Traitement du Signal
Vol. 41, No. 6, December, 2024, pp. 3335-3341
Journal homepage: http://iieta.org/journals/ts

Improved Campus Vehicle Detection Method Based on YOLOv11 and Grayscale Projection-Based Electronic Image Stabilization Algorithm

Shaoqing Tian[1], Yuxuan Lu[1], Fan Jiang[1], Chunrong Zhan[2], Chongli Huang[3*]

[1] Tropical Agriculture and Forestry College, Hainan University, Haikou 570228, China
[2] International Business School, Hainan University, Haikou 570228, China
[3] Library, Hainan University, Haikou 570228, China

Corresponding Author Email: 991242@hainanu.edu.cn

https://doi.org/10.18280/ts.410649

Received: 8 June 2024
Revised: 13 November 2024
Accepted: 27 November 2024
Available online: 31 December 2024

Keywords:
object detection, Receptive Field Attention Convolution (RFAConv), YOLOv11, Electronic Image Stabilization (EIS)

ABSTRACT

With the expansion of university campuses and the increase in internal traffic flow, campus traffic safety issues have become increasingly prominent. To effectively monitor and manage the vehicles within the campus, improve traffic efficiency, and ensure pedestrian safety, an improved vehicle recognition algorithm combining grayscale projection and the YOLOv11 model is proposed. The method first utilizes grayscale projection technology to preprocess the captured images, enhancing the contrast and reducing background interference to improve the clarity of target features. The Receptive Field Attention Convolution (RFAConv) optimization technique is applied to the C3K2 module of the YOLOv11 model, combined with electronic image stabilization (EIS) algorithms to form an optimized strategy. Experimental results show that the improved YOLOv11 model, integrated with the optimization strategy, maintains high recognition accuracy and real-time performance even in complex environments, providing reliable technical support for the construction of intelligent campus transportation systems.

1. INTRODUCTION

As campus safety issues have been increasingly emphasized, the application of vehicle detection by monitoring personnel has become an important tool to improve campus security management. The background of this application is to address the increasingly complex campus traffic environment and safety risks posed by external vehicles, ensuring the safety of students and staff in terms of personal property. By utilizing advanced image recognition technology and artificial intelligence algorithms, real-time monitoring and analysis of vehicles entering and exiting the campus can be achieved, providing important support for the development of intelligent campuses. In recent years, object detection has become one of the most critical tasks in the field of artificial intelligence and computer vision. Whether it is enabling autonomous vehicles to identify obstacles or helping surveillance systems track moving objects in real-time, object detection models have been applied across many industries, and a number of scholars have started researching the improvement of YOLO algorithms to meet the vehicle detection needs in various scenarios.

In terms of algorithm optimization, Wang et al. [1] first proposed the concept of using YOLO algorithm for real-time vehicle detection, laying the foundation. Subsequent research has continuously promoted the iteration and upgrade of the YOLO algorithm. Li and Huang [2] introduced the YOLOv2 model, enhancing detection accuracy and reducing false positive rates; Zhang and Liu [3] proposed an improved Tiny YOLO algorithm based on small Zynq SoC hardware acceleration for embedded systems, achieving efficient, low-power real-time vehicle detection. Li et al. [4] optimized the YOLOv2 algorithm under the Darknet framework, improving multi-object detection capabilities, and it was successfully applied in the transportation field. To further improve detection performance, Liu et al. [5] proposed an improved YOLO vehicle detection algorithm, enhancing detection speed and accuracy by adjusting the network structure and training strategy. Zhang et al. [6] focused on illegal vehicle detection and developed an efficient illegal vehicle recognition system using YOLOv2. Zhu et al. [7] combined TridentNet with YOLO and proposed a new method, particularly suited for vehicle detection tasks in complex backgrounds. Ma et al. [8] introduced the Deep-SORT algorithm in conjunction with an improved YOLOv3 model to achieve continuous tracking of road vehicles, paving the way for intelligent transportation system implementation. Zhang et al. [9] designed a new model called YOLO-DNF for vehicle and pedestrian detection in assisted driving environments, emphasizing safety improvements. Guo et al. [10] addressed the problem of vehicle detection in dim environments by proposing the Dim_env-YOLO algorithm, expanding the application range of this technology. The latest research shows that Sun et al. [11] applied the YOLO-MMCE algorithm in visual detection of high-risk area engineering vehicles, proving its effectiveness and reliability in specific fields.

In unmanned aerial vehicle (UAV) scenarios, the application of the YOLO model for vehicle recognition has become a research hotspot. Asifa et al. [12] focused on using semantic segmentation technology and YOLO detectors to analyze vehicle images captured by drones, with research having significant value for traffic flow monitoring, accident response, and other areas. Li and Abdullah [13] conducted a comprehensive review of the application of YOLO series algorithms in UAV images, pointing out their potential in military reconnaissance, agricultural monitoring, and other fields. Pargieła [14] used the YOLO algorithm to study vehicle detection and occlusion problems in drone images, improving photogrammetry products and providing new tools for geospatial data analysis. Rani et al. [15] introduced the use of YOLOv4 convolutional neural networks for multi-band optical detection with drones, demonstrating its flexibility and high performance across diverse tasks.

YOLO algorithms also play an important role in license plate recognition. Moussaoui et al. [16] enhanced the performance of automatic license plate recognition systems by integrating YOLOv8 and OCR technology, solving the problem of license plate detection in complex environments and greatly improving traffic management efficiency. Al-Batat et al. [17] combined YOLO algorithms with OCR technology to build a complete automatic license plate recognition system, with a high level of automation and applicable to various traffic law enforcement scenarios.

Dewantoro et al. [18] analyzed the accuracy of YOLO algorithms in vehicle count detection at intersections, validating their effectiveness in real-world traffic management and providing empirical support for intelligent transportation system construction. Rodríguez-Rangel et al. [19] analyzed the statistical and AI algorithms for real-time vehicle speed estimation based on YOLO, providing a theoretical foundation and technical implementation path. Nasehi et al. [20] explored the feasibility of using YOLOv5 and MobileNet for vehicle type and speed detection on Android devices. In recognition, Prethi et al. [21] proposed an intelligent safety vehicle filtering and tracking system based on edge computing, which combined YOLO object detection and EasyOCR text recognition technology to improve system response speed and security. Golyak et al. [22] optimized the YOLOv4 model in convolutional neural networks and applied it to UAV multi-band optical detection.

In urban management, the application of the YOLO model helps in planning and design. Mittal et al. [23] proposed a hybrid scheme combining the advantages of Faster R-CNN and YOLO models to more accurately estimate traffic density, assisting in traffic flow control and urban planning. Lechgar et al. [24] studied the use of YOLOv2 and aerial imagery to detect urban fleets, providing efficient monitoring tools for urban management and helping optimize urban traffic resource allocation.

Different scenarios affect the performance of YOLO models, and corresponding studies have been conducted. Zaghari et al. [25] researched the application of YOLO non-maximum suppression fuzzy algorithms in obstacle detection for autonomous vehicles, reducing false positive rates and enhancing the safety of autonomous driving systems. Kiran et al. [26] proposed an enhanced deep YOLO algorithm combined with discrete wavelet transform to address vehicle detection challenges in changing weather conditions (such as rain or fog). Mishra and Yadav [27] studied the application of YOLOv5 in vehicle detection in high-density traffic scenarios and proposed effective solutions to occlusion problems in crowded road environments. Rafi et al. [28] performed a performance analysis of deep learning YOLO models used in South Asia, revealing the applicability of different model versions in specific geographic regions. Mahmood et al. [29] used YOLO computational mechanisms to detect vehicles in infrared images, solving problems that could not be addressed under visible light conditions and expanding the application scope of vehicle detection technology.

In summary, these studies not only demonstrate the potential of the YOLO algorithm and its variants in vehicle detection but also reflect the deepening of research and the expansion of application scenarios. However, the aforementioned studies rarely address accuracy issues in unstable image scenarios. Therefore, this study proposes a processing strategy combining grayscale projection stabilization algorithms with YOLOv11, aiming to improve vehicle detection accuracy in unstable image conditions.

2. GRAYSCALE PROJECTION AND YOLOv11

2.1 Grayscale projection algorithm

The Grayscale Projection-based Electronic Image Stabilization Algorithm (GPEIS) [30] is a motion estimation algorithm that utilizes the grayscale projection curve of an image to estimate image motion. It fully leverages the variation pattern of the overall grayscale distribution of the image to determine the motion vector. Only a single correlation operation is needed on the projection curves of the image's rows and columns to calculate the motion vector, which greatly reduces the computational load while ensuring high computational accuracy.

The application of the grayscale projection algorithm mainly includes four steps [31]:

(1) First, the collected image is processed and converted into a grayscale image. Since EIS systems are often used in scenes with long focal lengths, large fields of view, and low resolutions, especially when capturing distant targets, the signal-to-noise ratio of the captured images is usually very low. To better detect the motion vectors of the image, preprocessing is necessary. Histogram equalization is commonly used for image enhancement preprocessing. Histogram equalization can distribute the grayscale values evenly across each pixel in the image and make the image histogram as uniformly distributed as possible. This improves the contrast of the image, which helps ensure the accuracy of motion vector extraction by the algorithm.

(2) Perform grayscale projection calculations, i.e., add the grayscale values of each row or column of the image together. The projection formula is as follows:

$$\begin{aligned} row(i) &= \sum_{j=1}^{n} pic(i,j) \;; \\ column(j) &= \sum_{i=1}^{m} pic(i,j) \end{aligned} \tag{1}$$

where, $row(i)$ is the projection value of the i-th row; $column(j)$ is the projection value of the j-th column; $pic(i,j)$ is the grayscale value at the point (i,j) in the image.

(3) Perform correlation operations to obtain the jitter offset. The column correlation operation formula is as follows:

$$C(w)=\sum_{j=1}^{n}\begin{bmatrix} column_x\left(j+w-1\right) \\ -column_r\left(M+j\right) \end{bmatrix}^2; \quad 1\le w\le 2M+1 \tag{2}$$

where, w is the search range; $C(w)$ is the correlation value; M is the detection range of image jitter; w_{min} is the value of w when $C(w)$ is the minimum.

According tow_{min}, the vertical displacement $motion_c$ of the image in the vertical direction and be obtained, and the row correlation operations can be done similarly.

$$motion_c = M+1-W_{\min} \tag{3}$$

(4) Reverse shift the jittered image by $motion_r$ and $motion_c$, then the resulting image after shifting is the stabilized image.

Although the GPEIS algorithm has the advantages of fast computation and stable performance, it has significant limitations in estimating rotation and scaling motion. Many studies have addressed these limitations by researching various scenarios such as unmanned boats, drones, and satellite assembly [32-34], compensating for the algorithm's shortcomings and enabling its wide application in fields like EIS.

2.2 YOLOv11 model architecture

YOLOv11 was released on September 30, 2024. Compared to previous versions, this version is faster, more accurate, and more efficient in detecting objects. It not only achieves significant improvements in accuracy but also maintains excellent real-time performance, making it particularly suitable for applications that require fast responses [35]. Its main features include: optimized speed and accuracy, the use of innovative designs such as the C3K2 block, SPFF module, and C2PSA block, which enhance the understanding and processing efficiency of spatial information. Through improved feature fusion mechanisms, it better captures and analyzes small-sized targets. As shown in Figure 1, the model structure of YOLOv11 is illustrated.

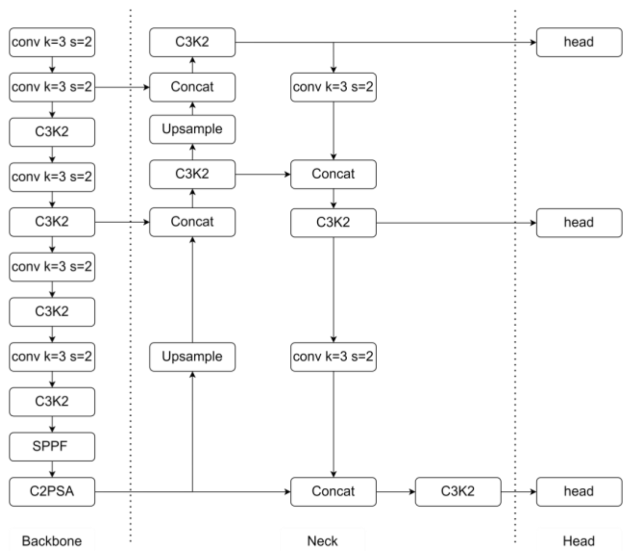

Figure 1. YOLOv11 model architecture diagram

YOLOv11 adopts an improved version of CSPDarknet53 as the backbone network, which generates feature maps at different scales (labeled P1 to P5) through five down-sampling operations. This multi-scale feature extraction helps capture targets of different sizes. In addition, the backbone network introduces the CBS (Convolution, Batch Normalization, and SiLU activation function) module for initial feature extraction, and the C3K2 module replaces the original C2f module, enhancing the network's expressive ability and flexibility. The Spatial Pyramid Pooling Fast (SPFF) module is used to increase the diversity of feature expressions, while the C2PSA module improves the SE attention mechanism by incorporating a multi-level pyramid slice attention (PSA) mechanism, making it more suitable for processing multi-level features.

The neck network of YOLOv11 adopts the Path Aggregation Network - Feature Pyramid Network (PAN-FPN) structure, which enhances the fusion of shallow position information and deep semantic information through a bottom-up path, compensating for the lack of object localization information in the FPN structure. This bi-directional information flow ensures that the feature maps contain both rich semantic information and precise position details, greatly improving the model's robustness and adaptability.

The head network of YOLOv11 adopts a decoupled structure, where independent branches are responsible for predicting class and location information, respectively. For the classification task, binary cross-entropy loss (BCELoss) is used; for the bounding box regression task, the Distributive Focal Loss (DFL) and Complete Intersection over Union (CIoU) loss functions are used to improve the accuracy of bounding box predictions. Additionally, two depthwise separable convolutions (DWConv) are added to the classification detection head, which not only reduces the number of parameters and computational complexity but also enhances the model's lightweight nature.

Overall, YOLOv11, through a series of innovative designs and optimization techniques, further improves the performance of real-time object detection, demonstrating its strong potential in various computer vision tasks. This study primarily focuses on improving YOLOv11 and developing optimization strategies for specific scenarios. In the following, YOLO will be used as the abbreviation for YOLOv11.

3. IMPROVED YOLO IMAGE STABILIZATION OPTIMIZATION DETECTION STRATEGY

3.1 Improved YOLO model

In campus monitoring images, the concentration of targets at certain time periods and the lack of distinct features often lead to missed and false detections. The performance of YOLO in these cases is not ideal. However, using a larger model would significantly increase computational load and parameter costs, especially in terms of feature extraction capabilities of the C3k2 module when processing campus vehicle detection.

Therefore, we consider optimizing the C3k2 module to improve the YOLO model. A RFAConv [36] is introduced, which proposes an RFA mechanism that assigns specific weights to each receptive field, solving the problem of parameter sharing. This can be expressed by the following formula:

$$
\begin{aligned}
\mathrm{F} &= \mathrm{Softmax}\left(\mathrm{g}^{1\times1}\left(\mathrm{AvgPool}(\mathrm{X})\right)\right) \\
&\quad \times \mathrm{ReLU}\left(\mathrm{Norm}\left(\mathrm{g}^{k\times k}(\mathrm{X})\right)\right) \\
&= \mathrm{A}_{rf} \times \mathrm{F}_{rf}
\end{aligned} \tag{4}
$$

Through the above computation process of RFA, a new RFAConv module is formed, which enhances its ability to process complex images, thereby resulting in an improved C3k2 module, referred to as C3k2-I, as shown in Figure 2.

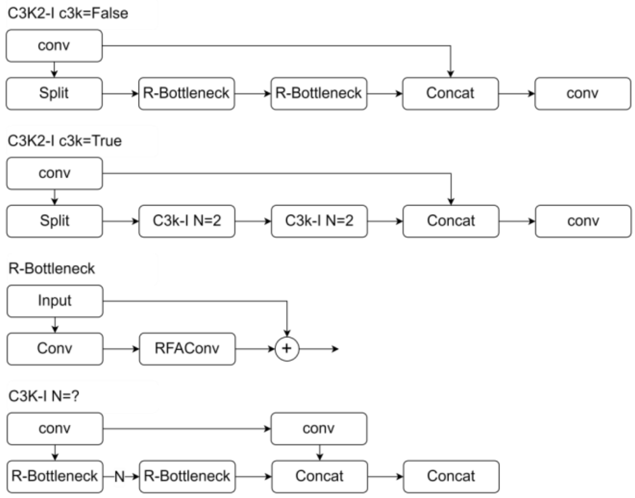

Figure 2. Improved C3k2 module

3.2 Image stabilization optimization detection strategy

In campus security management and intelligent monitoring systems, video surveillance equipment is an important tool for ensuring the safety of students and teachers, as well as maintaining order. However, campus video surveillance equipment often faces external environmental factors such as weather changes (rain, snow, fog), drastic fluctuations in lighting conditions (day-night transitions, shadow movement), and physical interference (camera shake, leaf obstruction, etc.). These factors can lead to a decrease in image quality, which in turn affects the accuracy of the YOLO model's recognition. Unstable image input may cause false positives or missed detections in the YOLO algorithm, weakening the system's reliability and response speed. Therefore, the GPEIS algorithm is introduced to preprocess the image, obtaining high-quality image data. The image stabilization process is shown in Figure 3.

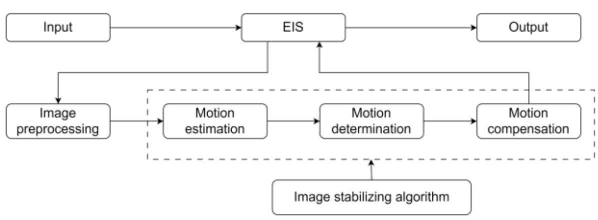

Figure 3. EIS algorithm process

By applying the EIS algorithm to preprocess the input video stream, the image blur and displacement caused by camera shake, wind, and other factors are effectively reduced. This significantly improves image stability without affecting real-time performance. Based on this, the improved YOLO algorithm further enhances the model's robustness in complex environments, forming a campus vehicle detection strategy that combines the improved YOLO algorithm with the EIS algorithm preprocessing, as shown in Figure 4.

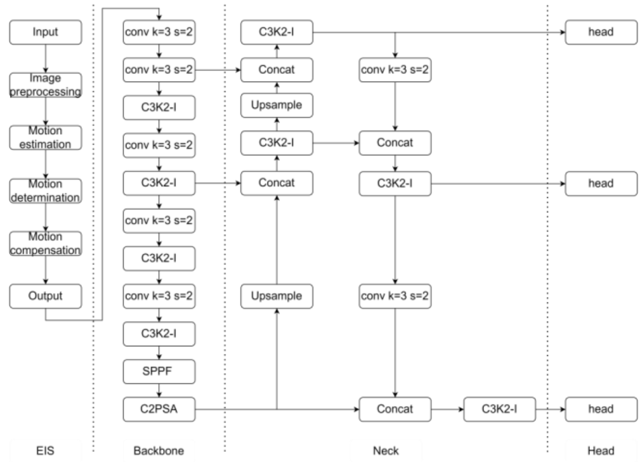

Figure 4. Improved YOLO algorithm and image stabilization optimization strategy model

This comprehensive strategy not only enhances the recognition accuracy of the YOLO algorithm under unstable image data conditions but also ensures the system's real-time performance, providing more reliable technical support for campus security management and intelligent traffic monitoring.

4. EXPERIMENT VALIDATION AND RESULT ANALYSIS

4.1 Experimental data

In this study, the VisDrone2019 dataset, publicly available from the Machine Learning and Data Mining Laboratory of Tianjin University, and a self-collected dataset are used for experiment validation to assess the effectiveness and generalizability of the improved YOLO algorithm and EIS preprocessing strategy. The VisDrone2019 dataset contains 6,471 training images, 548 validation images, and 1,610 test images. Additionally, a segment of video footage was recorded in front of the school gate using a handheld device, capturing traffic conditions over time and collecting rich data on school traffic targets.

The test set was expanded by adding video captured at the school gate using handheld devices. Artificially induced shake was added to the complex scenes, and the recorded video was then converted frame-by-frame into a certain number of images. These images were labeled using the Vott tool, where "0" represents pedestrians, "1" represents non-motor vehicles, and "2" represents motor vehicles.

4.2 Experimental process

The original images were first optimized through a data preprocessing module, including stabilization and enhancement, to address the issue of unstable images. The processed images were then input into the YOLOv11 network for training. After multiple forward and backward propagations, a weight model that meets the accuracy requirements was obtained. In the actual detection process, newly collected images were inferred through this trained model to obtain accurate detection results, ensuring the algorithm's efficiency and reliability in complex environments. The experimental process is shown in Figure 5.

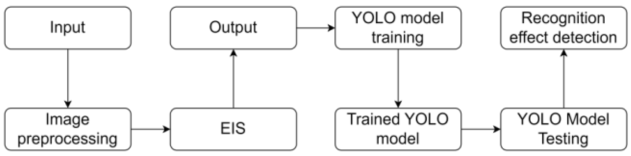

Figure 5. Experimental process

4.3 Experimental environment and parameter configuration

The experimental environment is a dedicated computer in the campus image processing laboratory, with the specific configuration shown in Table 1.

Table 1. Experimental environment and parameter configuration

Content	Configuration
Operating System	Windows11
CPU	Intel Core i9 13980HX 2.2GHz
Memory	16GB
GPU	NVIDIA GeForce RTX 4060 8GB
PyTorch Framework	2.5.1+CPU
Python Version	3.12.8
Batch Size	4
Image Size	640 × 640
Epochs	300

4.4 Experimental results and evaluation

First, the improved model was trained for 200 epochs, and the best weight parameter model obtained was used to perform inference on the target sample for detection. The results are shown in Figure 6.

Figure 6. Training results of the improved model

Figure 7. Recognition results with jitter

Next, images with jitter were input into the YOLO model for testing, and the test results are shown in Figure 7.

Then, the jittered image data was processed using the GPEIS algorithm, resulting in stabilized images. The EIS preprocessing process and stabilization output results are shown in Figure 8.

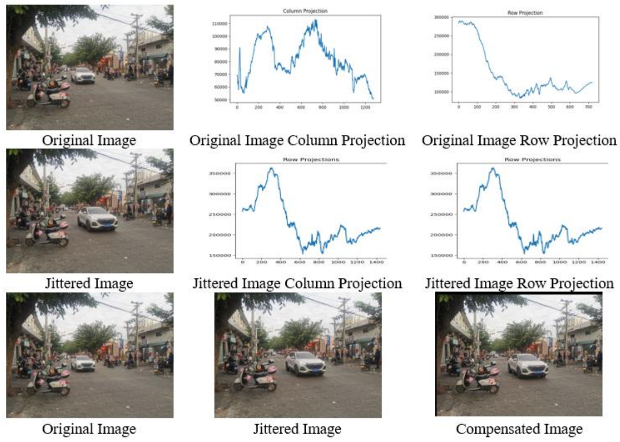

Figure 8. EIS grayscale projection preprocessing and stabilization output

Finally, the stabilized image output was fed into the improved YOLO model for target recognition to test the detection effect. The results are shown in Figure 9.

Figure 9. Recognition effect after optimization strategy

Evaluation metrics such as mAP, Precision, Recall, and training time were selected. By comparing the metrics, the model performance before and after optimization is summarized in Table 2.

Table 2. Comparison of improved YOLO algorithm

Model	mAP/%	Precision	Recall	Training time/h
YOLOv11	88.3	0.44	0.33	20h18m
YOLOv11 (RFAConv)	89.6	0.51	0.38	18h27m

The number of detections and average detection rate of the improved YOLO model and optimization strategy are compared in Tables 3 and 4.

From Tables 3 and 4, it can be seen that the improved YOLO model showed better relative detection improvements for pedestrians, non-motor vehicles, and motor vehicles. Since pedestrians are usually smaller in images compared to other targets, they are more prone to missed detections. The improved model significantly enhanced the detection ability for small targets, greatly increasing the pedestrian detection rate. Due to jitter in the data source leading to missed and false detections in the original model, preprocessing was applied to reduce instability caused by external factors, and the improved YOLO algorithm showed better detection rates for all three types of targets in specific scenarios. The model was able to more accurately distinguish between these three categories and reduce misclassification. Overall, the model with the optimization strategy demonstrated stronger detection performance, improving both detection rates and the accuracy of recognizing various traffic targets.

Table 3. Comparison of improved algorithm

Category	YOLOv11	Improved YOLOv11	Improvement Ratio/%
Pedestrians	106	128	20.7
Non-motor Vehicles	32	38	18.7
Motor Vehicles	15	18	20.0

Table 4. Comparison of optimization strategy

Category	Improved YOLOv11	Optimization Strategy	Improvement Ratio/%
Pedestrians	128	163	27.3
Non-motor Vehicles	38	44	15.7
Motor Vehicles	18	23	27.8

5. CONCLUSION

To enhance target detection performance in campus surveillance images under complex external environments, an improvement was proposed based on YOLO, addressing the shortcomings of the C3k2 module in small target feature extraction by introducing RFAConv. This module adaptively adjusts the receptive field size, improving the model's ability to perceive targets of different scales. A YOLO optimization strategy integrating the GPEIS algorithm was proposed to reduce the loss of detailed information caused by image jitter. With these optimizations, the improved model significantly enhanced detection performance in campus gate image target detection tasks. Experimental results show that the optimized model has improved detection ability in cases of image jitter, achieving higher accuracy for campus traffic intersection target recognition.

Looking ahead, this study only analyzed simple image jitter scenarios. Future work can focus on more complex jitter scenes and analyze the integration of YOLO with the EIS algorithm, modularizing the EIS algorithm into the YOLO model to further enhance the algorithm's robustness.

ACKNOWLEDGMENT

This work was also supported by Hainan Provincial Natural Science Foundation of China (Research on the Comprehensive Safety Risk Management Mechanism and Implementation Path of Schools in the Post Pandemic Era, 722RC632) and Hainan Provincial Philosophy and Social Sciences Planning Project (Research on the Construction Model and Implementation Path of Free Trade Port Game Industry System under the Background of Game Going Global, HNSK(YB)23-13). This work was supported by Humanities and Social Sciences Research Innovation Team of Hainan University, Hainan Free Trade Port Cross-border E-commerce Service Innovation Research Team (HDSKTD202025).

REFERENCES

[1] Wang, Y.N., Pang, Z.H., Yuan, D.M. (2016). Real-time vehicle detection based on YOLO algorithm. Journal of Wuhan University of Technology, 38(10): 41-46. https://doi.org/10.3963/j.issn.1671-4431.2016.10.008

[2] Li, Z., Huang, M.H. (2018). Vehicle detections based on YOLO_v2 in real-time. China Mechanical Engineering, 29(15): 1869-1874. https://doi.org/10.3969/j.issn.1004-132X.2018.15.015

[3] Zhang, Y., Liu, D. (2019). Real-time implementation of improved TINY YOLO vehicle detection algorithm based on Zynq SoC hardware acceleration. Journal of Computer Applications, 39(1): 192.

[4] Li, X., Liu, Y., Li, P., Zhang, L., Zhao, Z. (2018). Multi-object detection method for vehicles based on YOLO v2 algorithm in Darknet framework. Journal of Transportation Engineering, 18(6): 142-158. https://doi.org/10.19818/j.cnki.1671-1637.2018.06.015

[5] Liu, K., He, J.J., Zhang, Y.P. (2019). Improved YOLO vehicle detection algorithm. Modern Electronics Technique, 42(13): 47-50.

[6] Zhang, C., Tong, B., Cheng, J., Zhang, B., Zhang, R. (2020). Improved YOLO-v2 violation vehicle detection method. Computer Engineering and Applications, 56(20): 104-110. https://doi.org/10.3778/j.issn.1002-8331.1911-0382

[7] Zhu, M., Xing, H., Fang, R. (2020). Vehicle detection method based on YOLO TridentNet. Journal of Chongqing University of Technology (Natural Science), 34(11): 1-8.

[8] Ma, Y., Ma, Y., Cheng, S., Ma, Y. (2021). Road vehicle detection method based on improved YOLO v3 model and Deep SORT algorithm. Journal of Transportation Engineering, 21(2): 222-231. https://doi.org/10.19818/j.cnki.1671-1637.2021.02.019

[9] Zhang, X., Chen, C., Du, J., Li, J., Cheng, K. (2022). YOLO-DNF vehicle and pedestrian detection model for assisted driving. Computer Engineering and Applications, 58(24): 212-222. https://doi.org/10.3778/j.issn.1002-8331.2204-0505

[10] Guo, K., Wang, S., Li, X., Zhang, M. (2023). Multi-object detection of vehicles in dim scenes based on Dim env YOLO algorithm. Computer Engineering, 49(3): 312-320.

[11] Sun, H., Liu, L., Wang, D., Wang, T. (2024). YOLO-MMCE algorithm for visual inspection of engineering vehicles in high consequence areas. Oil and Gas Storage and Transportation, 43(9): 1031-1038.

[12] Qureshi, A.M., Butt, A.H., Alazeb, A., Al Mudawi, N., Alonazi, M., Almujally, N.A., Jalal, A., Liu, H. (2024). Semantic segmentation and YOLO detector over aerial vehicle images. Computers, Materials & Continua, 80(2): 3315-3332. https://doi.org/10.32604/cmc.2024.052582

[13] Li, J., Abdullah, M.I. (2024). YOLO series algorithms in object detection of unmanned aerial vehicles: A survey.

Service Oriented Computing and Applications, 18(3): 269-298. https://doi.org/10.1007/s11761-024-00388-w
[14] Pargieła, K. (2022). Vehicle detection and masking in UAV images using YOLO to improve photogrammetric products. Reports on Geodesy and Geoinformatics, 114(1): 15-23. https://doi.org/10.2478/rgg-2022-0006
[15] Rani, N.G., Priya, N.H., Ahilan, A., Muthukumaran, N. (2024). LV-YOLO: Logistic vehicle speed detection and counting using deep learning based YOLO network. Signal, Image and Video Processing, 18(10): 7419-7429. https://doi.org/10.1007/s11760-024-03404-w
[16] Moussaoui, H., Akkad, N.E., Benslimane, M., El-Shafai, W., Baihan, A., Hewage, C., Rathore, R.S. (2024). Enhancing automated vehicle identification by integrating YOLO v8 and OCR techniques for high-precision license plate detection and recognition. Scientific Reports, 14(1): 14389. https://doi.org/10.1038/s41598-024-65272-1
[17] Al-Batat, R., Angelopoulou, A., Premkumar, S., Hemanth, J., Kapetanios, E. (2022). An end-to-end automated license plate recognition system using YOLO based vehicle and license plate detection with vehicle classification. Sensors, 22(23): 9477. https://doi.org/10.3390/s22239477
[18] Dewantoro, N., Fernando, P.N., Tan, S. (2020). YOLO algorithm accuracy analysis in detecting amount of vehicles at the intersection. In IOP Conference Series: Earth and Environmental Science, 426(1): 012164. https://doi.org/10.1088/1755-1315/426/1/012164
[19] Rodríguez-Rangel, H., Morales-Rosales, L.A., Imperial-Rojo, R., Roman-Garay, M.A., Peralta-Peñuñuri, G.E., Lobato-Báez, M. (2022). Analysis of statistical and artificial intelligence algorithms for real-time speed estimation based on vehicle detection with YOLO. Applied Sciences, 12(6): 2907. https://doi.org/10.3390/app12062907
[20] Nasehi, M., Ashourian, M., Emami, H. (2024). Vehicle Type and Speed Detection on Android Devices Using YOLO V5 and MobileNet. Traitement du Signal, 41(3): 1377. https://doi.org/10.18280/ts.410326
[21] Prethi, K.N., Palanisamy, S., Nithya, S., Salau, A.O. (2024). Edge based intelligent secured vehicle filtering and tracking system using YOLO and EasyOCR. International Journal of Intelligent Transportation Systems Research, 1-24. https://doi.org/10.1007/s13177-024-00452-x
[22] Golyak, I.S., Anfimov, D.R., Fufurin, I.L., Nazolin, A. L., Bashkin, S.V., Glushkov, V.L., Morozov, A.N. (2020). Optical multi-band detection of unmanned aerial vehicles with YOLO v4 convolutional neural network. In SPIE Future Sensing Technologies, 11525: 195-200. https://doi.org/10.1117/12.2584591
[23] Mittal, U., Chawla, P., Tiwari, R. (2023). EnsembleNet: A hybrid approach for vehicle detection and estimation of traffic density based on faster R-CNN and YOLO models. Neural Computing and Applications, 35(6): 4755-4774. https://doi.org/10.1007/s00521-022-07940-9
[24] Lechgar, H., Bekkar, H., Rhinane, H. (2019). Detection of cities vehicle fleet using YOLO V2 and aerial images. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 42: 121-126. https://doi.org/10.5194/isprs-archives-XLII-4-W12-121-2019
[25] Zaghari, N., Fathy, M., Jameii, S.M., Shahverdy, M. (2021). The improvement in obstacle detection in autonomous vehicles using YOLO non-maximum suppression fuzzy algorithm. The Journal of Supercomputing, 77(11): 13421-13446. https://doi.org/10.1007/s11227-021-03813-5
[26] Kiran, V.K., Dash, S., Parida, P. (2024). Vehicle detection in varied weather conditions using enhanced deep YOLO with complex wavelet. Engineering Research Express, 6(2): 025224. https://doi.org/10.1088/2631-8695/ad507d
[27] Mishra, S., Yadav, D.K. (2024). Vehicle detection in high density traffic surveillance data using YOLO v5. Recent Advances in Electrical & Electronic Engineering (Formerly Recent Patents on Electrical & Electronic Engineering), 17(2): 216-227. https://doi.org/10.2174/2352096516666230428103829
[28] Rafi, M.M., Chakma, S., Mahmud, A., Rozario, R.X., Munna, R.U., Wohra, M.A.A., Paul, B. (2022). Performance analysis of deep learning YOLO models for South Asian regional vehicle recognition. International Journal of Advanced Computer Science and Applications, 13(9): 864-873.
[29] Mahmood, M.T., Ahmed, S.R.A., Ahmed, M.R.A. (2020). Detection of vehicle with infrared images in road traffic using YOLO computational mechanism. In IOP Conference Series: Materials Science and Engineering, 928(2): 022027. https://doi.org/10.1088/1757-899X/928/2/022027
[30] Huang, Y., Tian, S., Sun, X. (2014). Texture analysis method for musical instrument board based on grayscale projection. Computer Engineering and Applications, 50(6): 161-164. https://doi.org/10.3778/j.issn.1002-8331.1308-0273
[31] Tian, S. (2014). Research on forest fire stabilization and identification localization algorithm. Doctoral dissertation, Northeast Forestry University.
[32] Hu, J., Hou, Y. (2021). Electronic image stabilization method for unmanned boats based on multivariate grayscale projection. Journal of Missile and Guidance, 41(4): 65-68, 73.
[33] Yuan, W., Gao, Y., Wu, J. (2016). Unmanned aerial vehicle video stabilization method based on grayscale projection and block matching. Radio Engineering, 46(2): 19-22.
[34] Bai, F., Zhang, X., Zhang, M. (2016). Application of image stabilization algorithm based on partitioned grayscale projection in satellite assembly. China Mechanical Engineering, 27(2): 195-200.
[35] Li, B., Li, S. (2024). Improved YOLOv11n UAV small target detection algorithm. Computer Engineering and Applications.
[36] Zhang, X., Liu, C., Yang, D., Song, T., Ye, Y., Li, K., Song, Y. (2023). RFAConv: Innovating spatial attention and standard convolutional operation. arXiv preprint, arXiv:2304.03198.