# Industrial Tool Recognition and Classification System under Intelligent Manufacturing

**Zhanhong Liang, Shaoyong Hong***
*School of Artificial Intelligent, Guangzhou Huashang College, Guangzhou, China*
**Corresponding Author*

**Abstract: With the rapid development of industrial intelligence, traditional tool recognition technologies are increasingly unable to meet the requirements of high efficiency and precision in modern manufacturing. To overcome these limitations, this paper introduces a novel industrial tool recognition approach that combines convolutional neural networks (CNN) with computer vision to design the "Smart-Eye Scout" system. The proposed system processes real-time image data collected by industrial cameras, enabling robust recognition of tools in complex production environments while supporting visual interaction functions. The experimental evaluation demonstrates that the system achieves high recognition accuracy and real-time performance under conventional industrial conditions, although further optimization is necessary to address challenges such as extreme lighting and unfamiliar tool types. The main contribution of this study lies in the integration of deep learning and real-time computer vision into a practical recognition platform, providing an effective solution for industrial intelligence. This work offers significant potential for enhancing production efficiency and advancing intelligent manufacturing.**

**Keywords: Industrial Tool Recognition; Deep Learning; Convolutional Neural Networks (CNN); Computer Vision; Intelligent Manufacturing**

## 1. Introduction

### 1.1 Research Background and Significance

The wave of Industry 4.0 is sweeping across the globe, reshaping industrial landscapes through intelligence, digitalization, and networking. Emerging models such as flexible production lines and unmanned workshops have become mainstream, driving both industrial upgrades and economic growth. However, traditional manual tool identification methods are showing clear limitations during this rapid transformation. For example, errors in robotic arms on automotive assembly lines can disrupt production rhythms, while the low efficiency and poor accuracy of manual identification in 3C product assembly fail to meet the high demands of precision manufacturing. Consequently, automated tool recognition systems have become an inevitable choice. Such systems are capable of identifying tool status and position in real time, optimizing production processes, and reducing changeover times. In addition, they enhance quality control, provide timely alerts on tool abnormalities, and replace human labor in continuous operations[1]. These advantages significantly improve efficiency, reduce costs, and strengthen corporate competitiveness, thereby offering strong support for the intelligent transformation of modern manufacturing.

### 1.2 Limitations of Traditional Methods and Advantages of Deep Learning

Traditional industrial tool identification methods, such as template matching and feature engineering, show significant limitations in complex industrial scenarios. Template matching relies on pre-built template libraries; however, variable workshop lighting can cause tool reflections and shadows, which distort image features and reduce recognition accuracy. In addition, when tools appear at different angles and postures, deviations between contours and templates frequently lead to misjudgments.

Feature engineering, which depends on manually extracted features, also struggles to cope with industrial factors such as oil stains and wear, resulting in degraded recognition performance. It further requires extensive manual parameter tuning and lacks versatility, making it difficult to meet industrial demands.

By contrast, deep learning with automatic feature extraction and end-to-end learning has proven highly effective in complex environments. In industrial tool identification, convolutional neural networks (CNNs) play a central role. Convolutional layers capture local tool features[2], while pooling layers compress them to enhance robustness against environmental variations. The end-to-end architecture of deep learning models enables raw images to be used directly as input, with recognition results generated as output, thereby simplifying the workflow, avoiding error accumulation, and improving performance.

Furthermore, deep learning models demonstrate strong generalization capabilities. A model trained on one dataset can be rapidly adapted to new industrial scenarios through fine-tuning, which significantly improves the flexibility and practicality of tool recognition systems. Similar to image super-resolution, where CNN architectures are selected according to device load capacity and execution speed to balance performance and efficiency, deep learning offers a more efficient and reliable technical pathway for industrial tool identification.

### 1.3 The Purpose and Innovation of This Study

In today's era of accelerating industrial production intelligence, the precision and efficiency required for industrial tool recognition have become a critical constraint that significantly impacts production performance. Similar to the challenges faced by on-load tap changers in conventional transformers during fault identification, traditional industrial tool recognition methods also exhibit numerous issues under complex and diverse scenarios. These problems pose severe obstacles to the smooth operation of production. This study focuses on this crucial and challenging problem, aiming to develop an industrial tool recognition system with high accuracy and real-time characteristics.

The objective of this paper is to achieve accurate and efficient identification of industrial tools. In complex industrial environments, high-accuracy recognition prevents misjudgment-related issues, and real-time performance enhances production flow, meeting modern industrial requirements. The innovations of this study lie in algorithmic optimization, system architecture, and data augmentation.

In terms of algorithm optimization, we enhance deep-learning methods by customizing CNNs for industrial tool identification. Inspired by transformer fault-feature extraction, specific algorithms are applied to tool images. Adjustments to network structure and layer parameters make the model more sensitive to tool features and more robust against lighting and posture interference. An adaptive learning rate mechanism, similar to those addressing gradient explosion, dynamically adjusts the rate, accelerates convergence, improves recognition accuracy, and reduces training time, thus meeting industrial real-time needs.

At the system architecture level, we construct a new framework that breaks free from traditional constraints. A modular design philosophy is adopted to decouple functional modules such as image acquisition, data pre-processing, model recognition, and result output, significantly enhancing scalability and maintainability. When new features need to be added or existing functions upgraded, individual modules can be modified without affecting overall operation. In addition, by introducing a distributed computing architecture, we achieve parallel processing across multiple nodes, fully leveraging hardware resources to boost system performance. Even in complex scenarios involving large-scale industrial data processing, the system remains efficient and stable, providing robust and reliable support for industrial tool recognition.

## 2. Related Technologies and Algorithms

### 2.1 Convolutional Neural Network (CNN) Basics

In the "Smart-Eye Scout" industrial tool recognition system, deep learning plays a central role, with convolutional neural networks (CNNs) serving as the key technical support for system operation. A CNN is mainly composed of convolutional layers, pooling layers, and fully connected layers, which work together to achieve efficient identification of industrial tools.

The convolutional layer is the core module of a CNN. During industrial tool recognition, the convolutional layer slides a kernel across the image, multiplying and summing it with pixel values to extract local features. For example, in images of industrial tools, convolutional layers can effectively capture basic features such as tool edges and textures. As multiple convolutional layers are stacked, the model progressively extracts hierarchical features—from simple edge information to composite shape features—providing rich and representative feature information for subsequent recognition tasks.

The pooling layer, usually placed after the convolutional layer, downsamples feature maps. Max pooling, for instance, selects the maximum value in a local region, retaining key features while reducing data size and computational complexity. This improves model efficiency and adaptability to variations in tool position and angle, ensuring stable recognition. The fully connected layer at the end of a CNN integrates the features obtained from pooling layers. After flattening, the features are transformed into a one-dimensional vector. Neurons in this layer connect to all previous ones, applying weight matrices to generate outputs such as tool categories or detection results, thereby enabling precise classification.

Before the adoption of deep learning, traditional industrial tool recognition relied on manually extracted features combined with algorithms such as SVMs and random forests, which required substantial expertise and time. In complex scenarios, these methods often failed. CNN-based deep-learning models, which learn directly from raw images, have become the mainstream approach, improving recognition accuracy and robustness and supporting the intelligent transformation of industrial production.

## 2.2 Application of Deep Learning in Vision

In the digital era, the explosive growth of data, further fueled by cloud computing, calls for effective extraction of valuable information[3]. Image visual recognition technology emerges as a key solution, with deep learning playing a pivotal role, especially in computer image visual recognition. Neural networks in deep learning, modeled after human brain neurons, form the basis for processing and analyzing input data. As technology progresses, research on neural network architectures and algorithms is becoming increasingly sophisticated.

Deep learning advancements have greatly driven industrial tool recognition. Traditional methods, relying on manual feature design and classical machine learning, demand specialized knowledge and perform poorly in complex industrial scenarios like lighting changes and occlusions, often failing to meet accuracy requirements.

In contrast, deep - learning models such as convolutional neural networks, widely applied in recent years, have made significant progress. These models automatically extract features from vast image data, eliminating the need for manual feature engineering. As a result, they maintain high recognition accuracy even under challenging conditions. This innovation has significantly increased the intelligence of industrial production, meeting the industry's demand for high - precision and efficient tool recognition.

## 2.3 Target Detection and Classification Algorithm

This paper uses the YOLOv8 algorithm. The YOLO object detection series, with its single forward - pass design, excels in industrial vision. Integrating multi - scale feature extraction, it efficiently inspects industrial components of varying sizes. Trained on extensive industrial data, the system adapts well to complex conditions. To meet industrial needs, researchers propose optimizations. Data augmentation tailored to industrial images boosts the model's generalization. Model compression enables efficient running on embedded devices for real-time operation. Adjusting network structures and parameters fits specific industrial targets, while combining detection models enhances reliability, supporting industrial quality inspection. In electronics manufacturing, the optimized system offers high - accuracy and real - time processing, robustly handling lighting changes and occlusions. As industrial intelligence progresses, it requires further improvement in handling special conditions and recognizing tiny targets to satisfy evolving industrial inspection demands[4].

# 3. System Framework Design

The image acquisition module is responsible for capturing tool images in industrial scenarios, typically through high-resolution industrial cameras. As the core device for image acquisition, the selection of industrial cameras has a significant impact on subsequent visual inspection results, and multiple factors must be considered comprehensively.

In terms of resolution, it should be determined according to the precision requirements of the inspection task. For example, when inspecting tiny electronic components with a precision requirement of 0.01 mm/pixel and a field of view of 5 × 5mm, the single-directional resolution must reach 500 pixels. Considering edge distortion and system stability, a multiplier of four or higher is usually applied, meaning the single-directional resolution reaches 2000 pixels. The corresponding camera resolution is approximately four million pixels, which indicates that a five-megapixel camera can meet the requirement.

The frame rate is closely related to the ability to capture dynamic targets. When detecting fast-moving industrial components, such as objects on a production line moving at speeds of 1 m/s, the camera must complete the shooting action within 1ms. This requires a frame rate of at least 1000 fps to ensure clear images.

Pixel size also affects image quality and generally ranges from 3μm to 10μm. Although smaller pixel sizes increase manufacturing difficulty, they can improve resolution, which requires a balance between performance and practicality. Furthermore, the spectral response characteristics of the camera determine its sensitivity to different wavelengths of light, typically ranging from 350 nm to 1000 nm. If the detection scene involves infrared light, the camera must either support infrared detection or be configured with its infrared filter removed. The key parameters influencing industrial camera selection, including resolution, frame rate, pixel size, and spectral response, are summarized in Figure 1, providing a clear reference for practical applications.

**Figure 1. Industrial Camera Equipment**

After setting the number of samples collected by each type, the industrial phase will collect data for the placed items (the specific data set is shown in Figure 2 and Figure 3 below), and set another data set to store non-industrial tools to avoid identification errors.

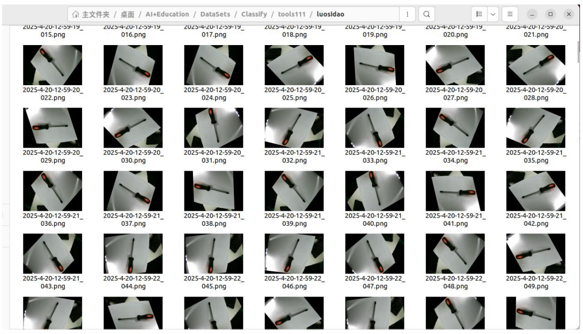

**Figure 2. Shows the Results of Screwdriver Data Acquisition**

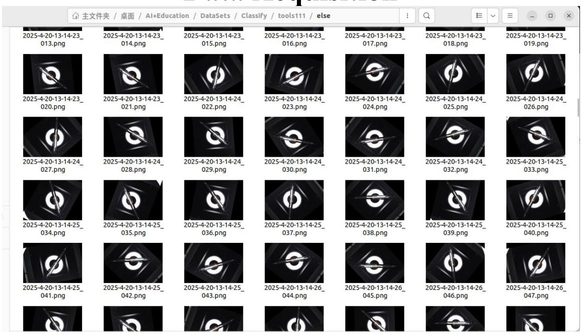

**Figure 3. Shows the Results of the Data Set for Storing Other Items**

Subsequently, the training set, validation set, and test set were divided according to a ratio of 7:1:2. The training set accounts for 70%, primarily serving to provide the model with ample data for parameter learning, enabling it to capture various characteristic patterns in industrial tool images. The validation set makes up 10%, used during the training process to evaluate the performance of the model at different stages, assist in adjusting hyper-parameters (such as learning rate and number of network layers), prevent overfitting, and ensure good adaptability of the model to new data. The test set accounts for 20%.

The data preprocessing module is essential in the industrial tool recognition system. It standardizes raw images, providing quality data for training and enhancing model generalization. For size normalization, as tool images vary, they are resized to common dimensions (e.g., 224 × 224 or 256 × 256 pixels) using bilinear or bicubic interpolation to avoid distortion. Bicubic interpolation compresses large images to preserve edges, while bilinear interpolation enlarges small ones. Grayscale or RGB conversion depends on the model and task[5].

Some models perform better with grayscale images, using a weighted average method:

$$Gray = 0.299 * R + 0.687 * G + 0.114 * B \quad (1)$$

Equation (1) represents the standard luminance conversion formula recommended by the ITU-R BT.601 standard. It reflects the different sensitivities of the human visual system to red, green, and blue light, with green contributing the most to perceived brightness, followed by red, and then blue. By applying this transformation, RGB images are converted into grayscale while preserving essential structural and brightness information.

Converting RGB images to grayscale can reduce data dimensions and lower computational complexity while retaining key image features, making it suitable for detection scenarios that are sensitive to brightness and edges. If the model needs to use color information to distinguish tools, the RGB format should be maintained to ensure complete color characteristics.

Data augmentation is a critical method for enhancing the model's generalization ability. Random rotation can simulate different placement angles in real-world scenarios; lighting adjustments can change brightness, contrast, and saturation to simulate complex workshop environments; additionally, random cropping and noise injection can expand dataset diversity, prevent overfitting, and ensure stable recognition performance under various industrial disturbances.

Model Framework(Figure 4): The model adopts a typical CNN architecture for image classification. Starting with the input layer that receives raw image data, it proceeds through multiple functional layers. The model contains two convolutional layers (Convolution Layer 1 and 2), which extract local features via kernel sliding. Each convolutional layer is paired with a pooling layer (Pooling Layer 1 and 2) for downsampling, reducing data size while retaining key features and enhancing robustness. After convolution and pooling, a flattening layer converts the multi - dimensional feature map into a 1D vector, feeding it into two fully connected layers.

Fully Connected Layer 1 further refines features, boosting nonlinear representation, while Fully Connected Layer 2 maps features to classification categories. Using the Softmax function, it outputs prediction probabilities for each category, completing the classification task. This framework, with convolutional feature extraction, pooling for dimensionality reduction, and fully connected classification, effectively addresses image classification, showing strong adaptability and accuracy with complex data.

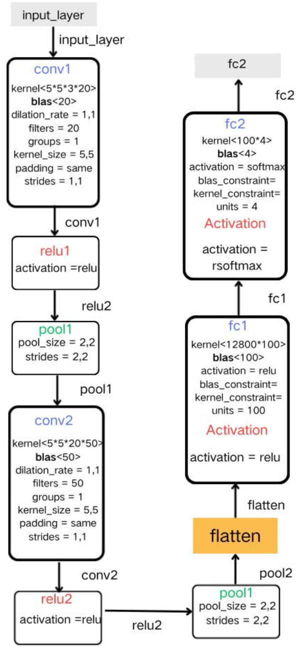


**Figure 4. Model Framework Diagram**

During the testing process, the system page connects to the industrial camera. When the user places the item to be identified within the lens range, the system rapidly performs recognition. As shown in Figure 5, a large blue font displaying "Image Classification Tag: qianzi" appears in a prominent position, enabling the operator to quickly capture the key identification information and clearly confirm that the current tool is a pair of pliers. In contrast, Figure 6 shows the classification result "else," indicating that the detected object does not belong to the predefined tool categories.

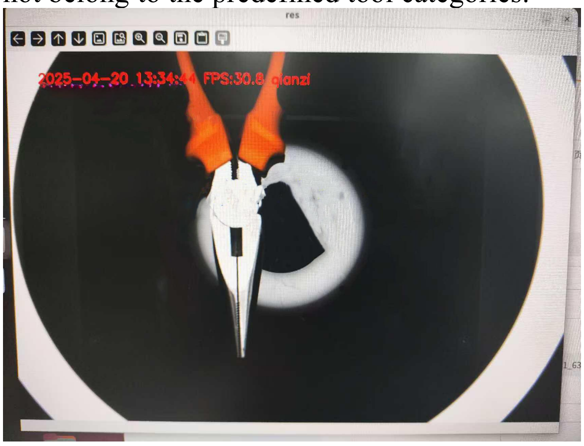


**Figure 5. Tool Test Results**

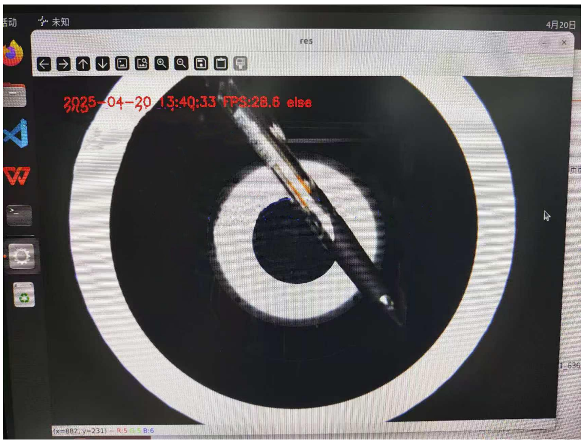


**Figure 6. Non-Tool Detection Results**

In addition, the system will present the current time and frame rate information in real time. The operator only needs to tap the space bar to record the experimental data, which can be used for subsequent evaluation and provide a basis for optimizing the system.

## 4. Experimental Process and Visualization

### 4.1 Model Training Process

Training Environment Configuration: The system's hardware consists of several key components. At the core is the intelligent edge platform, equipped with an i5 processor, 1T solid-state drive, and 16G memory, responsible for computation and data storage. Extensive USB, HDMI, and other common interfaces are reserved to facilitate the connection of external devices. An industrial camera and an external camera handle image acquisition tasks, with the external camera featuring specific resolution parameters. The visual system regulator adjusts visual system parameters, while the visual dark box creates a stable lighting environment to ensure accurate image collection. On the software side, Ubuntu 22.04.5 LTS is used as the operating system, providing a basic runtime environment. AI software components include Anaconda for environmental management, Python programming language, TensorFlow and Pytorch deep learning frameworks, OpenCV computer vision library, Jupyter notebook development environment, QT application development framework, as well as Streamlit and Gradio for building user interfaces, all supporting the intelligent functionality of the system[6].

Training Parameter Settings: The initial learning rate is set to 0.001. This crucial parameter controls model - parameter update steps. Based on experiments and historical training files, this rate enables rapid early - stage convergence and avoids non - convergence or oscillation caused by a high rate.The number of training epochs is set to 120 (Figure 7). Analyzing historical data, model performance stabilizes around 120 epochs. Further increasing epochs may cause overfitting, so 120 is chosen as the training end - point.

- Epoch 112/120 : - acc: 1.0000 - loss: 0.0202 - val_acc: 0.9957 - val_loss: 0.0036
- Epoch 113/120 : - acc: 0.9971 - loss: 0.0183 - val_acc: 1.0000 - val_loss: 0.0012
- Epoch 114/120 : - acc: 1.0000 - loss: 2.2352e-08 - val_acc: 0.9957 - val_loss: 0.0047
- Epoch 115/120 : - acc: 0.9986 - loss: 0.0037 - val_acc: 0.9957 - val_loss: 0.0093
- Epoch 116/120 : - acc: 1.0000 - loss: 6.7800e-07 - val_acc: 0.9957 - val_loss: 0.0089
- Epoch 117/120 : - acc: 1.0000 - loss: 7.4171e-05 - val_acc: 0.9957 - val_loss: 0.0074
- Epoch 118/120 : - acc: 1.0000 - loss: 3.7028e-06 - val_acc: 0.9957 - val_loss: 0.0074
- Epoch 119/120 : - acc: 0.9952 - loss: 0.0220 - val_acc: 0.9913 - val_loss: 0.0444
- Epoch 120/120 : - acc: 1.0000 - loss: 0.0047 - val_acc: 0.9870 - val_loss: 0.0523
- 2025-04-30 12:07:08,198 - [line:108 ] - INFO : [ 200312 ]

**Figure 7. Model Training Process**

### 4.2 Model Testing and Evaluation

Test set samples are selected based on the i.i.d. principle, separate from the training set to objectively evaluate model generalization. Stratified sampling during data partitioning maintains consistent data distributions between sets, crucial for multi - class classification to balance class-sample ratios and prevent bias[7].Test set construction focuses on comprehensive scene coverage. In computer vision, it includes standard images and samples under extreme lighting, multi angle, and multi - resolution conditions[8]. In natural language processing, it encompasses texts from diverse fields, language forms, and lengths, providing a realistic evaluation benchmark[9].

Model testing adheres to strict protocols. The test set is partitioned by criteria, and the trained model runs batch predictions on test samples with all results recorded. Task specific metrics are then used for evaluation. For classification models, a multi dimensional evaluation system is key. Classification accuracy assesses overall prediction correctness, while the misclassification rate highlights errors[10]. The category recall rate measures sensitivity to specific categories, and the F1 score balances precision and recall. In regression tasks, the mean squared error quantifies prediction bias, and the mean absolute error reflects error magnitudes directly. These metrics form a comprehensive framework for objective model evaluation. Test results are shown in Figure 8.

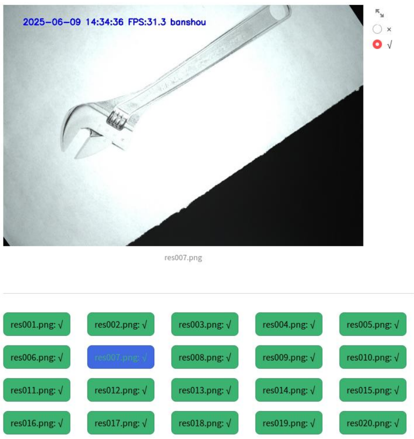


**Figure 8. Model Detection and Evaluation Diagram**

## 5. Conclusion

This study developed an industrial tool recognition system connected to industrial cameras for automated tool identification. The system labels recognition results in real time on a visual interface and displays key parameters like time and frame rate for data recording. Experimental results show high accuracy in conventional industrial settings, with acceptable image processing times. The study's main contributions in industrial tool recognition are two - fold: it presents a complete model training and testing process as a reference for related development, and its practical application reduces manual identification costs and boosts production automation. However, the system's performance is limited under extreme lighting, and its recognition accuracy for similar - looking and new tools needs improvement. Future research will focus on optimizing algorithm structures and training strategies to enhance the system's adaptability in complex environments.

## Acknowledgments

This work was supported by Guangzhou Huashang College Research Mentor Program (2021HSDS14), Guangzhou Huashang College Integrated Curriculum Project (HSRHKC202471).

## References

[1] Amelia E.H. Bridges; Eleanor Cross; Kyran P. Graves; Nils Piechaud; Antony Raymont; Kerry L. Howell. Practical application of artificial intelligence for ecological image analysis: Trialling different levels of taxonomic classification to promote convolutional neural network performance. Ecological Informatics, 2025, 88 103146-103146.

[2] Chen Peiran, Liang Detian, Liu Bin. A method for river and lake water level recognition based on improved convolutional neural network. Zhejiang Hydrotechnics, 2025, 53(02): 18-24.

[3] Yu Xinna. Monitoring and Analysis of Industrial Manufacturing Structure Change Trends Based on Big Data Tools. Enterprise Reform and Management, 2025, (13): 139-141.

[4] Jie Dengfei, Shen Hengwei, Li Jiajun, et al. Study on pig body temperature detection method based on infrared and RGB image registration. China Animal Husbandry Journal, 2025, 61(05): 410-418.

[5] Gao Zhigang. Adaptive fault recognition method for on-load tap changer of transformer based on improved convolutional neural network. Electrical Engineering Technology, 2024, (20): 66-70.

[6] Guo Junyu, Miao Yuhang, Fang Zixuan. Research on computer image vision recognition technology and application based on deep learning. Portrait Photography, 2025, (04): 183-184.

[7] Ye Xiaoli, Zhao Yuhang, Xu Qiming, et al. Integrated development of electrical equipment automation and quality control under the background of Industry 4.0// China Electromechanical Equipment Engineering Association. Proceedings of the Academic Exchange on Electromechanical Technology and Seminar on High-Quality Development of Scientific and Technological Talents. Wenzhou Yongbang Machinery Co., Ltd.; Wenzhou Guantian Technology Co., Ltd., 2025: 129-135.

[8] Wei Xiang, Chen Xi, Wang Feng, et al. Application of convolutional neural network in image super-resolution algorithms. Command Information Systems and Technology, 2021, 12(05): 43-47.

[9] Sun Huyi. Application of industrial robot vision recognition technology in assembly production. Mass Standardization, 2025, (04): 158-160.

[10] Zhuang Jian, Zhang Jing, Xu Yuwen. Deep Learning Image Recognition Technology. Beijing: China Machine Press, 2020: 286.