation score $Score_{tsk}$ between tasks.

Let F_i be the feature vector of task i extracted by the analytics model. The value of $Score_{tsk}$ is calculated by defined task priority P_i and cosine similarity between two feature vectors:

$$Score_{tsk} = \sum_{i=1}^{N} \left(P_i \cdot \sum_{j=1,j \neq i}^{N} \cos\left(F_i, F_j\right) \right) \tag{4}$$

4.3 Model Performance Score

Since we focus on multi-task video analytics, each type of task has a corresponding set of models. For the same task, different models may vary in terms of accuracy and latency. To accurately quantify the performance of model, we design a model performance score $Score_{mdl}$ described in Eq.5, which integrates model accuracy and latency, aiming to identify the performance of model under the current system state.

$$Score_{mdl} = Acc_{mdl} \cdot e^{-k \cdot L_{mdl}} \tag{5}$$

where Acc_{mdl} represents the accuracy of video analytics models and L_{mdl} denotes the latency of each task. Besides, an exponential decay function is applied as a constraint to emphasize real-time requirements, with k serving as the decay coefficient.

4.4 Resource Utilization Score

Resources on edge devices are limited, so resource utilization efficiency has become one of the key factors in system optimization. Especially in multi-task scenarios, models of various tasks need to efficiently share and utilize the limited system resources. The resource utilization score $Score_{rsc}$, comprehensively takes into account the estimated resource consumption such as memory usage and the occupancy of CPU and GPU. can be obtained by:

$$Score_{rsc} = U_{Memory} \cdot U_{CPU} \cdot U_{GPU} \tag{6}$$

where U_{CPU} and U_{GPU} denote the CPU and GPU utilization of system, respectively. U_{Memory} represents the system memory usage rate, which can be calculated from the current used memory M_u and the model switching cost M_{mdl} as follows:

$$U_{Memory} = \frac{M_u + M_{mdl}}{M_t} \tag{7}$$

5 MODEL SCALING AND SCHEDULING

5.1 Problem Definition

The optimization problem in Section 2.3 has multiple optimization objectives. The process of finding the Pareto optimal solution is quite complex, and it is difficult to make real-time decisions. Combining all the scoring metrics discussed above, we define a comprehensive metric $TScore$, which integrates the key aspects of videos, tasks, models and resource factors. Then we re-define the problem of minimizing latency as the problem of maximizing $TScore$.

For edge video analytic system, $TScore$ is positively correlated with the accuracy and the system resource utilization (below the threshold), and negatively correlated with the system analytic latency. Therefore, the $TScore$ can be calculated by Eq.8:

$$TScore = \alpha \cdot Score_{cmplx} \cdot Score_{tsk} \cdot Score_{mdl} + \beta \cdot Score_{rsc} \tag{8}$$

incorporates two parameters, α for analytic models and β for system resource utilization, allowing it to adapt to various edge scenarios with differing optimization goals. And the problem can be formulated as:

$$\underset{mdl}{arg\ max} \sum_{tsk=1}^{N} TScore \#(9) \tag{9}$$

Our objective is to maximize the sum of model $TScore$ across all tasks through model scaling and scheduling, which serves as the system's comprehensive score.

5.2 Working Process

According to the definitions in Section 5.1, the working process of the model scaling and scheduling module is shown in Figure 3.
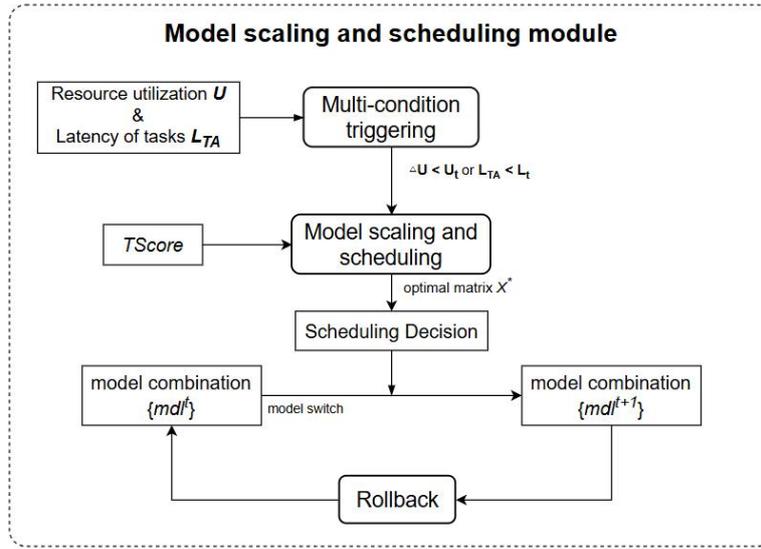

Figure 3 The Process of Model Scaling and Scheduling

As illustrated, this process consists of the following parts.

5.2.1 Multi-condition triggering.
The edge video analytic scenarios are highly dynamic, with the video complexity and system load changing in real time. In our framework, the appropriate timing for triggering the scaling and scheduling algorithm is particularly crucial. The multi-condition triggering mechanism determines when to evaluate model performance, recalculate, and perform model scaling and scheduling based on the score. In the edge scenario, two relatively important metrics are system resource utilization and the latency of video analytic tasks. The triggering conditions are:
(1) System resource utilization: The change ΔU in the utilization of CPU, GPU, or memory exceeds utilization change threshold U_τ or the utilization itself does not meet the interval value.
(2) Latency of video analytic tasks: The latency L_{TA} fluctuates and tend to be greater than threshold L_τ.
Meanwhile, we have set a triggering interval time, within which the models can only be scaled and scheduled at most once.

5.2.2 PSO-based model scaling and scheduling
The model scaling and scheduling module drive scheduling decisions based on the Particle Swarm Optimization (PSO) algorithm, aiming to maximize the system $TScore$. This section will introduce its definition and the solving process.
Particle representation. Given N video analytical tasks with M model variants per task, each particle is represented as a matrix $X_{M \times N} = (x_{ij})$, where element x_{ij} indicates the selection status of model j for video analytic task i. The swarm size is dynamically configured based on task load and problem complexity.
Velocity update mechanism. Particle velocity $V_{M \times N} = (v_{ij})$ determines search direction and step size, with element v_{ij} representing selection tendency for model x_{ij}. The velocity update rule integrates three components:

$$v_{ij}(t+1) = \omega_v \cdot v_{ij}(t) + c_1 r_1 [pb_{ij} - x_{ij}(t)] + c_2 r_2 [gb_{ij} - x_{ij}(t)] \tag{10}$$

where c_1 and c_2 are cognitive coefficient and social coefficient. r_1 and r_2 are uniform random variables for subsequent searching. Vector pb_{ij} represents the particle's historical best position, and vector gb_{ij} represents swarm's global best position, referring to the model scaling and scheduling consequence that achieves the maximum $TScore$ value in the current system state. ω_v is the adaptive inertia weight controlling global and local search balance, and it follows a linear decay scheme:

$$\omega_v = \omega_{init} - \frac{t}{T_{max}}(\omega_{init} - \omega_{end}) \tag{11}$$

The algorithm initializes with higher weight ω_{init} to enhance global exploration, and progressively decrease to ω_{end} for intensified local exploitation as iterations approach T_{max}.

Fitness Function. The optimization objective is to maximize the composite performance score $TScore$. Therefore, the fitness function is:

$$Fitness(X_{M \times N}) = \sum_{tsk=1}^{N} TScore \tag{12}$$

Scheduling Process. First, the PSO algorithm initializes the initial positions and velocities of particles in the particle swarm based on the analytical task and model information, and sets a series of parameters such as learning factors c and the number of iterations T_{max}.

Then, according to the model performance scoring module, the fitness values of each particle are calculated respectively. Then, through comparison, the historical optimal position of each particle and the historical optimal position of the swarm are updated. If the maximum number of iterations T_{max} is reached or the difference between two consecutive fitness values is less than the threshold, the loop terminates and the optimal position is output; otherwise, it continues to calculate the new fitness value for a new round.

Finally, the algorithm output matrix $X_{M \times N}^*$ based on the global best position. This matrix corresponds to the currently optimal model combination to be scheduled.

Once the PSO algorithm generates the model scheduling decision, the module preloads the selected models for each task and gradually redirects the input video frames from the current models to the new selected models in a data-driven manner until this process is complete.

5.2.3 Rollback

The rollback mechanism is designed to retain a snapshot of models before scheduling. When the scheduled models combination encounter an anomaly and fail to meet the requirements of edge video analytics, such as a prolonged decline in $TScore$, the system will automatically rollback to the previous models combination, and wait for the next time interval to perform new scheduling.

6 EXPERIMENTS

6.1 Implementation

Video analytic tasks. To verify the performance of the proposed framework, we selected common abnormal behavior detection tasks: intrusion detection [14], fall detection [15], loitering detection [16], and crowd detection [17]. We use $TScore$ as evaluation metrics. In intrusion detection task, we selected mAP50 as the accuracy metric. In fall detection task, we selected precision as the accuracy metric. In loitering detection task, we selected the Identity F1 score (IDF) as the accuracy metric. The IDF can be calculated by the Identity Precision (IDP) and Identity Recall (IDR) metrics:

$$IDF = \frac{2 \cdot IDP \cdot IDR}{IDP + IDR} \tag{13}$$

In crowd detection task, we selected the mean squared error (MSE) as the accuracy metric.

The abnormal behavior detection in four tasks was implemented based on the YOLOv8 model. YOLOv8 is an advanced object detection model that integrates an optimized neural network architecture, dynamic input adaptation, and multi-scale training techniques. For intrusion detection task, we used YOLOv8 to detect intruding pedestrians within the set target area. For fall detection task, we used YOLOv8 to recognize human postures in the frame and make fall judgments. For loitering detection task, we used YOLOv8 and DeepSORT [18] to locate and track pedestrians in the video, and determined loitering based on the movement trajectories and residence time. For crowd detection task, we used YOLOv8 for crowd counting and calculated the crowd density in the frame accordingly.

When training these video analytic models, we set the learning rate to 0.001, the batch size to 4, and the training epochs to be between 300 and 500. In the multi-condition triggering mechanism proposed in Section 5.2.1, we set the utilization change threshold U_τ and the latency threshold L_τ to 20% and 150 ms, respectively.

Workload. We selected three types of datasets as the workloads for video analytic tasks. For intrusion detection and fall detection task, we selected the Le2i Fall detection Dataset as the workload. This dataset was captured by real surveillance cameras and contains videos of various fall behaviors as well as video frames without pedestrians. For loitering detection and crowd detection task, we selected the MOT16 and MOT20 datasets [20] as the workloads. The MOT16 dataset includes pedestrian tracking in different scenarios, while the MOT20 dataset has a denser pedestrian scenario.

6.2 Baselines

We compared our method with the following frameworks:

(1) Static Model Deployment: a basic method where video analytic models are statically deployed on the experimental platform without dynamic adjustments based on real-time operating conditions.

(2) CerberusDet [5]: a unified multi-dataset object detection framework, employs an innovative architecture design and advanced feature fusion techniques to fully exploit the complementary characteristics across datasets, thereby enhancing detection accuracy.

(3) Learning-Based Query Scheduling and Resource Allocation (LQSRA) [6]: an adaptive mobile-edge-coordinated RL -based framework to handle unpredictable query arrivals and fluctuant available resources.

6.3 Performance Evaluation

We evaluate our approach based on accuracy and latency, and compute the $TScore$ accordingly. Experimental results are presented in Table 1 and Figure 3, respectively.

In terms of accuracy, our method achieves a 6.0% improvement compared to the static method but decreases by 6.7% and 5.4% compared to CerberusDet and LQSRA, respectively. As for latency, our method reduces latency by 5.0% compared to static deployment but incurs 1.9% and 5.2% latency increases compared to CerberusDet and LQSR. The static deployment method lacks any scheduling optimization, leading to poor accuracy in multi-task edge video analytic. CerberusDet enhances video analytics accuracy for highly similar tasks. However, for dissimilar tasks (e.g., object detection and object tracking), interference between multiple tasks reduces its average detection accuracy and increase latency of both tasks. LQSRA employs the PPO method for task scheduling and resource allocation, addressing resource shortages in high-load edge scenarios by selecting lightweight models.

During scheduling, our method incorporates accuracy as part of the optimization objective (Eq 5), performing model scaling and scheduling by maximizing the comprehensive $TScore$. Although our approach does not have advantages in single metrics like accuracy or latency, there are improvements of 41.9%, 17.3%, and 11.4% compared with static method, CerberusDet, and LQSRA, respectively. The $TScore$ calculation results demonstrate that it achieves the highest $TScore$ by optimizing both model performance and resource utilization of video analytic system.

Table 1 Experiment Results

Framework	Task	Accuracy	Average accuracy	Latency(ms)	$TScore$
Ours	Intrusion	0.73	0.70	262	0.88
	Fall	0.70			
	Loitering	0.67			
	Crowd	0.69			
Static	Intrusion	0.69	0.66	276	0.62
	Fall	0.68			
	Loitering	0.63			
	Crowd	0.63			
CerberusDet	Intrusion	0.79	0.75	257	0.75
	Fall	0.77			
	Loitering	0.71			
	Crowd	0.74			
LQSRA	Intrusion	0.75	0.74	249	0.79
	Fall	0.74			
	Loitering	0.72			
	Crowd	0.73			

As shown in Figure 4, in multi-task scenarios (with the number of tasks > 4), compared to CerberusDet and LQSRA, both accuracy and latency fluctuations of our method remain within 10%. The static method achieves lower video analytic latency under fewer tasks. As the number of tasks grows, it experiences significant latency fluctuations due to the lack of multi-task optimization. Compared to our approach, CerberusDet and LQSRA focus on single optimization objectives, thus demonstrating certain advantages in accuracy or latency.
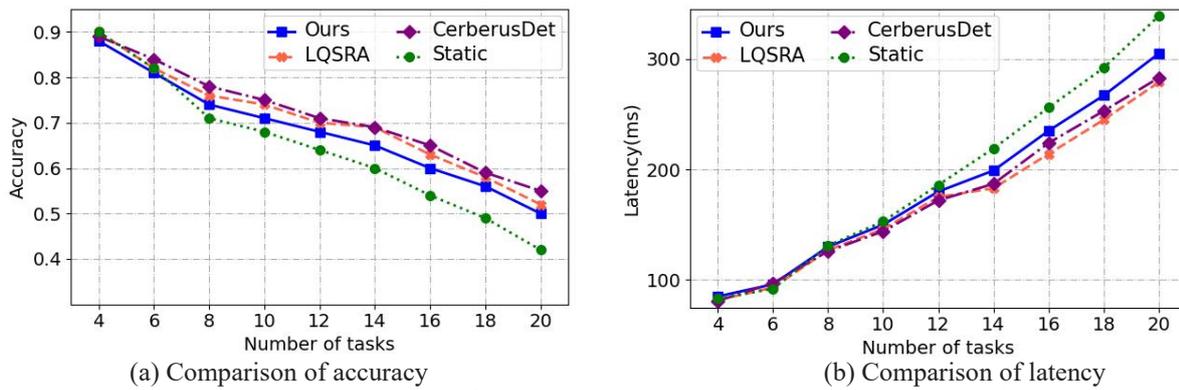
(a) Comparison of accuracy

(b) Comparison of latency

Figure 4 Comparison of Accuracy and Latency under Multi-Task Scenario

7 CONCLUSION

To address the challenges of redundant computations and single-objective scheduling in edge video analytics, we propose a model scaling and scheduling framework driven by a comprehensive performance metric *TScore* and PSO-based optimization. The *TScore* holistically evaluates video complexity, task dependencies, model accuracy-latency trade-offs, and resource utilization, enabling adaptive model selection. Experimental results on edge devices demonstrate that our framework achieves the best *TScore* performance. The elastic scheduling mechanism ensures real-time responsiveness and robust quality of service in scenarios like fall detection and crowd monitoring. The current framework requires manual calibration of *TScore* parameters for specific scenarios. Future directions include (1) self-adaptive parameter tuning via online meta-learning to enhance plug-and-play adaptability and (2) hardware-aware lightweight scheduling through neural architecture search (NAS)-enhanced PSO for heterogeneous platforms.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Bogdoll D, Nitsche M, Zöllner J M. Anomaly detection in autonomous driving: A survey. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 4488-4499.

[2] Tay N C, Connie T, Ong T S, et al. A review of abnormal behavior detection in activities of daily living. IEEE Access, 2023, 11: 5069-5088.

[3] Zhang Q, Sun H, Wu X, et al. Edge video analytics for public safety: A review. Proceedings of the IEEE, 2019, 107(8): 1675-1696.

[4] Xu R, Razavi S, Zheng R. Edge video analytics: A survey on applications, systems and enabling techniques. IEEE Communications Surveys & Tutorials, 2023, 25(4): 2951-2982.

[5] Tolstykh I, Chernyshov M, Kuprashevich M. CerberusDet: Unified Multi-Dataset Object Detection. arXiv preprint arXiv:2407.12632, 2024.

[6] Lin J, Yang P, Wu W, et al. Learning-based query scheduling and resource allocation for low-latency mobile-edge video analytics. IEEE Internet of Things Journal, 2023, 11(3): 4872-4887.

[7] Ma H, Li R, Zhang X, et al. Reliability-aware online scheduling for DNN inference tasks in mobile-edge computing. IEEE Internet of Things Journal, 2023, 10(13): 11453-11464.

[8] Wang C, Yang P, Hou J, et al. Dependence-aware multi-task scheduling for edge video analytics with accuracy guarantee. IEEE Internet of Things Journal, 2024, 11: 26970-26983.

[9] Ahmad S, Guan H, Friedman B D, et al. Proteus: A high-throughput inference-serving system with accuracy scaling. In: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, 2024(1): 318-334.

[10] Yang Y, Shen H, Tian H. Scheduling workflow tasks with unknown task execution time by combining machine-learning and greedy-optimization. IEEE Transactions on Services Computing, 2024, 17(3): 1181-1195.

[11] Zhou X, Liang W, Yan K, et al. Edge-enabled two-stage scheduling based on deep reinforcement learning for internet of everything. IEEE Internet of Things Journal, 2022, 10(4): 3295-3304.

[12] Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks, 1995, 4: 1942-1948.

[13] Jain M, Saihjpal V, Singh N, et al. An overview of variants and advancements of PSO algorithm. Applied Sciences, 2022, 12(17): 8392.

[14] Mei X, Zhou X, Xu F, et al. Human intrusion detection in static hazardous areas at construction sites: Deep learning–based method. Journal of Construction Engineering and Management, 2023, 149(1): 04022142.

[15] Chen B, Chen C, Hu J, et al. Computer vision and machine learning-based gait pattern recognition for flat fall prediction. Sensors, 2022, 22(20): 7960.

[16] Huang T, Han Q, Min W, et al. Loitering detection based on pedestrian activity area classification. Applied Sciences, 2019, 9(9): 1866.

[17] Song Q, Wang C, Jiang Z, et al. Rethinking counting and localization in crowds: A purely point-based framework. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021: 3365-3374.

[18] Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric. In: Proceedings of the IEEE International Conference on Image Processing, 2017: 3645-3649.

[19] Shao J, Zhang X, Zhang J. Task-oriented communication for edge video analytics. IEEE Transactions on Wireless Communications, 2023, 23(5): 4141-4154.

[20] Dendorfer P, Osep A, Milan A, et al. Motchallenge: A benchmark for single-camera multiple target tracking. International Journal of Computer Vision, 2021, 129: 845-881.

ANALYSIS OF EPIDEMIOLOGICAL CHARACTERISTICS OF PATIENTS WITH VENOUS THROMBOSIS IN EMERGENCY DEPARTMENT OF A 3A HOSPITAL IN SHENZHEN DURING COVID-19 EPIDEMIC

YuXiao Zhang

Department of Statistics and Data Science, Southern University of Science and Technology, Shenzhen 518055, Guangdong, China.
Corresponding Email: 12210960@mail.sustech.edu.cn

Abstract: Objective To explore the impact of COVID-19 on the incidence of thrombosis in Shenzhen, analyze the epidemiological characteristics of thrombotic diseases during the epidemic period, and predict the long-term incidence trend. Methods The data of thrombus patients in the emergency Department of Peking University Shenzhen Hospital from January 2017 to June 2024 were collected. R4.3.1 was used for statistical analysis. Results There were 115 patients with thrombus before the epidemic(January 2017 to December 2019), the male to female ratio was about 0.89:1. During the epidemic period(January 2020-Decenmber 2022), 190 patients with thrombosis were diagnosed, with a male/female ratio of 1.23:1. Since the epidemic(January 2023 to June 2024), 67 patients with thrombosis, the ratio of male to female is about 0.91:1. The number of patients in the 31~60 age group was the largest(51.30%,45.79%,44.78%, respectively). The main type of thrombotic disease was venous thrombosis (90.32%). Conclusion The COVID-19 epidemic significantly increased the incidence of thrombotic diseases, but with the popularization of vaccines and epidemic control, the incidence of thrombotic patients gradually recovered to the pre-epidemic level.
Keywords: Thrombosis; COVID-19; Epidemiological characteristic; Incidence rate

1 INTRODUCTION

The global emergence of the SARS-CoV-2 virus has brought about a wide range of health challenges beyond the well-known respiratory symptoms. Emerging evidence from various clinical and research settings indicates that SARS-CoV-2 infection may induce significant vascular endothelial injury. The endothelium, which lines the interior surface of blood vessels, plays a crucial role in maintaining vascular homeostasis. When compromised by the virus, it can lead to a cascade of pathological events, including inflammation and the activation of coagulation pathways. Consequently, this increases the risk of thrombotic events, posing a serious threat to patients' health.

Thrombotic disorders, encompassing conditions such as deep vein thrombosis, pulmonary embolism, and arterial thrombosis, represent clinically significant pathologies with substantial morbidity and mortality. These conditions can lead to long-term health complications, reduced quality of life, and increased healthcare costs. Given their severe impact, understanding the epidemiological patterns of thrombotic disorders is crucial for developing effective prevention strategies, especially in the context of the ongoing pandemic.

The COVID-19 pandemic has disrupted many aspects of healthcare delivery and disease incidence. It is essential to investigate whether and how the pandemic has influenced the incidence of thrombotic events. Such an analysis can provide valuable insights into the interplay between viral infections and thrombotic risk, and help inform public health policies and clinical practices moving forward.

This study aims to fill this knowledge gap by analyzing data from thrombosis patients presenting to the Emergency Department of Peking University Shenzhen Hospital. The dataset spans a period from January 2017 to June 2024, covering both the pre-pandemic and pandemic periods. By comparing the incidence rates of thrombotic events before and during the pandemic, we can evaluate the pandemic's impact on thrombotic incidence. Additionally, this study will project long-term trends to anticipate future challenges and guide the allocation of healthcare resources. The findings from this research will contribute to a better understanding of the complex relationship between SARS-CoV-2 infection and thrombotic disorders, ultimately aiding in the development of targeted prevention and treatment strategies.

2 MATERIALS AND METHODS

2.1 Data Sources

The study cohort comprised consecutively enrolled patients diagnosed with thrombotic disorders at Peking University Shenzhen Hospital Emergency Department from January 2017 through June 2024.

2.2 Data Collection

Electronic medical records were systematically reviewed for:

Demographic characteristics (name, sex, age)
Diagnostic timestamps
Thrombosis classification
Inclusion Criteria:
(1) Physician-confirmed thrombotic diagnosis
(2) Venous thrombosis or non-cerebral/non-coronary arterial/microvascular thrombosis
Exclusion Criteria:
Cerebral/coronary thrombosis cases secondary to hypertension, atherosclerosis, or diabetes mellitus
Statistical Analysis
Analyses were performed using R version 4.3.1. Categorical variables were expressed as proportions. Inter-group comparisons (pre-pandemic vs. pandemic vs. post-pandemic) employed χ^2 tests with Yates' correction and Fisher's exact tests where appropriate. A Bonferroni-adjusted significance threshold of $\alpha=0.025$ (two-tailed) was applied for multiple comparisons.

2.3 Results

Demographic Characteristics: The index case of pneumonia of unknown etiology was identified on December 8, 2019. The COVID-19 pandemic period in China was defined as January 2020 through December 2022. Accordingly, we categorized thrombotic cases as follows:
Pandemic cohort: All thrombosis patients presenting to the emergency department during January 2020-December 2022
Pre-pandemic cohort: Cases from January 2017-December 2019
Post-pandemic cohort: Cases from January 2023-July 2024

The pre-pandemic cohort comprised 115 thrombosis cases, representing 0.0160% of total emergency department visits during this period. The cohort showed:
Gender distribution: 54 males (46.96%) and 61 females (53.04%)
Age distribution peak: 31-60 years age group (51.30% of cases) (Table 1)

During the pandemic period, we identified 190 thrombotic cases (0.0393% of emergency visits), with:
Significant male predominance: 105 males (55.26%) vs 85 females (44.74%)
Highest proportion in 31-60 age group (45.79%) (Table 1)

The post-pandemic cohort included 67 cases (0.0141% of emergency visits), demonstrating:
Gender distribution: 32 males (47.76%) and 35 females (52.24%)
Predominance in 31-60 age group (44.78%) (Table 1)

Table 1 Distribution of Demographic Characteristics of Thrombosis Patients Visiting the Emergency Department of Peking University Shenzhen Hospital before, during and after the Pandemic

Before the Pandemic			During the Pandemic			After the Pandemic		
Demographic Characteristics	Number of Cases	Constituent Ratio (%)	Demographic Characteristics	Number of Cases	Constituent Ratio (%)	Demographic Characteristics	Number of Cases	Constituent Ratio (%)
Age Group (years old)			Age Group (years old)			Age Group (years old)		
< 30	4	3.48	< 30	21	11.05	< 30	7	10.45
31~60	59	51.30	31~60	87	45.79	31~60	30	44.78
61~90	43	37.39	61~90	80	42.11	61~90	29	43.28
>90	9	7.83	>90	2	1.05	>90	1	1.49
Gender			Gender			Gender		
Male	54	46.96	Male	105	55.26	Male	32	47.76
Female	61	53.04	Female	85	44.74	Female	35	52.24

Epidemiological Characteristics of Thrombotic Disorders
The predominant thrombotic subtype was venous thrombosis, accounting for 90.32% of all cases. Comparative analyses demonstrated:
Age-specific distribution: No statistically significant differences in thrombotic subtypes were observed across age groups ($\chi^2 = 3.0349$, P = 0.3863).
Sex-specific distribution: Similarly, no significant variations were noted between sexes ($\chi^2 = 1.9856$, P = 0.1588). However, stratification by sex revealed:
Venous thrombosis prevalence: 87.96% in males vs. 92.82% in females
Non-cerebral/non-coronary arterial and microvascular thrombosis prevalence: 12.04% in males vs. 7.18% in females
(See Table 2 for detailed distributions.)

Table 2 Types of Thrombosis in Patients Visiting the Emergency Department of Peking University Shenzhen Hospital

(n, %)

Type of Disease	Age Group (years old)				Gender		Total
	< 30	31~60	61~90	>90	Male	Female	
Thrombosis of the Venous System	31(96.88)	156(88.64)	139(91.45)	10(83,33)	168(87.96)	168(92.82)	336(90.32)
Arterial Thrombosis in Other Parts (Excluding Cerebral Arterial Thrombosis and Coronary Arterial Thrombosis) and Microcirculatory Thrombosis	1(3.12)	20(11.36)	13(8.55)	2(16.67)	23(12.04)	13(7.18)	36(9.68)

3 RESULT

3.1 Data Analysis

Regarding the change in the number of thrombosis patients over time, by comparing the period before the pandemic with that during the pandemic and using Fisher's exact probability test, the obtained p-value was 1.372×10^{-15} (the p-value was less than 0.025), indicating that the pandemic had a significant impact on thrombotic diseases. When comparing the period before the pandemic with that after the pandemic, the p-value obtained was 0.8054 (the p-value was greater than 0.025), which means that there was no significant difference in the number of thrombosis patients before and after the pandemic. The statistical data from Beishen Hospital shows that the proportion of the number of thrombosis patients in the total number of patients seeking medical treatment during the same period can be used to estimate the incidence of thrombotic diseases in the population of Shenzhen, thus indirectly reflecting the overall incidence of thrombotic diseases.

By comparing the number of patients with venous thrombosis during the pandemic with that before the pandemic and the number of patients with arterial thrombosis in other parts (excluding cerebral arterial thrombosis and coronary arterial thrombosis) and microcirculatory thrombosis, similarly, the p-value obtained was 0.284 (greater than 0.025), indicating that there was no significant difference in the impact of the COVID-19 pandemic on thrombotic diseases of the venous system and its impact on arterial thrombosis in other parts (excluding cerebral arterial thrombosis and coronary arterial thrombosis) and microcirculatory thrombosis.

Secondly, the number of thrombosis patients was counted in units of half a year. Bar charts were used to present the trends of the number of thrombosis patients over time, the number of patients with venous thrombosis over time, and the number of patients with arterial thrombosis in other parts (excluding cerebral arterial thrombosis and coronary arterial thrombosis) and microcirculatory thrombosis over time. Line charts were used to present the trend of the proportion of the number of thrombosis patients in the total number of patients seeking medical treatment during the same period over time and the trend of the proportion of venous thrombosis and arterial thrombosis in other parts (excluding cerebral arterial thrombosis and coronary arterial thrombosis) and microcirculatory thrombosis in thrombotic diseases over time (see Figure 1).
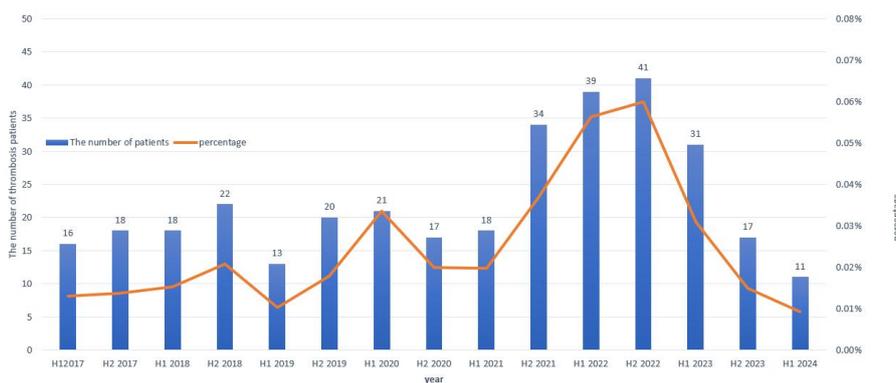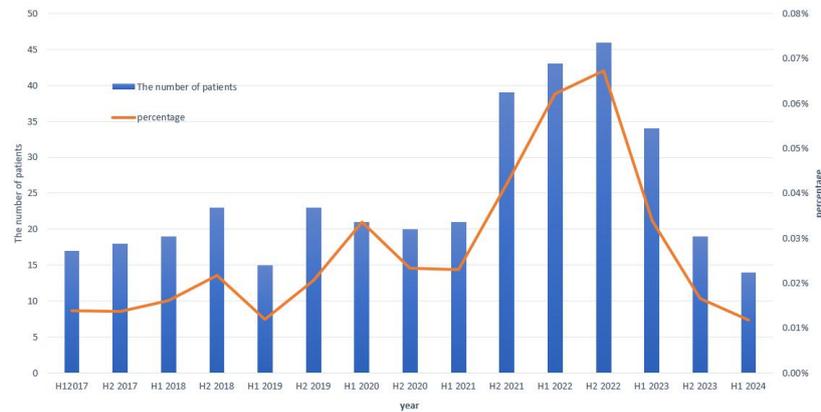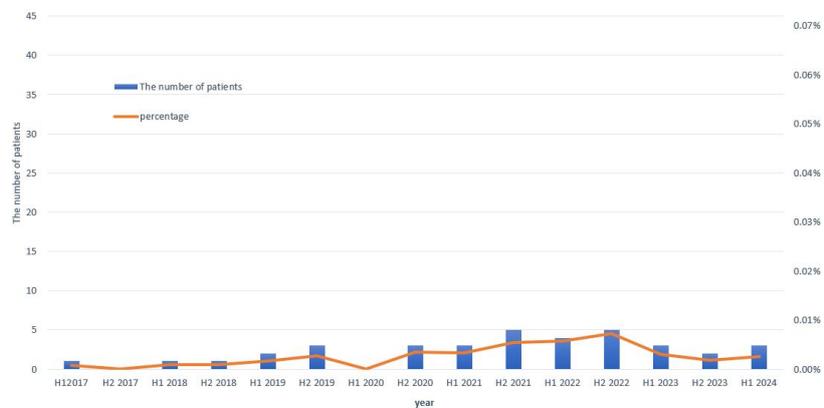

Figure: The Number of Patients with Venous Thrombosis in the Emergency Department of Beishen Hospital and the Proportion of the Number of Patients with Venous Thrombosis in the Total Number of Patients Seeking Medical Treatment during the Same Period over Time from the First Half of 2017 to the Second Half of 2024

(a)

Time - plot of the number of thrombus patients in the emergency department of Beishen Hospital from the first half of 2017 to the first half of 2024 and the proportion of the number of thrombus patients in the total number of patients treated during the same period.

(b)

Time - plot of the number of patients with arterial thrombosis and microcirculatory thrombosis in other parts except for cerebral arterial thrombosis and coronary arterial thrombosis in the emergency department of Beishen Hospital from the first half of 2017 to the second half of 2024, and the proportion of the number of such thrombus patients in the total number of patients treated during the same period.

(c)

Figure 1 The Proportion of Thrombosis Patients in the Emergency Department of Beishen Hospital to the Total Number of Thrombosis Patients over Time and the Number of Thrombosis Patients over Time from the First Half of 2017 to the Second Half of 2024

3.2 Comparison of the Incidence Rate of Thrombotic Diseases among the Study Subjects

It was calculated that during the pandemic from January 2020 to December 2022, the proportion of the number of thrombosis patients among the total number of patients seeking medical treatment during the same period was approximately 0.0405%. For the patients seeking medical treatment before the pandemic from January 2017 to December 2019, the proportion of the number of thrombosis patients among the number of patients seeking medical treatment during the same period was 0.0160%.

For the patients seeking medical treatment after the pandemic from January 2023 to June 2024, the proportion of the number of thrombosis patients among the number of patients seeking medical treatment during the same period was approximately 0.0201%. The proportion of the total number of thrombosis patients among the number of patients seeking medical treatment during the same period in the non-pandemic period was approximately 0.0173%. By comparing the proportion of the number of thrombosis patients among the number of patients seeking medical treatment during the same period in the two non-pandemic periods, the calculated p-value was 0.1653, which was greater than 0.05, indicating that the difference between 0.0160% and 0.0201% was not statistically significant.

When comparing the proportion of the number of thrombosis patients among the number of patients seeking medical treatment during the same period in the pandemic period and the non-pandemic period, the calculated p-value was less than 2.2×10^{-16}, indicating that there was a significant difference in this proportion ($p < 0.01$). Therefore, in the one-sided test, the proportion of the number of thrombosis patients among the number of patients seeking medical treatment during the same period increased significantly during the pandemic.

Conclusion of the Analysis of Image Data Information
Figure 1 shows that during the periods when the novel coronavirus was not prevalent (from January 2017 to January 2020 and from July 2023 to July 2024), the incidence rate of thrombosis patients in the Emergency Department of Peking University Shenzhen Hospital remained basically stable. Since the outbreak of the pandemic, the number of thrombosis patients has gradually increased and reached its peak in the second half of 2022. After 2022, the number of thrombosis patients gradually decreased and returned to the baseline level before the pandemic after the pandemic was brought under control.

4   DISCUSSION

Exploration of the reasons for the peak in the number of thrombosis patients shown in Figure 1 (a): Firstly, the COVID-19 pandemic fully erupted and continued to spread from 2021 to 2022. Correspondingly, the number of thrombosis patients increased from the first half of 2021 to the second half of 2022. It is reasonably speculated that the novel coronavirus may have led to an increase in the incidence rate of thrombotic diseases. Secondly, consider the factor of vaccination: Since China launched the COVID-19 vaccination for key groups on December 15, 2020, the growth rate of the number of thrombosis patients began to slow down in the first half of 2022 and gradually decreased in the second half of 2022. By the second half of 2023, the COVID-19 pandemic was basically under control, and the number of thrombosis patients stabilized at the baseline level before the pandemic.
The novel coronavirus has a significant impact on the incidence rate of thrombotic diseases, increasing the incidence rate of such diseases. After the vaccine took effect and the pandemic was brought under control, the incidence rate of patients with thrombotic diseases stabilized at the baseline value again. The impact of the novel coronavirus on thrombotic diseases is mainly manifested in the following aspects.
First, endothelial cell damage: SARS-CoV-2 directly invades endothelial cells, causing damage to vascular endothelium and the release of pro-inflammatory cytokines. Its spike protein can activate the alternative complement pathway [1, 2]. The damage caused by SARS-CoV-2 to vascular endothelial cells can widely trigger inflammatory reactions and abnormal coagulation functions, and may also lead to lesions or even dysfunctions in the lungs, heart, brain, kidneys, lower limb veins, skin and other parts. This is not only an important mechanism for causing damage to multiple organ functions in the human body but also the main reason for the significantly increased risk of venous thromboembolism (VTE) in COVID-19 infected patients [3].
Second, hypercoagulability of the blood: Due to the release of acute inflammatory reaction mediators caused by factors such as intravascular catheters or severe infections, the coagulation and fibrinolysis systems are affected through multiple pathways. There are factors such as a decrease in the levels of protein C and antithrombin-III (AT-III), an increase in the level of plasminogen activator inhibitor-I (PAI-1), an increase in coagulation factor VIII, an increase in fibrinogen, and neutrophil extracellular traps (NETs). Eventually, it leads to the activation of the coagulation cascade reaction and the inhibition of the fibrinolysis system, making the blood in a hypercoagulable state and further promoting the occurrence of VTE [4, 5].
Third, blood stasis: COVID-19 infected patients may experience insufficient fluid volume due to water loss caused by symptoms such as fever, diarrhea, and poor appetite, or blood stasis may occur due to reasons such as bed rest, obesity, shock, and coma, which further creates conditions for the formation of venous thrombosis and the occurrence of VTE [6, 7].

5   CONCLUSION

During the COVID-19 pandemic, the number of thrombosis patients increased significantly, and the risks of venous, arterial, and microvascular thrombosis increased significantly [8, 9, 10], which has attracted widespread attention from the global medical community. Through the analysis of the epidemiological characteristics of a total of 372 thrombosis patients in the emergency department of a large tertiary first-class general hospital during the COVID-19 pandemic and the non-pandemic period, as well as the data analysis of the proportion of the number of thrombosis patients among the number of patients seeking medical treatment during the same period before, during, and after the pandemic, it was found that before and after the pandemic, there were more female patients with thrombotic diseases than male patients, with male-to-female ratios of 0.89:1 and 0.91:1 respectively. During the pandemic, there were more male patients with thrombosis than female patients, with a male-to-female ratio of 1.24:1. The largest number of patients with thrombotic diseases was in the age group of 31 to 60 years old (accounting for 51.30%, 45.79%, and 44.78% respectively). The main type of thrombotic disease was venous thrombosis (accounting for 90.32%).
According to the results of this study, there is reason to believe that the novel coronavirus can significantly increase the incidence rate of thrombotic diseases. After the end of the COVID-19 pandemic, the proportion of the number of thrombosis patients among the number of patients seeking medical treatment during the same period tends to be stable. It is speculated that the long-term incidence rate of thrombotic diseases in the population will stabilize around the baseline value[11].
In conclusion, through the investigation and analysis of the population of thrombosis patients seeking medical treatment in the Emergency Department of Peking University Shenzhen Hospital, a preliminary prediction of the long-term incidence rate of the general population in Shenzhen was made. There is reason to believe that the novel coronavirus will not cause an increase in the long-term incidence rate of thrombotic diseases in the population.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] F Albani, L Sepe, F Fusina, et al. Thromboprophylaxis with enoxaparin is associated with a lower death rate in patients hospitalized with SARS-CoV-2 infection. A cohort study. EClinicalMedicine, 2020, 27: 100562.

[2] Z Zhai, C Li, Y Chen, et al. Prevention and treatment of venous thromboembolism associated with coronavirus disease 2019 infection: A consensus statement before guidelines. Thromb Haemost, 2020, 120(6): 937–948.

[3] EM Conway, N Mackman, RQ Warren, et al. Understanding COVID-19-associated coagulopathy. Nat Rev Immunol, 2022, 22(10): 639–649.

[4] JA Heit, MD Silverstein, DN Mohr, et al. The epidemiology of venous thromboembolism in the community. Thromb Haemost, 2001, 86(1): 452–463.

[5] A Shaik, Q Chen, P Mar, et al. High blood viscosity in acute and recent COVID-19 infections. Clinical Hemorheology and Microcirculation, 2022, 82(2): 149–155.

[6] M Begbie, C Notley, S Tinlin, et al. The Factor VIII acute phase response requires the participation of NFkappaB and C/EBP. Thromb Haemost, 2000, 84(2): 216–222.

[7] BK Manne, F Denorme, EA Middleton, et al. Platelet gene expression and function in patients with COVID-19. Blood, 2020, 136(11): 1317–1329.

[8] JM Connors, JH Levy. COVID-19 and its implications for thrombosis and anticoagulation. Blood, 2020, 135(23): 2033–2040.

[9] JD Mcfadyen, H Stevens, K Peter. The emerging threat of (micro)thrombosis in COVID-19 and its therapeutic implications. Circ Res, 2020, 127(4): 571–587.

[10] AC Walls, YJ Park, MA Tortorici, et al. Structure, function, and antigenicity of the SARS-CoV-2 spike glycoprotein. Cell, 2020, 181(2): 281–292.

[11] Wang Y. Efficient Adverse Event Forecasting in Clinical Trials via Transformer-Augmented Survival Analysis. Preprints, 2025. DOI:10.20944/preprints202504.2001.v1.