Academic Journal of Emerging Technologies
ISSN Print: 3104-4417
ISSN Online: 3104-4425
CD ISSN: 3104-4433
https://doi.org/10.63313/AJET.9023

Research on Multi-source Video Streaming Vehicle Object Detection and Parallel Statistics Based on YOLO-OBB

Qian Jiang
School of Automotive and Traffic Engineering, Liaoning University of Technology, Jinzhou 121000, China
Email: 2398311101@qq.com

How to cite this paper: Jiang, Q. (2025). Research on Multi-source Video Streaming Vehicle Object Detection and Parallel Statistics Based on YOLO-OBB. *Academic Journal of Emerging Technologies*, 1(3), 68-77. ISSN Print: 3104-4417; ISSN Online: 3104-4425.
https://doi.org/10.63313/AJET.9023
Published: 2025-11-25

Abstract

Based on the YOLO-OBB rotating object detection model, a statistical analysis system for multi-intersection vehicles is proposed. The system realizes the accurate analysis of vehicles running in complex intersection scenarios by analyzing the characteristics of the vehicle's driving trajectory and the characteristic change law of the steering angle in the intersection area, and integrating the rotating bounding box detection technology and the innovative dual-threshold steering recognition algorithm. The system can not only accurately identify the entrance direction of the vehicle, but also judge the steering behavior and complete the model classification in real time. To improve efficiency, the system adopts a multi-threaded parallel architecture that can analyze 9 high-definition video streams simultaneously, and establishes an intuitive visual interface through the PySide6 framework. The experimental results show that on the 1920×1080 resolution video stream, the vehicle recognition accuracy of the four intersections is 92.2%, and the accuracy of vehicle and steering judgment is also more than 80%, which can meet the demand for real-time and accurate vehicle flow data in actual scenarios, and provide data support for decisions such as signal light optimization and road planning.

Keywords

YOLO-OBB; statistical analysis of multi-intersection vehicles; dual-threshold steering recognition algorithm; multi-threaded processing; PySide6

1. Introduction

With the acceleration of global urbanization, traffic congestion has become a core problem restricting the sustainable development of cities. Early studies have shown[1]that residents in major cities around the world lose up to 150 hours per year due to traffic congestion, and traffic congestion costs economic growth at about 2% of GDP, plus related losses such as traffic accidents and environmental pollution.

According to data from the Traffic Management Bureau of the Ministry of Public Security[2], as of the end of January 2024, the number of motor vehicles in the country reached 460 million, including 359 million cars and 550 million motor vehicle drivers, including 515 million car drivers. In this context, traditional traffic management methods face severe challenges: manual statistics are inefficient, typical intersections need to be equipped with 3-5 traffic managers, and only 8 hours of data collection can be completed in a single day, and the subjective error rate is as high as 15%-20%; The update cycle of static traffic survey data is long, and it cannot adapt to the dynamic changes of the urban road network. The lack of real-time data support for multi-intersection collaborative control leads to low regional traffic efficiency.

In the early days, coil detectors[3]were used for data acquisition and then for traffic management, but geomagnetic coil detectors are not only susceptible to environmental interference but also extremely expensive to maintain. Although the use of sensors such as microwave radar[4], lidar[5], and video detector[6]for data collection has become mainstream, there are obvious shortcomings in accurately identifying vehicle steering behavior. Although the video analysis system,such as YOLOv5[7]、Faster R-CNN[8]、Fast R-CNN[9]has certain advantages in target positioning speed, there are problems such as insufficient direction sensitivity and ineffective detection of vehicle overlapping scenes, which are further amplified in complex scenarios such as intersections.

At the same time, most of the existing studies focus on single-camera scene analysis, lack of collaborative processing capabilities for multi-monitoring point data, and the multi-channel video processing architecture is inefficient. Therefore, the development of high-precision and low-latency multi-video collaborative analysis systems has become a technical problem that needs to be overcome urgently in the field of intelligent transportation.

Liu Chao[10]et al. explored a method to construct a long-term wide-domain vehicle trajectory dataset based on roadside fixed multi-camera data. A cross-frame trajectory association algorithm is proposed, which effectively copes with the trajectory interruption caused by occasional vehicle missed detection by setting up a temporary data cache area, and conducts self-inspection in combination with the associated trajectory to find and correct the problem trajectory in time, which improves the reliability and integrity of the data.

Jocher et al.[11]proposed to solve the rotation detection problem for the first time while maintaining real-time performance through angular decoupling head and innovative Gaussian distribution modeling, which provides a technical basis for vehicle flow analysis. However, there are still obvious shortcomings in the application of existing studies in traffic scenarios: steering judgment relies on a single position information and ignores vehicle attitude changes; Lack of tracking robustness design for intersection occlusion.

In response to the above challenges, this paper proposes systematic solutions. At the algorithm level, the dual-threshold decision-making mechanism of position and angle is innovatively developed, and the position offset vector and angle change are integrated to effectively solve the problem of single feature misjudgment.

2. Overall design and implementation of the system

2.1. System architecture design

The system adopts a modular hierarchical architecture, forming a closed loop from video input to statistical result output. The system core consists of six modules: the video input interface receives multiple video streams and supports RTSP/RTMP protocols; The frame preprocessing module optimizes image quality, including denoising, gamma correction, ROI cropping, and size normalization. The OBB detection module adopts the improved YOLOv8-OBB model and outputs a detection frame with a rotation angle. The multi-target tracking module integrates high and low confidence detection frames, uses Kalman filtering to predict the target position and correlate cross-frame data.

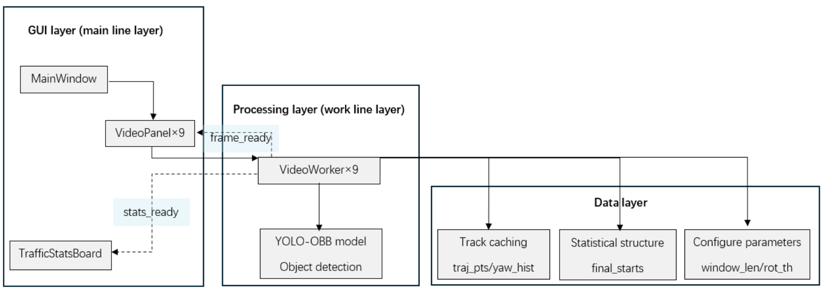

Figure 1. System process architecture diagram

2.2. Maintaining the Integrity of the Specifications

2.2.1. Rotary box detection technology

The system uses the YOLOV8-OBB (Oriented Bounding Box) model for vehicle detection, which breaks through the limitations of traditional horizontal bounding boxes. In the process of video processing of each frame, the model first extracts features from the input image and generates the parameters of the rotating bounding box. Normalized angular value through equation (1):

$$\theta_{norm} = (\theta + 180) mod 360 - 180 \quad (1)$$

These parameters accurately describe the vehicle's position and orientation in a two-dimensional plane, especially suitable for complex scenarios at dense intersections. During the detection process, the model extracts multi-scale features through deep convolutional neural network, uses the anchor box mechanism to predict the parameters of the rotating box, and uses the non-maximum suppression algorithm to eliminate redundant detection results. The final output rotation box not only contains the vehicle's position information, but also accurately reflects the vehicle's driving direction, laying the foundation for subsequent trajectory analysis.

2.2.2. Dual behavior analysis algorithm

Aiming at the problem of identifying vehicle behavior at complex intersections, the system innovatively integrates the dual behavior analysis algorithms of geometric position and rotation angle, which improves the accuracy and consistency of behavior judgment.

(1) Trajectory caching mechanism: The system maintains a fixed-length trajectory cache queue for each vehicle, that is, establishes a "dynamic trajectory memory", which records its movement trajectory in real time by continuously tracking the position and direction changes of the vehicle:

$$Trajectory = \left\{ \left(P_t, \theta_t \right) \right\}_{t=1}^{T} \tag{2}$$

The default length T=10 frames records the coordinates of the center point of the vehicle in detail， $P_t = \left(x_t, y_t \right)$ and yaw angle derived from the OBB detection frame θ_t. The sliding window design can not only understand the current operating characteristics of the vehicle, but also make comprehensive research and judgment based on its historical trends, ensuring the timeliness and accuracy of behavior analysis.

(2) Geometric position analysis: Based on the preset intersection area division, the module divides the picture into four quadrants according to the center point: left, right, upper and lower, and infers its theoretical turning direction according to the quadrant position of the vehicle . Q_{t_s} and Q_{t_e} at the start frames t_s and end frames t_e of the trajectory window.The correspondence between position changes and behavior types is established through predefined geometric mapping rules M (e.g., M(Left→Bottom)=Right Turn). The preliminary judgment results of geometric position analysis C_{ego} are given by the following equation:

$$C_{ego} = M\left(Q_{t_s}, Q_{t_e} \right) \tag{3}$$

Damit hinein $C_{ego} \in \left\{ Left\,Turn, Straight, Right\,Turn \right\}$

(3) Analysis of Rotational Angles: When geometric position analysis cannot clearly determine the behavior type due to the same starting and ending quadrants, the system activates the rotational angle analysis as a supplementary module. This component calculates the maximum variation $\Delta\theta_{max}$ and direction of the vehicle's yaw angle θ within the trajectory window. If the angular change exceeds a predetermined threshold, the behavior is classified as a turning action, with the direction of change indicating a left or right turn.

Initially, we compute the sequence of angular changes during the specified window period:

$$\Delta\theta_t = \theta_t - \theta_{t-1}, \text{for t} \in [t_s + 1,\ \ t_e] \tag{4}$$

Then calculate the cumulative angular change $\Delta\theta_{total}$ or maximum step change $\Delta\theta_{max}$ within the window. When the absolute value of the angle change exceeds the set threshold $\Theta_{thresh} = 30°$, it is judged as a turning behavior. The direction of change determines whether to turn left or right:

$$C_{angle} \begin{cases} Left\,Turn, & if\,\Delta\theta_{total} > +\Theta_{thresh} \\ Right\,Turn, & if\,\Delta\theta_{total} < -\Theta_{thresh} \\ Straight, & otherwise \end{cases} \tag{5}$$

(4) Decision tree integration mechanism

The results of the dual analysis are fused through a single decision tree to ensure the reliability of the final verdict:

① Geometric position analysis conclusion C_{ego} is preferred: when $C_{angle} = Straight$ (i.e., the start and end quadrants are the same), the rotation angle analysis results C_{angle} are enabled as the final verdict.

② If both are invalid or conflicting, it is conservatively judged to be straight.

To eliminate instantaneous jitter and ensure the stability of the behavioral state, the system introduces a state transfer smoothing mechanism. The final state of the current frame S_t is determined by the following formula:

$$S_t = \begin{cases} C_{cand}, & if\ C_{cand} = S_{t-1} \text{且} cnt \geq 3 \\ S_{t-1}, & otherwise \end{cases} \tag{6}$$

Here, C_{cand} represents the candidate behavior identified by the frame fusion algorithm, and cnt indicates the number of consecutive frames in which this candidate behavior appears.

The dual analysis mechanism significantly improves the accuracy of behavior recognition in challenging scenarios such as vehicle occlusion and trajectory interruption by combining the macroscopic motion trajectory and microscopic attitude changes of the vehicle.

2.2.2. Multi-threaded parallel processing architecture

To support the synchronous real-time analysis of multi-channel video, an efficient and robust multi-threaded parallel computing architecture is designed. The architecture takes "parallel pipelines" and "intelligent resource scheduling" as the core ideas, and builds a complete data processing thread by allocating each video stream to an independent worker thread.

(1) Parallel assembly line design

Each video channel is equipped with a dedicated processing thread, forming a modular, autonomous analysis unit. Each unit follows the assembly line operation mode:

the video decoding module analyzes the original data flow in real time, the rotating frame detection algorithm accurately locates the vehicle boundary and attitude, the trajectory tracker constructs the vehicle motion sequence based on the spatio-temporal association, and finally the behavior analysis module fuses multi-frame features to complete the steering judgment.

$$Thread_i = Video\ decode \rightarrow OBB\ detection \tag{7}$$

If the number of video streams processed concurrently by the system is N (N=9 in this system), a total of N such worker threads are created. All threads are physically parallel and executed by operating system scheduling.

(2) Intelligent resource scheduling strategies

The system monitors global computing resources (such as GPU utilization U_{GPU}) in real time and implements dynamic resource scheduling policies, and its decision-making logic can be described in the following formulas:

① Dynamic frame reduction policy: When GPU utilization is detected to exceed a safety threshold U_{thresh} , the system automatically reduces the processing frame rate F_i for non-user-focused channels:

$$F_i^{new} = \begin{cases} F_i^{original} \times \alpha, & if\ channel\ i\ is\ inact \\ F_i^{original}, & x \geq 0 \end{cases} \tag{8}$$

Here, α is the frame reduction factor $0.5 \leq \alpha < 1$.

② Resource focus strategy: When a user double-clicks on a video to zoom in, the system dynamically allocates more computing resources to the channel, which may manifest as increasing its processing priority or allocating more CPU time slices to ensure the smoothness and real-time analysis of the video.

(3) Performance models and evaluations

The theoretical processing power of this architecture can be evaluated with a simplified model. If the processing speed of a single channel is f_{single} FPS, then ideally, the total processing speed F_{total} of N channels of video is related to the number of CPU cores c:

$$F_{total} \approx \min(N, C) \times f_{single} \times \eta \tag{9}$$

Among them, the parallel efficiency factor η includes thread synchronization, resource competition and other overheads $(0.5 \leq \eta < 1)$.

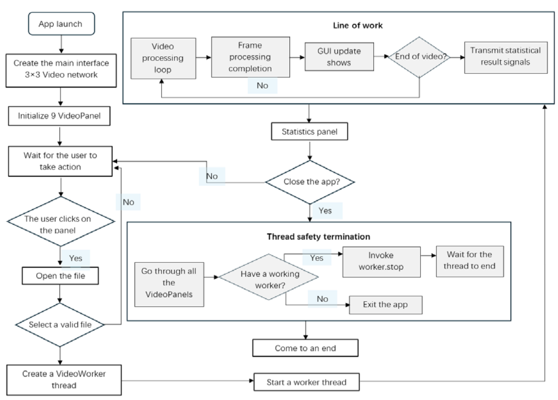

Figure 2. Flowchart of multithreaded parallel architecture

2.3. User interface design

The system's user interface aims to provide users with an interactive environment for multi-intersection traffic flow monitoring and analysis. The entire interface consists of two core functional modules: a video panel matrix and a multi-layer statistical table. The two work together to realize everything from video data visualization to statistical results.

Multi-level statistical tables are the core of the system's data output. The structured hierarchy can deeply organize and summarize vehicle information. These tables follow the data analysis dimensions of traffic management and are divided into three layers: The spatial dimension divides data by the direction of entry and defines the source of vehicles; The behavioral dimension is subdivided by steering type to express driving intentions. The attribute dimensions are classified by vehicle type. The table data is linked in real time to the video processing process to ensure the dynamic update of the analysis results, guarantee the real-time and accuracy of the statistical information, and at the same time meet the dual demands of efficient monitoring and in-depth analysis.

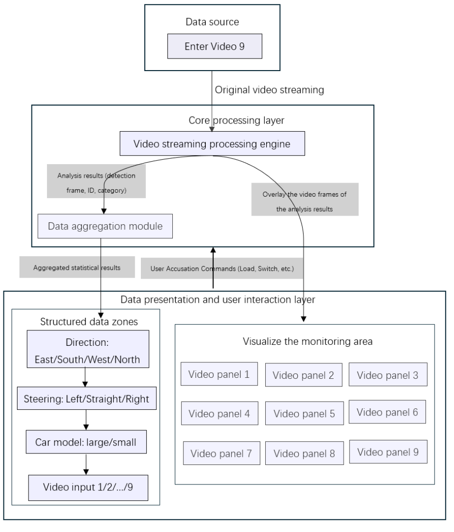

Figure 3. Interface architecture diagram

3. Experiment and Result Analysis

3.1. Video capture scheme

In order to evaluate the actual performance of the YOLO-OBB model in the vehicle trajectory analysis and statistical system of multiple intersections, the signal control intersections of Xueyuan Road and Beixin Road, Xueyuan Road and Xishan Road, Dali North Road and Beixin Road, and Dali North Road and Xishan Road in Tangshan City were selected for field verification. DJI drones were used to take aerial photos of the intersection and obtain traffic flow video data. The shooting settings are as follows: video resolution is 1920×1080 with a frame rate of 30 fps. Data acquisition was carried out during the day on a sunny weekday and was shot for 5 minutes. In the model verification stage, the manual counting results are used as the benchmark for accuracy comparison.

3.2. Analysis of experimental results

Based on the established system, the statistical test is carried out on the traffic flow in the intersection scenario, taking the data of the intersection of Xueyuan Road and Beixin Road as an example, the detection results of each entrance steering and vehicle identification are shown in Table 1, and the total detection results of the four intersections are shown in Table 2. Among them, the actual traffic flow is based on manual counting, and the system detection accuracy is calculated based on this benchmark.

Experimental data show that the overall accuracy of the model in the four inlet directions is more than 80%, and the average reaches 93.8%. Among them, the detection effect of the southern import is the best, with an accuracy rate of 97.7%. Although there are a small number of false detections and missed detections at each entrance, it is analyzed that the main reason is that the camera is not completely vertical to the ground during the aerial photography process, and there is a certain tilt, which leads to the reduction of the target size of the vehicle in the intersection area far away from the lens, which increases the difficulty of feature recognition. Nevertheless, from the perspective of the detection rate of the four intersections, the average detection accuracy of the model is close to 92.2%, indicating that it has good detection accuracy and practicability in practical application scenarios.

Table 1. Test results for vehicles

<table>
<tr><th>Intersection</th><th>Veer</th><th>Model</th><th>Detect traffic per vehicle</th><th>Actual traffic volume per vehicle</th><th>False detection rate per vehicle</th><th>False detection rate per vehicle</th><th>Accuracy/%</th></tr>
<tr><td rowspan="6">East import</td><td rowspan="2">Left</td><td>Small car</td><td>17</td><td>20</td><td>0</td><td>3</td><td>85.0</td></tr>
<tr><td>Large car</td><td>2</td><td>1</td><td>1</td><td>0</td><td>50.0</td></tr>
<tr><td rowspan="2">Straight</td><td>Small car</td><td>71</td><td>78</td><td>2</td><td>5</td><td>71.0</td></tr>
<tr><td>Large car</td><td>6</td><td>4</td><td>2</td><td>0</td><td>66.7</td></tr>
<tr><td rowspan="2">Right</td><td>Small car</td><td>17</td><td>18</td><td>0</td><td>1</td><td>94.4</td></tr>
<tr><td>Large car</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td rowspan="6">West import</td><td rowspan="2">Left</td><td>Small car</td><td>9</td><td>10</td><td>0</td><td>1</td><td>90.0</td></tr>
<tr><td>Large car</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100</td></tr>
<tr><td rowspan="2">Straight</td><td>Small car</td><td>93</td><td>102</td><td>2</td><td>8</td><td>91.1</td></tr>
<tr><td>Large car</td><td>3</td><td>1</td><td>2</td><td>0</td><td>33.3</td></tr>
<tr><td rowspan="2">Right</td><td>Small car</td><td>14</td><td>13</td><td>1</td><td>0</td><td>92.8</td></tr>
<tr><td>Large car</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td rowspan="5">South import</td><td rowspan="2">Left</td><td>Small car</td><td>13</td><td>16</td><td>0</td><td>3</td><td>81.2</td></tr>
<tr><td>Large car</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100</td></tr>
<tr><td rowspan="2">Straight</td><td>Small car</td><td>43</td><td>50</td><td>1</td><td>6</td><td>86</td></tr>
<tr><td>Large car</td><td>2</td><td>1</td><td>1</td><td>0</td><td>50</td></tr>
<tr><td>Right</td><td>Small car</td><td>23</td><td>18</td><td>2</td><td>3</td><td>78.2</td></tr>
</table>

		Large car	4	2	2	0	50
North im-port	Left	Small car	11	15	0	4	73.3
		Large car	0	1	0	1	0
	Straight	Small car	36	36	0	0	100
		Large car	0	0	0	0	100
	Right	Small car	4	6	0	2	66.7
		Large car	0	0	0	0	100

Table 2. Inspection results at each intersection

Intersection	Detect traffic per vehicle	Actual traffic volume per vehicle	Accuracy/%
Intersection of Xueyuan Road and Beixin Road	369	393	93.8
Intersection of Xueyuan Road and Xishan Road	387	422	91.7
Intersection of Dali Road and Beixin Road	323	357	90.5
Intersection of Dali Road and Xishan Road	265	286	92.7

4. Conclusions

This paper proposes an innovative multi-intersection vehicle trajectory analysis and statistics system, which is based on YOLO-OBB, which effectively solves the problem of vehicle behavior recognition in complex intersection scenarios by cleverly integrating rotating box detection technology and dual behavior analysis algorithm. In a quad-core CPU environment, the system can process 9 channels of 1080P video in parallel at a speed of 28.5FPS, while stably controlling CPU utilization within 94%, achieving a perfect balance between computing resources and processing efficiency, which is significantly improved compared to the traditional horizontal bounding box method.

The experimental results show that the system architecture has good scalability and can dynamically adjust the number of concurrent channels according to different hardware configurations. Its memory guardian mechanism effectively avoids the risk of memory leakage in long-term operation through dynamic monitoring and optimization strategies, ensuring the sustainable and stable operation of the system. At the same time, the asynchronous communication mode avoids the blockage of threads while waiting for UI responses, ensuring the continuous and efficient operation of the processing pipeline.

References

[1] Liu Dongwen, Yang Keqiang, Lu Jiaming. Chinese Cities with "Capital Letters"[N]. Manager Daily, 2012-08-31.

[2] Ministry of Public Security of the People's Republic of China. In 2023, the number of motor vehicles in China will reach 435 million, and the total number of drivers will reach 523 million [EB/OL]. (2024-01-11) [2025-10-20].https://www.mps.gov.cn/n6557563/c9386285/content.html

[3] Parsa B A ,Taghipour H ,Derrible S , et al.Real-time accident detection: Coping with im-

balanced data[J].Accident Analysis and Prevention,2019,129202-210.DOI:10.1016/j.aap.2019.05.014.

[4] LYU N C，PENG L F，WU C Z，et al. Real-time crash-risk prediction model that distinguishes collision types [J]. China Journal of Highway and Transport，2022，35（1）：93-108.（in Chinese）

[5] GUAN L M，ZHANG Q，CHU Q L，et al. Roadside dual lidar data fusion based on improved ICP algorithm[J]. Laser Journal，2021，42（9）：38-44（. in Chinese）

[6] XUE Q W，JIANG Y M，LU J. Risky driving behavior recog90 nition based on trajectory data[J]. China Journal of Highway and Transport，2020，33（6）：84-94（. in Chinese）

[7] JOCHER G, STOKEN A, CHAURASIA A, et al. Ultralytics/YOLOv5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support[Z/OL]. (2021-10-29) [2025-10-20]. https://doi.org/10.5281/zenodo.5563715.

[8] REN S Q,HE K M,GIRSHICK R,et al.Faster R-CNN:Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligen ce,2017,39(6):1137-1149.

[9] Girshick R. Fast R-CNN[C]//Proceedings of the IEEE International Conference on Computer Vision（ICCV）,Santiago,Chile,2015:1440-1448.

[10] Liu Chao, Luo Ruyi, Liu Chunqing, et al. Vehicle Continuous Trajectory Construction Method Based on Roadside Multi-camera Video Target Correlation and Trajectory Stitching[J].Traffic Information and Safety,2023,41(03):80-91.DOI:CNKI:SUN:JTJS.0.2023-03-009.

[11] Jocher, G., Chaurasia, A. and Qiu, J. (2023) YOLO by Ultralytics. https://github.com/ultralytics/ultralytics Guo S, Li W, Zhang Q, et al. Vehicle pose estimation for turn detection in complex intersections[J]. IEEE Robotics and Automation Letters, 2024, 9(2): 1567-1574.