World Journal of Information Technology
ISSN: 2959-9911
DOI: 10.61784/wjit3004

RESEARCH ON TRAFFIC OBJECT TRACKING AND TRAJECTORY PREDICTION TECHNOLOGY BASED ON DEEP LEARNING

An Chang[1], Dan Li[1*], GanFeng Cao[1], WeiWei Liu[2], LiKai Wang[2], Nan Zhou[2]
[1]Xuzhou University of Technology, Xuzhou 221000, Jiangsu, China.
[2]Traffic police detachment of Xuzhou Public Security Bureau, Xuzhou 221000, Jiangsu, China.
Corresponding Author: Dan Li, Email: 1026673665@qq.com

Abstract: The purpose of this study is to propose a deep learning-based solution, aiming at the problem of insufficient accuracy and real-time performance in traffic target tracking and trajectory prediction technology. We used YOLOv8 for real-time target detection, combined with the multi-target tracking track algorithm to achieve accurate tracking of traffic targets. At the same time, the trajectory prediction through the long-and short-term memory network (LSTM) can effectively deal with the dynamic changes of traffic flow. The experimental results show that the method tracks better than conventional algorithms in multiple traffic environments, with better robustness and real-time performance. Moreover, this study explores the impact of data enhancement and hyperparameter optimization on model performance, which provides new ideas and methods for the implementation of intelligent transportation system.
Keywords: YOLOv8; Traffic target tracking; Trajectory prediction; Multi-target tracking; Long-and short-term memory network (LSTM)

1 INTRODUCTION

With the acceleration of urbanization, traffic congestion and frequent traffic accidents have become important problems restricting urban development. To solve these problems, intelligent transportation systems emerge. Among them, target tracking and trajectory prediction, as an important part of the intelligent transportation system, are of great significance for improving road safety, alleviating traffic congestion, optimizing traffic management and other aspects [1].

Traditional methods of target tracking and trajectory prediction, such as image processing and computer vision technology, often struggle to achieve ideal results when facing complex traffic environments [2]. For example, the performance of traditional methods can be severely affected under high vehicle density, changeable traffic rules, and complex weather conditions. The introduction of deep learning technology has brought about a new breakthrough in this field [3].

Deep learning realizes the abstract modeling of the data and the knowledge representation [4] by simulating the working mode of the neural networks in the human brain. In target tracking and trajectory prediction, deep learning can automatically extract the feature information of targets, such as speed, acceleration, direction and other, thus improving the accuracy of tracking and prediction [5]. In addition, deep learning can also learn the behavior patterns and traffic rules of traffic participants by training a large number of data models, and further improve the accuracy of trajectory prediction.

The whole simplified process of traffic target tracking and trajectory prediction is shown in Figure 1.

Figure 1 Simplified Flow Chart of the Traffic Target Tracking and Trajectory Prediction System

2 INTRODUCTION TO THE YOLOV8 ALGORITHM

YOLO (You Only Look Once) is an advanced real-time target detection algorithm, proposed by Joseph Redmon et al. in 2015 for [6]. YOLO is unique in that it treats the object detection problem as a regression problem rather than the classification problem [7]. It processes the entire image through a single neural network, divides the image into multiple regions, and predicts bounding boxes and category probabilities for each region. Due to its high speed and high accuracy, YOLO has been widely used in many real-time applications, such as autonomous driving, video surveillance, and robotics.

Since its launch, the YOLO algorithm has undergone several iterations, from the initial YOLOv1 to the latest YOLOv8, each generation has improved in performance and functionality. YOLOv8 Is the latest version of the YOLO series, developed by the Ultralytics team. It reaches new heights in terms of accuracy and speed and is suitable for a variety of target detection tasks.

YOLOv8 Introseveral new features and optimizations, including advanced backbone network and neck architecture,

© By the Author(s) 2024, under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0).

with improved feature extraction and target detection performance. It uses the anchor-free segmentation head, which has higher accuracy and more efficient detection process than the anchor-based method. Moreover, YOLOv8 performs well in maintaining accuracy and speed balance and is suitable for a variety of real-time target detection tasks12.

In terms of traffic target tracking, YOLOv8 is particularly well. It can detect and track traffic targets such as vehicles, pedestrians and bicycles in real time. Combined with the track algorithm in Ultralytics, YOLOv8 can achieve multi-target tracking, ensuring that each target can also be accurately identified and tracked in complex traffic environments3.

In addition, YOLOv8 can be combined with LSTM (long and short-term Memory Network) for trajectory prediction. This combination can cope with the dynamic changes of traffic flow and provide more accurate trajectory prediction, thus improving the efficiency and safety of traffic management3. For example, in the traffic monitoring system, YOLOv8 can monitor the traffic flow in real time, automatically identify the behavior and state of the traffic targets, and provide strong support for traffic management.

In short, YOLOv8 has not only performed well in target detection and tracking, but also demonstrated great potential in practical applications such as traffic management. Its high efficiency and accuracy make it an important tool in the intelligent transportation system, which helps to improve traffic safety and optimize traffic management. Therefore, we use yolo algorithm for target tracking.

2.1 YOLOv8 Algorithm

YOLOv8 Is the latest version of the YOLO series and has made remarkable progress in the field of target detection. YOLOv8 It is mainly composed of the backbone network (Backbone), the feature fusion layer (Neck), and the detection head (Head). YOLOv8 The network structure is shown in Figure 2 [8]:

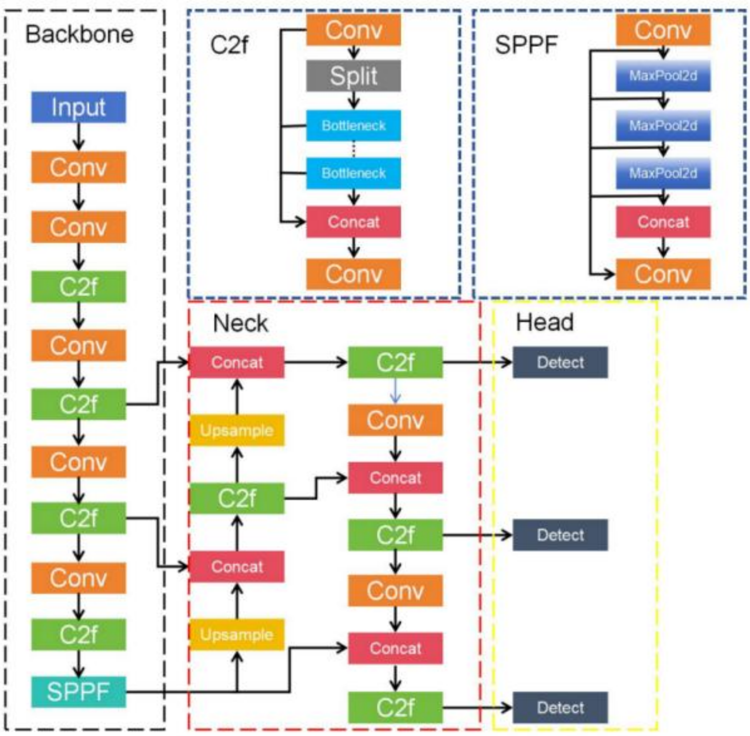

Figure 2 The YOLOv8 Network Structure Diagram

2.1.1 Trunk network (backbone)

The backbone network of YOLOv8 is responsible for extracting the features from the input images. It adopts an efficient convolutional layer structure, combining the advantages of deeply separable and standard convolution, to improve the computational efficiency and reduce the number of parameters . Depth separable convolution significantly reduces the computation [9] by splitting standard convolutions into deep and point convolutions.

YOLOv8 The CSP (Cross-Stage Partial Network) idea continues in Backbone, but the C3 module in YOLOv5 is replaced with the C2f module, and this improvement brings further lightweight. The C2f module improves both efficiency and speed by reducing the computation and model parameters. YOLOv8 The SPPF (Spatial Pyramid Pooling-Fusion) module was also used to enhance the feature expression capacity.

2.1.2 Feature fusion layer (Nk)

YOLOv8 Feature pyramid network (FPN) and path aggregation network (PAN) are used to integrate multi-scale features to enhance the detection ability of targets of different sizes. FPN uses the features of different levels through the top-down path, which improves the detection ability of small targets. PAN further increases the bottom-up path on the basis of feature fusion to achieve more fine-grained feature aggregation.

In contrast to YOLOv5, YOLOv8 removed the convolution structure during the upsampling phase of PAN-FPN and replaced the C3 module with the C2f module. This adjustment not only reduces the computational amount, but also improves the feature fusion and context capture capabilities.

2.1.3 Test head (head)

The test head of YOLOv8 is responsible for generating the final test results, including the location and category of the target. It adopts the Anchor-Free mechanism, avoids the design complexity of the anchor frame, and directly predicts the center point, width, height, and classification information of the target through the convolutional layer.

YOLOv8 Also introduces the Decoupled-Head (decoupling head) design, a design method that separates the classification and regression tasks. Model performance and accuracy were improved by handling the classification and regression tasks separately.

2.2 ByteTrack Algorithm

The ByteTrack algorithm is a tracking method based on the tracking-by-detection paradigm, that is, the location information of the target is obtained through the target detection algorithm, and then the data information is used for the target tracking [10]. The biggest innovation of this algorithm lies in the utilization of the low score detection box.

ByteTrack The algorithm is as follows:

2.2.1 Detection box classification

For all detection box information obtained by the detector, they are divided into two parts according to the confidence: the detection score higher than threshold high is classified as D, the detection score below threshold Tlow is classified as Dlow.

2.2.2 Trajectory prediction

For all tracks in the trajectory set T, the Kalman filter is used to predict their coordinates in the current frame.

2.2.3 First match

Match the high score detection box Dhigh and all tracks T. The intersection ratio (IOU) of the position in the current frame in the current frame is calculated, and then matched by the Hungarian algorithm. For rejected matches with IOU less than 0.2, the unmatched detection box is stored in Dremain; the unsuccessful track T is stored in Tremain.

2.2.4 Second match

For the low score detection box Dlow (for example, the object with severe occlusion in the current frame) and the remaining track Tremain, the method is similar to the first match. The two unsuccessful tracks were deposited in Tre-remain, and the unmatched low score detection boxes were removed.

2.2.5 Track processing

For the trajectory in Tre-remain, it is considered to lose the target temporarily, and put it into Tcost. If a trajectory in Tcost exists for more than a certain time (e. g., 30 frames), remove from T; otherwise, continue saving. If late matches to, it is also removed from Tcost. For detection in Dremain with a score above some threshold E and that has survived more than two frames, it is initialized as a new trajectory .

The flowchart is shown in Figure 3:

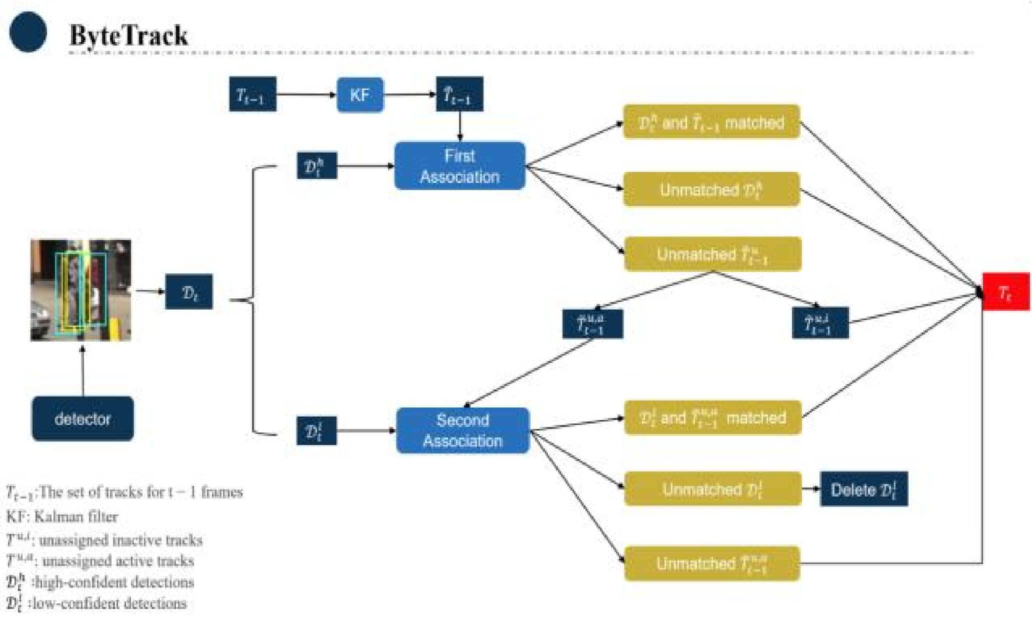

Figure 3 The ByteTrack Flow Chart 3 Experimental Process

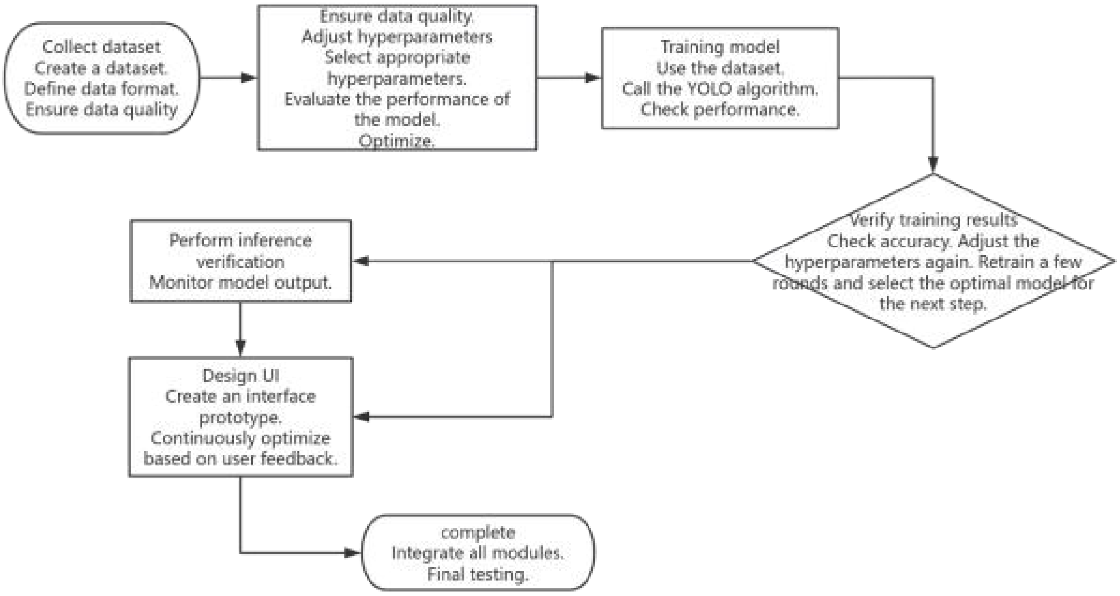

Figure 4 Experimental Procedures

3 SYSTEM IMPLEMENTATION

3.1 Source of the Data Set

In this project, we mainly use the datasets from open source dataset platforms such as Roboflow and Baidu Feiyang to study vehicle and pedestrian target tracking. Roboflow Is a platform focused on computer vision data set management and processing, providing rich resources of public datasets, such as COCO, ImageNet, Open Images, etc. These datasets cover a wide range of application scenarios and categories, and are able to meet the needs of different tasks。

With Roboflow, we can easily access and manage the datasets. First, we searched and selected the dataset suitable for the project requirements on the Roboflow platform. Then, the dataset was annotated and preprocessed with Roboflows tools, including image cropping, rotation, zoom and other operations to ensure the quality and consistency of the dataset. Finally, we export the processed dataset into the format required for the YOLOv8 model to facilitate subsequent model training and testing.

Baidu paddle (PaddlePaddle) also provides a large number of high-quality open source data sets, especially in Chinese scenarios. We acquired the relevant vehicle and pedestrian dataset through the flying paddle platform, and used the tools it provided for data enhancement and preprocessing to improve the generalization capability of the model。

By using these high-quality open-source datasets, we are able to effectively study vehicle and pedestrian target tracking and achieve good results in practice. In the YOLOv8 target tracking project, the collection of datasets is a crucial step. Our data set contains about 2,700 pictures of traffic and pedestrians (some samples are shown below):

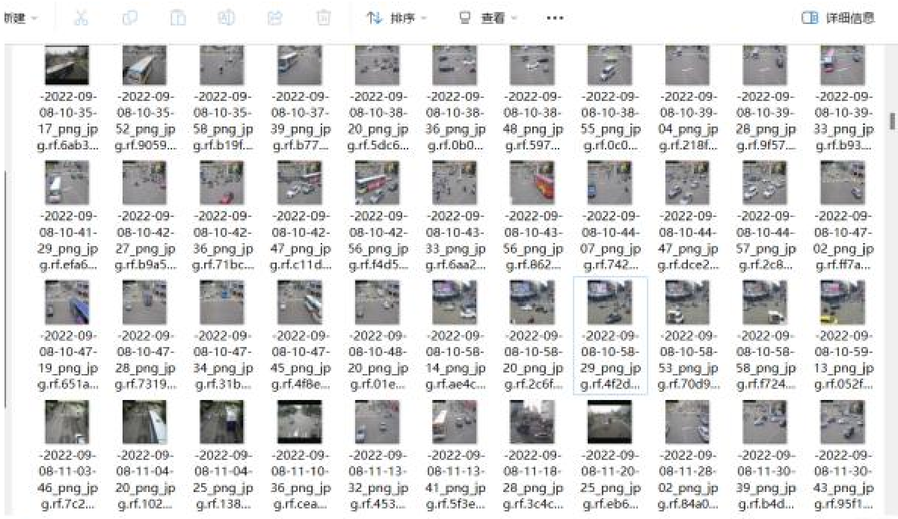

Figure 5 Part of the Sample Data

It aims to provide rich training materials for the model. The quality and diversity of the data set directly affect the performance of the model, so we focus on the following aspects when collecting the data:

First, to ensure the diversity of the datasets. The data set covers a variety of different scenarios and conditions, such as different weather (sunny, rainy, snowy), different times (day and night), and different traffic conditions (peak, off-peak). This helps the model to perform well in a variety of practical applications.

Secondly, make accurate annotation. Each image needs to be accurately labeled, including the targets bounding box and category label. We used the X-Anylabeling annotation tool for manual annotation, with part of the data obtained from the open-source dataset platform Roboflow. The accuracy of the annotation directly affects the training effect of the model, which requires special attention.

After data collection, the data need to be cleaned to remove ambiguous, incomplete, or duplicate images. Make sure that each image is clearly visible and that the target object is not obscured or partially missing.

Finally, the dataset was divided into the training, validation, and test set. The data partitioning ratio was 70% for training, 20% for validation, and 10% for testing. This allows the performance of the model to be evaluated during training and adjusted accordingly.

Through the above steps, the quality and diversity of the data set can be ensured, laying a solid foundation for the training of the YOLOv8 target tracking model.

In the YOLOv8 target tracking project, image preprocessing is a key step to improve the model performance. Efficient preprocessing can not only improve the training efficiency of the model, but also significantly improve its accuracy. The various aspects of image preprocessing are described in detail below, especially the implementation of data enhancement operations.

First, the adjustment of the image size is the fundamental step of the preprocessing. YOLOv8 Model generally require a uniform size of the input image. You can use the cv2.resize function in the OpenCV library to adjust all the images to a uniform size (like 640x640 pixels). This operation not only reduces the computational complexity, but also ensures the consistency of the model inputs.

Second, normalization processing is one of the key steps of preprocessing. Normizing the image pixel values to the range of 0 to 1 helps to accelerate the model convergence and improve training. YOLOv8 The normalization step is performed automatically in its preprocessing pipeline, so no manual processing is required. In addition, we also do data enhancement operations such as random cropping, rotation, flip, brightness adjustment, color space conversion and other data, which enhance the generalization ability of the model.

3.2 Design and Implementation of the System

The research system of traffic target tracking and trajectory prediction technology based on deep learning aims to realize the automatic identification and analysis of the goals in traffic scenarios. The system can monitor traffic images or videos in real time, automatically identify the behavior and status of traffic targets, such as vehicle speed, location changes, etc., so as to help traffic managers better understand traffic flow and abnormal conditions, and find potential traffic problems in time, so as to provide more effective management and response measures. This system provides an important support for improving traffic safety and optimizing traffic management. System initial interface is divided into three pieces, the top is the detection target type, target number, frame rate and use model, the image of the left sidebar mainly for the control panel, the middle for the system and the corresponding detection results, the control panel includes the image of the import, video import, batch detection folder pictures, connected to the surveillance camera, the right of the module of the original image and the left display detection results. The UI interface is shown in Figure 6:

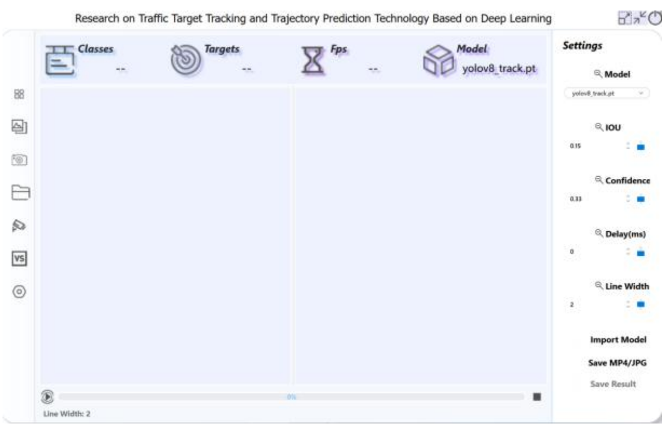

Figure 6 The UI Interface

Click the open folder in the system interface: the user clicks a left picture button on the graphical user interface (UI) of the system to open the folder. After folder open, select the picture or video to detect, import into the interface, interface will set the default model, if you need to modify the detection model, click the left bottom set the button will pop, can change the model and modify the corresponding parameters, if you want to add a new model can find the ptfiles folder in the folder home directory, can want to add the model into the folder to call it in the interface, and then click start will directly call yolov8 model to detect the detection target. The successful image or video will be displayed on the interface on the right. If you need to save the detected picture or video, there is Save MP4 / JPG or Save Result at the bottom of the right sidebar. After clicking, select the position to be saved, and then click to confirm that the preservation is successful. Click the empty button (the square button), and the system will empty the detection content and restore it to the initial interface.

Figure 7 UI Second Interface

As shown in Figure 7. The second interface of the UI interface is to compare the two Models, so as to simply and clearly show the advantages and disadvantages of the two models, so as to improve the shortcomings of the model, so as to achieve the desired results.

3.3 Target Identification

In the process of target detection, the detection model can accurately detect the required, detect the car and people on the road, and the detection target with a box, and display the category label on the box of the detected object, for example, "people", "car" and confidence score, show the confidence of the test results, indicating the confidence of the model to the target detection. At the same time to the YOLOv8 reasoning results (after the boundary box picture), and then add tracker results (to each identified target plus ID), at the same time using the append method to add the current center of the target coordinates, keep the nearest 30 tracking points, in the video will show 30 points like a straight line, so as to draw the tracking path, tracking path is formed by connecting the points in the tracking history to track, finally achieve the tracking of the target track.

3.4 Output the Identification Results

When the system detects, the detection target is circled with a box, and the category label on the box shows the category of the detection object, such as "person", "car" and confidence score show the confidence of the detection result, and add an ID to each detection target, so as to show its movement trajectory. Above the system, the detected target species, number of target, and fps are displayed in the interface. Depending on the specific needs, these results can be further used for decision-making or applications in other systems. In this experiment, we mainly carried out target detection on cars and people, and target detection on other objects. This requires us to change the required dataset during training, adding a large number of different data for the training of the model. As shown in Figure Figure 8 and 9:

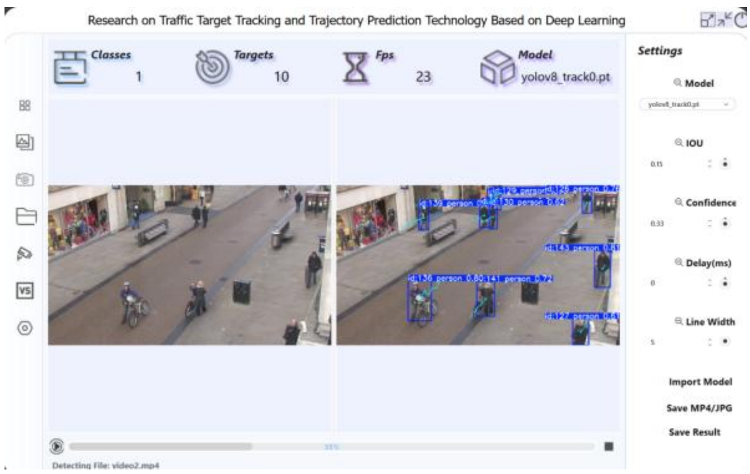

Figure 8 Track Results for Pedestrians

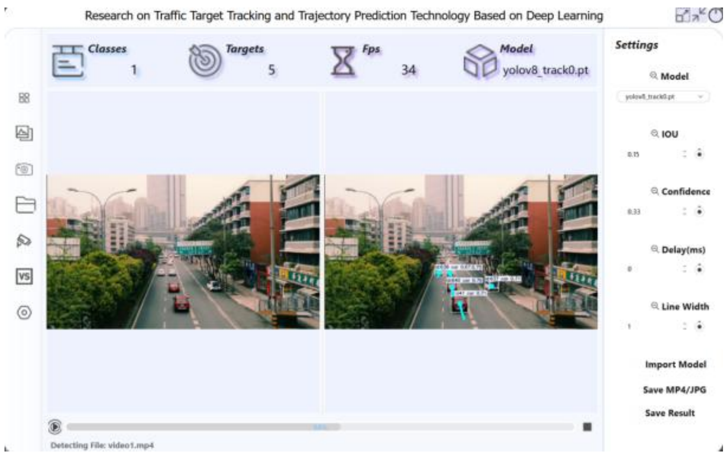

Figure 9 For the Vehicle Tracking Results

4 CONCLUSION

This study focuses on improving the accuracy and real-time performance of traffic target tracking and trajectory prediction techniques, and proposes a deep learning-based solution. We used YOLOv8 algorithm for real-time target detection and combined with SORT algorithm to realize multi-target tracking to cope with the dynamic changes of traffic flow. The results show that the proposed method performs better than conventional algorithms in multiple traffic environments, showing good robustness and real-time performance.

In the study, the collection and processing of the datasets is the critical step. The dataset contains about 2700 images of traffic and pedestrians covering different weather, timing and traffic conditions to ensure the diversity and generalization capability of the model. After accurate annotation and cleaning, the data are divided into the training set, validation set and test set. Image preprocessing includes image size adjustment, normalization processing as well as data augmentation operations such as random cropping and rotation to improve model performance.

The system interface design is simple and clear, divided into the detection target type, target number, frame rate and the model display area used, as well as the control panel and the image or video display area. The control panel allows the user to import pictures and videos for batch detection. The system can monitor traffic images or videos in real time, automatically identify the behavior and status of traffic targets, and provide support for traffic management.

This study also explores the influence of data enhancement and hyperparameter optimization on model performance, which provides new ideas and methods for the implementation of intelligent transportation system. The results show that this deep learning solution is very useful in real traffic environments and helps to improve traffic safety and optimize traffic management. In short, deep learning-based traffic target tracking and trajectory prediction technology is a research field with promising application prospects. In this experiment, we have obtained some valuable research results, but also found some problems and deficiencies. In the future, we will continue to study this technology in depth, constantly improve the algorithm performance, expand the application scenarios, strengthen the combination with practical application, and make greater contribution to the intelligent development in the field of transportation.

COMPETING INTERESTS

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

FUNDING

This work was supported in part by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (Grant: 23KJA520013), Xuzhou Engineering College Student Innovation and Entrepreneurship Training Program Project (Grant: xcx2024187), the Sixth "333 project" of Jiangsu Province.

REFERENCES

[1] Kejia Zhu, Yikuan Zhang, Qiang Liu, et al. Research on Application Method of Object Detection in Video. 2022 International Conference on Cloud Computing, Big Data and Internet of Things. 2022, 10, 22.

[2] Li Liu, Wanli Ouyang, Xiaogang Wang, et al. Deep Learning for Generic Object Detection: A Survey. International Journal of Computer Vision, 2020.

[3] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, et al. Object Detection in 20 Years: A Survey. CoRR, 2019.

[4] Kequan Li, Yan Chen, Jiachen Liu, et al. Overview of object detection algorithms based on deep learning. Computer Engineering, 2022, 48(7): 1-12.

[5] Ghania Zidani, Djalal Djarah, Abdslam Benmakhluof, et al. Optimizing Pedestrian Tracking for Robust Perception with YOLOv8 and Deepsort. Applied Computer Science, 2024, 20(1): 72-84.
[6] Shuxin Yang, Yang Xie, Jianqing Wu, et al. CFE-YOLOv8s: Improved YOLOv8s for Steel Surface Defect Detection. ELECTRONICS, 2024, 13(14).
[7] Mukaram Safaldin, Nizar Zaghden, Mahmoud Mejdoub. An Improved YOLOv8 to Detect Moving Objects. IEEE Access, 2024, 12: 59782-59806.
[8] Ji-Yuan Ding, Wang-su Jeon, Sang-Yong Rhee, et al. DM-YOLOv8: Improved Cucumber Disease and Insect Detection Model Based on YOLOV8. 2024.
[9] Shuai Xu, Jiyou Fei, Geng Zhao, et al. CCL-YOLO: Catenary Components Location Based on YOLO and Gather-Distribute Mechanism. IEEE Access, 2024: 1.
[10] Jing Li, Zhengjun Xu, Liang Xu. Vehicle and Pedestrian Detection Method Based on Improved YOLOv4-tiny. Optoelectronics Letters, 2023, 19 (10): 623-628.