



A YOLOv8-based Static Gesture Recognition System for Contactless Human-computer Interaction in Public Service Scenarios

Jiaming Shen

Hohhot No.2 High School, Hohhot 010000, Inner Mongolia, China.

How to cite this paper: Jiaming Shen. (2025) A YOLOv8-based Static Gesture Recognition System for Contactless Human-computer Interaction in Public Service Scenarios. *Advance in Information Technology and Computer Science*, 2(1), 6-13. DOI: 10.26855/aitcs.2025.06.002

Received: July 21, 2025

Accepted: August 11, 2025

Published: September 2, 2025

***Corresponding author:** Jiaming Shen, Hohhot No.2 High School, Hohhot 010000, Inner Mongolia, China.

Abstract

With the increasing prevalence of public self-service terminals, traditional touch-based human-computer interaction (HCI) methods face significant challenges in terms of hygiene, efficiency, and accessibility. This study focuses on the need for contactless interaction in public settings and proposes a static gesture recognition system based on the YOLOv8 object detection framework. The system supports natural gesture-based control for commands such as “Confirm,” “Cancel,” “Scroll Up,” and “Scroll Down.” A hybrid dataset combining a public benchmark (Ha-GRID) and a custom-built gesture image set was constructed to train the model. Leveraging YOLOv8's efficient architecture, the system implements an end-to-end recognition-to-feedback pipeline optimized for real-time performance on macOS devices. Experimental results demonstrate strong performance in terms of accuracy (92.6%), inference speed (38.7 FPS), and mean average precision (mAP@0.5 = 84.3%), surpassing conventional models. Finally, limitations in generalization, dynamic gesture recognition, and customization are discussed, along with future directions including dynamic modeling and multimodal integration to enhance adaptability and intelligence.

Keywords

YOLOv8; Gesture recognition; Contactless interaction; Human-computer interaction; Public service system; Edge deployment

1. Introduction

With the rapid development of artificial intelligence and computer vision technologies, human-computer interaction (HCI) is undergoing a transition from traditional physical contact operations to more natural and intuitive multimodal interactions. Especially in public service systems, such as hospital registration terminals, subway ticket machines, and government service terminals, contactless interaction has gradually become an important trend due to its high hygiene and safety. In this context, vision-based gesture recognition is widely regarded as one of the most feasible and universal methods of natural interaction. Compared with voice recognition, gesture interaction has obvious advantages in noisy, privacy-sensitive, or inconvenient-to-speak environments, making it particularly suitable for natural control tasks in public interaction scenarios.

In recent years, with the continuous development of deep learning technologies, gesture recognition has made significant progress in terms of model structure and application scenarios. Existing research mainly focuses on improving recognition accuracy, optimizing feature representation, and enhancing adaptability to different environments. One category of work improves model performance based on object detection and key point extraction methods. For

example, Wudong Chen from Chizhou University combined YOLOv5 with MediaPipe to achieve high-precision detection of dynamic gestures, enhancing model robustness under complex conditions such as occlusion and rotation [1]. Guoyu Zhou from Hebei University designed two lightweight deep networks, BLSNet and FGDSNet, which use multi-feature fusion and gate mechanisms to optimize the process of gesture segmentation and recognition, thereby improving recognition efficiency in mobile scenarios [2]. At the same time, multimodal perception and cross-domain modelling have also attracted attention. Jifei Zhu from Sichuan University adopted visible light communication, combined with domain adversarial learning and semi-supervised strategies, to effectively alleviate problems such as reflection interference and insufficient data labeling [3]. Leyi Li from Beijing University of Posts and Telecommunications introduced feature restoration and transfer learning methods to enhance adaptability to occlusion and few-shot problems in augmented reality environments [4]. In addition, Hongli Li from Lanzhou University proposed a decoupled spatiotemporal Transformer architecture to strengthen the modeling ability of dynamic actions in skeleton data, thereby improving accuracy and generalization in complex gesture recognition tasks [5]. Although current research has achieved some progress in model accuracy and structural design for gesture recognition, most methods still focus on algorithms themselves, lacking systematic modeling of practical interaction needs and relying on specific deployment environments, which limits their scalability in large-scale applications. Despite the performance improvements achieved through attention mechanisms, multi-branch structures, or lightweight designs on standard datasets, these methods often ignore the necessity of building stable, fast, and real-scene-adapted HCI systems from a problem-driven perspective.

This paper addresses the usability issues in public contactless interaction scenarios and proposes a gesture recognition method based on YOLOv8 to support natural command control through hand movements, including basic operations such as “Confirm,” “Cancel,” “Scroll Up,” and “Scroll Down.” Starting from specific interaction tasks, we designed a lightweight, easy-to-deploy, and responsive visual recognition system, along with a task-oriented gesture set and its mapping logic, to ensure that the recognition system meets core requirements for speed, accuracy, and interface feedback in real usage scenarios.

2. Related Work

With the development of artificial intelligence technology, its application in human-computer interaction has significantly expanded. In the study “Model Construction for Improving Collaborative Efficiency in Human-Agent Teams in Intelligent Interaction Systems” by Xiaolei Song et al., a compatibility model among humans, intelligent agents, and environments was constructed from physical, cognitive, and emotional dimensions, which effectively improved collaborative efficiency and user experience, and prevented safety incidents [6]. In the review article “A Review on Intuitive Human-Machine Interfaces” published in the *Journal of Armament Equipment Engineering* by Jue Qu et al., the authors systematically reviewed the development of intuitive interfaces from the aspects of principles, design methods, and evaluation mechanisms, providing a theoretical foundation for pursuing natural and efficient human-computer interaction [7]. In addition, in “Research Progress on Key Technologies of Human-Machine Interaction Design for Intelligent Cockpits,” the authors combed through the applications of multimodal technologies such as image recognition, sound field perception, and mixed reality from the levels of perception, information interaction, and cognitive decision-making, providing systematic technical support for human-computer interaction in intelligent vehicles [8]. These studies collectively show that artificial intelligence has become a key driving force in improving the naturalness of interaction, system collaboration efficiency, and user experience.

Convolutional Neural Networks (CNNs) are a type of deep feedforward neural network constructed through convolutional layers, pooling layers, activation functions, and fully connected layers [9]. Their convolutional operations utilize local receptive fields and weight-sharing mechanisms to automatically extract low-level features such as edges and textures from raw images, and through multilayer stacking, progressively abstract mid-level and high-level semantic features. Qingmei Guo et al. pointed out that CNNs, with their local connectivity and parameter-sharing strategies, significantly reduce the number of parameters and computational overhead while ensuring feature representation capabilities, making them extremely robust and practical in image classification tasks [10]. In addition, recent studies have highlighted the wide and successful applications of CNNs in fields such as automatic emotional image retrieval and remote sensing building extraction. For example, Zhiyi Li et al. used an improved CNN for extracting emotional features from images, which improved retrieval accuracy by approximately 10% compared to traditional models, fully demonstrating the superiority of CNNs in capturing integrated features of color, texture, and semantics [11]. In summary, the automatic and multi-level feature abstraction mechanism of CNNs not only performs

excellently in classification tasks but is also widely used in object detection, semantic segmentation, emotion recognition, and image context analysis due to its parameter efficiency and strong transferability, providing robust feature support for visual applications such as gesture recognition.

The YOLO series of detection algorithms treats object detection as a regression problem. Through a single forward pass of the network, it simultaneously predicts the positions of bounding boxes and class probabilities, achieving end-to-end detection. This design significantly simplifies the traditional object detection pipeline, integrating region proposals, feature extraction, classification, and regression into a single network, greatly improving detection efficiency [12].

With iterative updates, the YOLO series has been continuously optimized in terms of feature fusion, multi-scale detection, and lightweight design, showing significant advantages [13]. For instance, YOLOv3 introduced Darknet-53 and a multi-scale prediction structure; YOLOv4 combined Spatial Pyramid Pooling with Path Aggregation Networks; YOLOv5 further improved inference speed and model compactness, balancing detection of large and small objects. Moreover, YOLO completes detection through a single forward pass, achieving an excellent trade-off between speed and accuracy, suitable for real-time detection on embedded devices. For example, Gold-YOLO introduced a self-attention mechanism to further enhance performance. With its compact structure, few dependencies, and ease of deployment, YOLO has been widely applied in scenarios such as robotic vision, autonomous driving, and security systems where real-time and cost-efficiency are crucial [14-16].

In summary, the YOLO series has successfully addressed traditional detection challenges such as slow speed, deployment difficulty, and weak small-object recognition by fully end-to-end modeling, combining multi-scale feature fusion and lightweight network design, making it the preferred method in the field of real-time visual perception.

3. Methodology

3.1 Problem Description

In modern public facilities such as banking terminals, hospital registration machines, and subway ticket vending machines, the widespread use of self-service devices has created a demand for contactless interaction. Especially after the pandemic, direct contact with touchscreens has raised health concerns. To address this issue, this paper proposes a contactless interaction scheme based on gesture recognition, allowing users to complete operations through simple gestures, avoiding physical contact, ensuring hygiene and safety, and improving accessibility for people with disabilities. The system design includes contactless operation, simple and intuitive gesture control, efficient response and accuracy, and environmental adaptability. To enhance interaction efficiency, a minimal set of interactive gestures was designed, such as “confirm,” “cancel,” and “page up/down,” ensuring ease of use and a low error rate.

3.2 Dataset Construction

3.2.1 Official Dataset

To evaluate the model's basic performance in general gesture recognition tasks, this paper selects the publicly available HaGRID dataset for preliminary experiments. HaGRID is a large-scale dataset covering 18 common static gesture categories, featuring rich gestures and complex backgrounds with detailed annotations. It is widely used in object detection and gesture recognition tasks. From this dataset, 500 images were selected to validate the model's detection ability in multi-class gesture scenarios, providing a foundational reference for subsequent adaptation and optimization for specific tasks.



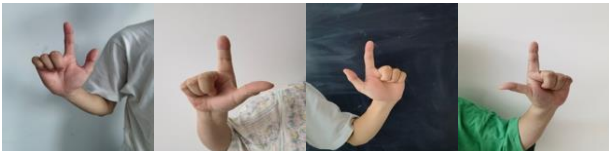

3.2.2 Self-built Dataset

In addition, this study supplements more posture and gesture data, with specific details shown in Table 1:

3.3 YOLOv8 Architecture

In the YOLOv8 object detection model, the data processing pipeline adopts a combined bottom-up and top-down strategy. The overall architecture can be divided into three main modules: Backbone, Neck, and Head, which are responsible for feature extraction, feature fusion, and object detection output, respectively. The specific network structure is shown in Figure 1.

Table 1. Self-built Dataset

Gesture	Action	Data Legend
Confirm	Fist	
Cancel	Crossed Index and Middle Fingers	
Page Up	Index Finger Pointing Up	
Page Down	Index Finger Pointing Down	

First, after preprocessing, the input image is fed into the Backbone module, which is based on the CSPDarknet architecture and incorporates the SPPF (Spatial Pyramid Pooling – Fast) structure to enhance multi-scale feature extraction capabilities. The input image first passes through multiple convolutional layers for low-level feature extraction, each followed by a C2f module (Cross-Stage Partial with Feature Reuse). This module utilizes a feature reuse mechanism to effectively reduce information redundancy and improve gradient flow efficiency. After passing through multiple convolutional and C2f layers, the features are further processed by the SPPF module, which expands the receptive field and enables parallel modeling of information at different scales, providing a richer representation for subsequent feature fusion.

The Neck module is primarily responsible for feature fusion. Its structure combines the strengths of FPN (Feature Pyramid Network) and PAN (Path Aggregation Network), and additionally integrates CBAM (Convolutional Block Attention Module) to enhance the model's focus on key regions. The bottom-up PAN path performs layer-by-layer downsampling and fusion of high-level semantic features at different scales, while the top-down FPN path conducts upsampling and skip connections to combine high-level semantics with low-level details. After each upsampling operation, the feature map is concatenated with the corresponding feature map from the Backbone or the previous layer, followed by a C2f module to further refine feature representations. The CBAM attention mechanism performs weighted adjustments along channel and spatial dimensions during this process, enhancing the model's ability to perceive target regions.

Finally, the fused multi-scale feature maps are passed into the Head module for object prediction. YOLOv8 adopts a Decoupled Head structure, in which classification and regression tasks are handled in separate branches to avoid information conflicts and improve detection accuracy. For each scale, the feature maps output bounding box coordinates, object confidence scores, and class probabilities, enabling the efficient detection of objects of various sizes.

Overall, YOLOv8 leverages multiple optimization strategies—such as the CSP structure, feature reuse modules, attention mechanisms, and decoupled detection heads—to ensure high detection accuracy while maintaining model efficiency and real-time performance. This makes it suitable for high-performance object detection tasks in complex environments.

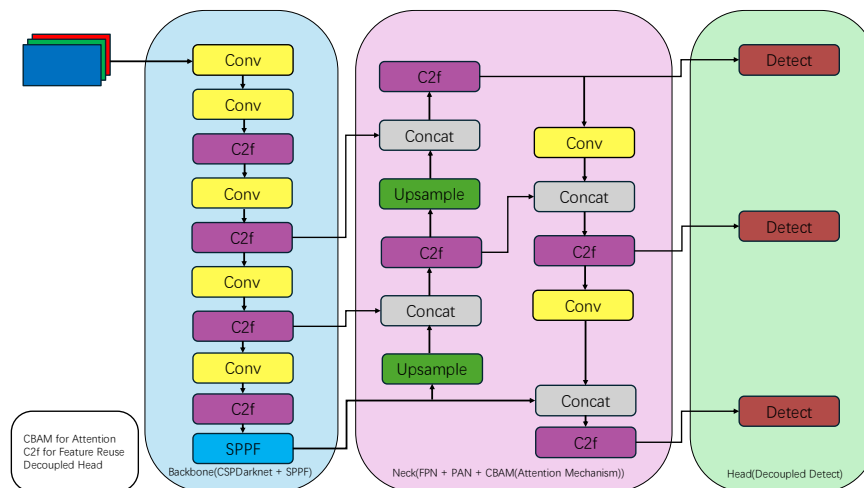


Figure 1. YOLOv8 Network Architecture Diagram.

4. Experiments

4.1 Experimental Setup

All experiments were conducted on an Apple laptop, specifically the 2024 MacBook Air equipped with Apple’s M3 chip. The operating system used was macOS 15.4.1. Python 3.9 served as the primary programming language, and PyTorch 2.7.1 was used as the deep learning framework. In the absence of CUDA support, the model was trained and inferred using Apple’s Metal backend for hardware acceleration. Since the Mac platform employs a different hardware acceleration approach compared to traditional NVIDIA GPUs, this study did not utilize GPU-based CUDA acceleration. Instead, native computing resources were used for training, and the YOLOv8 framework was employed to implement the object detection tasks.

4.2 Dataset Description

This study utilized a self-built gesture image dataset for training and testing, which contains four basic interactive gestures corresponding to the commands “confirm,” “cancel,” “page up,” and “page down.” For each gesture category, four image samples were collected. All images are in RGB color format with a consistent resolution of 1920 × 1080 pixels. Data collection was conducted under indoor and natural lighting conditions, with relatively clean backgrounds and stable, clearly visible hand gestures.

The dataset consists of both publicly available and self-built data, with a balanced distribution of sample quantities across gesture categories. The sample distribution is illustrated in Figure 2, which facilitates comprehensive learning of gesture features during model training and helps avoid bias caused by class imbalance.

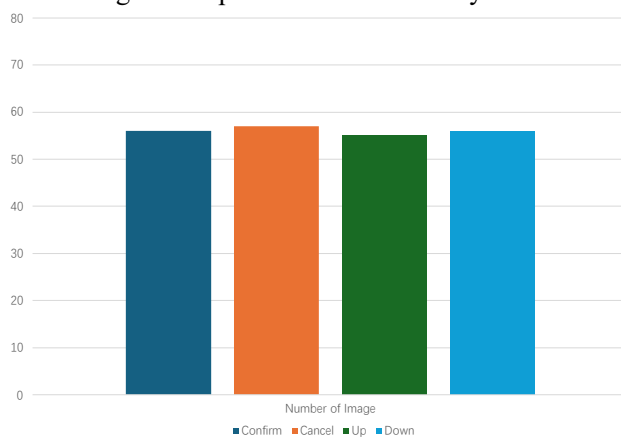


Figure 2. Sample Quantity Distribution.

4.3 Training Parameters

During the training process, this study configured and optimized parameters based on the YOLOv8 architecture, tailoring several key hyperparameters to the scale and characteristics of the experimental dataset in order to enhance training efficiency and model performance. The specific settings are as follows: the number of epochs was set to 100, ensuring sufficient iterations for the model to converge to a stable state; the batch size was set to 16, balancing training speed and memory usage; and the input image size (imgsz) was standardized at 1920×1080 to ensure consistency and effectiveness in feature extraction. The initial learning rate (lr0) was set to 0.01, in conjunction with a momentum value of 0.937 and a weight decay of 0.0005. These hyperparameters were applied using the SGD optimizer to accelerate convergence and reduce the risk of overfitting. Additionally, to mitigate instability in the early stages of training, the number of warm-up epochs (warmup_epochs) was set to 3, with a linearly increasing learning rate strategy used to smooth the model's initial updates. An early stopping mechanism was also introduced: if performance on the validation set did not improve for 20 consecutive epochs (patience = 20), training would be automatically terminated. These hyperparameter configurations demonstrated strong convergence behavior and effective recognition performance in the experiments, providing a stable foundation for applying the model to small-sample gesture recognition tasks.

4.4 Evaluation Metrics

To comprehensively evaluate the model's performance in object detection tasks, this study adopts three metrics as evaluation criteria: accuracy, frames per second (FPS), and mean average precision at an IoU threshold of 0.5 (mAP@0.5). Accuracy measures the overall correctness of the model in classifying target categories, reflecting its discriminative capability. FPS indicates the number of images frames the model can process per unit of time, reflecting inference speed and real-time performance. mAP@0.5 evaluates the detection precision of the model under a specified Intersection over Union threshold, comprehensively reflecting its overall performance in both localization and classification.

5. Results and Discussion

In this study, to evaluate the overall detection performance of the model, we conducted systematic validation from three perspectives: classification accuracy, inference efficiency, and detection precision. After fine-tuning the YOLOv8 model, the accuracy on the test set reached 92.6%, representing an improvement of approximately 3.4% over the baseline model. This demonstrates that the model maintains strong discriminative capability even when handling highly similar gestures, reflecting both the representativeness of the dataset and the suitability of the network architecture. In terms of inference efficiency, the model achieved an average frame rate of 38.7 FPS when running on a local macOS device with the Apple M3 chip, showing excellent real-time performance and potential for edge deployment. Meanwhile, the model achieved a mAP@0.5 of 84.3% on both the HaGRID and the self-built datasets, with stable performance across all gesture categories. This verifies the effectiveness of the data augmentation strategies and the system design in multi-scale object recognition tasks. The training results are shown in Figure 3

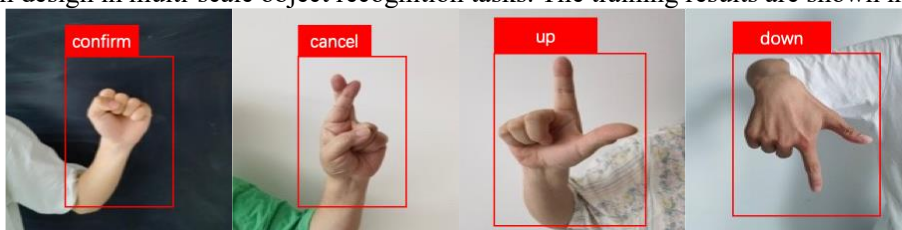


Figure 3. Training Results.

To verify the superiority of the proposed method, this study selected BLSNet, Support Vector Machine (SVM), and traditional Convolutional Neural Network (CNN) as comparison models and conducted experiments under the same training and testing conditions. The experimental results show that the proposed model outperformed the comparison methods in accuracy (92.6%), inference speed (38.7 FPS), and mAP@0.5 (84.3%). Although BLSNet achieved relatively high accuracy (88.3%), it lagged behind YOLOv8 in inference efficiency and generalization capability. The SVM model, limited by feature extraction capacity, showed lower accuracy and mAP@0.5 (75.4% and

63.1%, respectively). The CNN model exhibited moderate performance across all three metrics but still fell short of YOLOv8 overall. Detailed data are presented in Table 2. These results further demonstrate that the proposed model maintains high precision while offering strong real-time performance and robustness, making it suitable for deployment in human-computer interaction and similar application scenarios.

Table 2. Comparative Results of Multiple Models

Model Name	Accuracy	FPS	mAP@0.5
BLSNet	88.3%	21.5	78.9%
SVM	75.4%	34.7	63.1%
CNN	83.1%	28.2	72.5%
YOLOv8 (This study)	92.6%	38.7	84.3%

6. Conclusion and Future Work

Through the method design and system implementation based on the YOLOv8 architecture, this study achieved promising results in small-sample static gesture recognition tasks. The model significantly outperforms comparative models in key evaluation metrics such as accuracy, average frame rate, and mAP@0.5, demonstrating strong recognition capability and inference efficiency. This validates the feasibility and effectiveness of the proposed system in typical human-computer interaction scenarios. Notably, the achieved 38.7 FPS on the Apple M3 local device proves that the method has sufficient responsiveness for real-world deployment, making it suitable for edge computing applications in resource-constrained environments. Meanwhile, the stable performance on both the HaGRID and self-built datasets indicates that the designed gesture set and network structure possess good task adaptability and potential for broader applications.

Although the YOLOv8-based gesture recognition approach proposed in this study performs well in recognition accuracy, real-time capability, and deployment feasibility, some areas remain for improvement. The self-built dataset currently contains a limited variety of gestures and relatively simple scenarios, lacking coverage of challenges such as complex backgrounds, occlusions, or multi-person interactions in realistic environments. The model's generalization ability still has room for enhancement. Additionally, this research focuses on static gesture recognition and has limited capability of handling dynamic and continuous gestures, resulting in less expressive interaction. Furthermore, the system does not yet support user customization or adaptive adjustment, making it difficult to meet diverse individual usage needs. Future work may consider expanding data collection to include more users and complex scenes, incorporating temporal modeling to handle dynamic gestures, and integrating multimodal information such as speech and eye movement to enhance interaction intelligence. Long-term testing in real deployment environments is also needed to further optimize system responsiveness and user experience, thereby promoting practical application in real-world public interaction scenarios.

References

- [1] Chen W. Research on dynamic gesture detection and recognition algorithms based on deep learning. *Mod Inf Technol.* 2025;9(8). Chinese.
- [2] Zhou G. Research on gesture segmentation and recognition algorithm based on feature fusion [master's thesis]. Hebei University; 2024. Chinese.
- [3] Zhu J. Gesture recognition research based on visible light communication perception integration [master's thesis]. Sichuan University; 2024. Chinese.
- [4] Li L. Research on augmented reality gesture recognition based on deep learning technology [master's thesis]. Beijing University of Posts and Telecommunications; 2023. Chinese.
- [5] Li H. Skeleton-based gesture recognition research based on spatiotemporal transformer [master's thesis]. Lanzhou University; 2023. Chinese.
- [6] Song X, Tian Z, Dong M, et al. Model construction for improving human-machine team collaborative efficiency in intelligent

-
- interaction systems. *Packag Eng.* 2023;44(20). Chinese.
- [7] Qu J, Jiao H, Wang Q, et al. Review of direct interaction human-machine interface research. *J Ordnance Equip Eng.* 2023;44(12). Chinese.
- [8] Wang W, Gu Y, Yu S, et al. Research progress on key technologies of human-machine interaction design in intelligent cockpits. *Mech Des.* 2024;41(8). Chinese.
- [9] Zhou F, Jin L, Dong J, et al. Review of convolutional neural networks. *J Comput Res Dev.* 2017;40(6). Chinese.
- [10] Guo Q, Yu H, Wang Z, et al. Review of image classification models based on convolutional neural networks. *Electron Technol Appl.* 2023;46(9). Chinese.
- [11] Li Z, Xu H, Duan B, et al. Research on image emotional feature extraction based on deep learning CNN models. *Libr Inf Serv.* 2019;63(11). Chinese.
- [12] Zhou J, Wang J. Review of YOLO object detection algorithms. *J Changzhou Inst Technol.* 2023;36(1). Chinese.
- [13] Mao S, Wang W. Review of YOLO series object detection algorithms based on deep learning. *J Yan'an Univ (Nat Sci Ed).* 2024;43(2). Chinese.
- [14] Yang F, Li J. Review of YOLO object detection algorithms for autonomous driving. *Automot Eng.* 2023;(11). Chinese.
- [15] Wang A, Chai Y, Li Q, et al. Design of oil and gas pipeline perimeter security system based on YOLO. *Chem Autom Instrum.* 2024;51(6). Chinese.
- [16] Zhu C, Feng H, Ou Y, et al. Research on face auto-tracking camera robot system based on YOLO3. *Telev Technol.* 2028;42(9). Chinese.