Traffic & Transportation

Road Guideline Detection Method Based on E-YOLOv5 Algorithm in Autonomous Driving

Wenshu LI[1], Yuxuan JI[2], Nan YU[3]

Original Scientific Paper
Submitted: 20 Jan 2025
Accepted: 21 May 2025
Published: 30 Mar 2026

[1] Corresponding author, liwenshu2003@163.com, College of Information Engineering, Shanghai Maritime University, Shanghai, China
[2] jiyuxuan2004@163.com, College of Information Engineering, Shanghai Maritime University, Shanghai, China
[3] yun@shmtu.edu.cn, College of Information Engineering, Shanghai Maritime University, Shanghai, China

Publisher:
Faculty of Transport and Traffic Sciences, University of Zagreb

ABSTRACT
To address the performance limitations of traditional road surface guideline detection models, this study proposes an optimised detection model based on the YOLOv5 algorithm. By integrating the convolutional block attention module and optimising anchor box parameters, the model enhances feature extraction capabilities for small targets, thereby improving detection accuracy and real-time performance. Experimental results demonstrate that the improved model exhibits superior performance in both simulated and real-world scenarios, effectively mitigating issues such as missed detections, false positives and positional deviations. It significantly enhances robustness in complex environments. This research provides an efficient and reliable vision-based detection solution for autonomous driving and smart logistics, advancing the practical application of computer vision technologies.

KEYWORDS
YOLOv5; attention mechanism; identification and detection; autonomous driving; neural network; image processing.

1. INTRODUCTION

With the advancement of artificial intelligence and the development of 5G technology, autonomous driving is gradually becoming mainstream in smart transportation and modern logistics [1]. The detection of road surface guidelines is a prerequisite for achieving autonomous driving. According to different detection methods, there are traditional image processing methods and machine learning methods [2-3]. Among them, traditional image processing methods utilise techniques like edge detection, colour segmentation and Hough transform to recognise straight lines, which are suitable for simple environments. Machine learning can automatically learn and recognise guidelines, adapt to complex lighting and weather conditions, and have higher accuracy [4-5]. However, with the increasing complexity of application scenarios, traditional road surface guide detection algorithms are no longer able to meet practical needs. For example, the template matching method relies on the quality and quantity of templates. When the templates are incomplete or of poor quality, the recognition effect is poor [6]. The method based on the Hough transform requires high pattern integrity, and the detection performance decreases significantly when encountering incompleteness or occlusion [7]. Traditional machine vision algorithms are sensitive to geometric transformations, and their detection performance is unstable when the target undergoes rotation, scaling and other situations [8]. These issues seriously affect the efficiency and accuracy of road guideline detection in autonomous driving. To summarise, traditional methods rely on manually designed features, which have harsh requirements on template quality, pattern integrity and environmental stability, and are difficult to adapt to complex lighting, occlusion or small target scenes. Moreover, the existing methods generally suffer from insufficient small-target

feature extraction and limited real-time performance, which leads to the detection accuracy and efficiency being difficult to meet the actual needs of autonomous driving.

Starting from the effective extraction and processing optimisation of road guideline features, this study focuses on the basic logic and excellent performance of you only look once version 5 (YOLOv5), and combines it with convolutional block attention module (CBAM) to ultimately construct an autonomous driving road guideline detection model based on enhanced-YOLOv5 (E-YOLOv5).The research aims to optimise the efficiency and accuracy of the road surface guideline detection model, enhance its resistance to interference in complex environments, and make positive contributions to the wider application of autonomous driving technology, laying the foundation for the rapid development of computer vision computing. The innovations of the study are: (1) targeted optimisation of the integration method of CBAM, through the channel and spatial attention synergistic enhancement of small target feature expression, and to reduce noise interference; (2), the new adaptive anchor frame parameter adjustment mechanism, combined with the feature map on the sampling and fusion strategy, to improve the positioning accuracy of fine markers.

The research is structured into four sections. The first section introduces the current research on the logic and algorithms of road surface guideline detection worldwide. The second section starts from modules such as YOLOv5 and CBAM to establish a precise and efficient E-YOLOv5 autonomous driving road guideline detection model. The third section provides numerical examples and practical application analysis of the proposed detection model to verify its reliability. The final section provides a comprehensive summary and analysis of the article.

2. RELATED WORKS

Driven by computer vision technology in various industries, the application of road surface guideline detection in fields such as autonomous driving is rapidly increasing [9]. Road surface guideline detection is a key technology for achieving autonomous driving and an important application direction that needs to be continuously expanded and deepened in computer vision technology [10]. However, due to the complex and ever-changing traffic conditions, the detection performance of road surface guidelines is not stable, so many researchers have optimised this problem. Z. Qin et al. proposed a globally feature-based anchor-driven ordinal classification algorithm to address the efficiency and accuracy issues of road surface guideline detection in complex scenarios. The algorithm represented lanes using sparse coordinates and modelled them on mixed row and column anchors, transforming lane detection tasks into ordinal classification problems. This significantly reduced computational costs and improved detection performance and real-time performance [11]. D. K. Dewangan et al. built a two-layer deep learning framework suitable for multiple different weather conditions to address the accuracy issue of road surface guide detection under complex weather conditions. By extracting local vector pattern texture features and using optimised deep convolutional neural networks for road and lane classification, the computation time for detection and recognition was shortened, and the detection efficiency and robustness under harsh weather conditions were improved [12]. Y. Dong et al. proposed a novel spatiotemporal sequence for a single-frame deep learning architecture to address the accuracy issue of road surface guide detection in complex driving scenarios. Through spatial convolutional neural networks, spatiotemporal recurrent neural networks and encoder-decoder structures, single-frame features were extracted and spatiotemporal features were fused to improve detection performance in challenging driving scenarios [13]. In response to the time-consuming and laborious problem of road boundary annotation in high-precision maps of urban scale, Z. Xu et al. proposed an urban boundary detection system, which combined deep learning and graph inference techniques to stitch the boundary maps of all image blocks to generate a large-scale urban road boundary network, significantly improving the accuracy and efficiency of road boundary detection [14].

In addition, D. H. Lee et al. built a lightweight depthwise separable convolutional U-shaped model for road guideline detection and path prediction. By introducing depthwise separable convolution and path prediction algorithms, the prediction curvature and lateral deviation errors were reduced, and the efficiency and accuracy of lane detection and path prediction were improved [15]. In response to the problem that existing lane detection algorithms are difficult to robustly extract lane instances under different numbers of lanes, Y. Sun et al. proposed an adaptive multi-lane detection method on the basis of instance segmentation. Through instance segmentation, as well as the introduction of cosine metrics into the objective function, the discriminative ability of foreground features was enhanced, and the robustness of the algorithm was improved [16]. In response to the challenge of accurately understanding lane positions and types in autonomous driving, L. Zhang et al. built

a cascaded deep neural network model that compensated for the lack of low dimensional encoded data and edge information by fusing spatial features extracted by encoders and decoders, improving the accuracy of real-time detection and classification of road guidelines [17]. H. Gao et al. combined channel attention and multidimensional regression loss to address the 3D detection of autonomous vehicles in front-view images. This method combines the internal parameters of the camera and depth estimation to reduce image depth calculation errors, improve detection accuracy and enhance driving safety [18]. In response to the human and material resource consumption in road marking detection and evaluation, C. Lin proposed a detection method based on a low-channel LiDAR sensor. By optimising the built-in characteristics of the LiDAR, an elite-preserving genetic algorithm was applied. This method could obtain the optimal coverage range and point cloud density of road markings [19].

In summary, numerous researchers worldwide have noted the problems that exist in the operation of road surface guide detection models and have conducted multiple research works to address these issues. Accurate and efficient detection of road surface guidelines is a prerequisite for expanding computer vision technology in fields such as smart transportation and autonomous driving, and its importance is self-evident. However, the above studies have rarely optimised the performance of small object detection. YOLOv5 can achieve fast and accurate object detection, suitable for real-time road surface guide recognition [20]. CBAM can enhance key features and improve the robustness and detection accuracy in complex scenes [21]. Therefore, based on YOLOv5, combined with the CBAM module, and optimised model parameters, a road guideline detection model for autonomous driving on the basis of E-YOLOv5 is established to optimise the efficiency of lane line feature processing and achieve real-time and efficient detection. The research aims to provide comprehensive and innovative solutions for the detection accuracy and efficiency of complex road surface guidelines.

3. METHODS AND MATERIALS

This section involves two subsections. The first section provides a detailed explanation of YOLOv5 and its internal modules. The second section combines YOLOv5 with CBAM and optimises the anchor box-related parameters. The E-YOLOv5 is built to optimise the feature processing performance of the road guideline detection model.

3.1 Road surface guideline detection method based on YOLOv5

The feature extraction and processing of road surface guideline images are key operations for capturing their details and structures, and their efficiency determines the accuracy and stability of road surface guideline detection [22]. However, traditional road surface guide detection models are often affected by multiple factors, resulting in low efficiency in extracting and utilising feature information from road surface guide images, leading to low detection efficiency and high efficiency [23]. YOLOv5 can efficiently and in real-time detect multiple objects, with high accuracy and fast inference capabilities, making it easy to deploy and apply [24]. Therefore, YOLOv5 is used as the basic skeleton of the road surface guideline detection model, and its structure is shown in *Figure 1*.

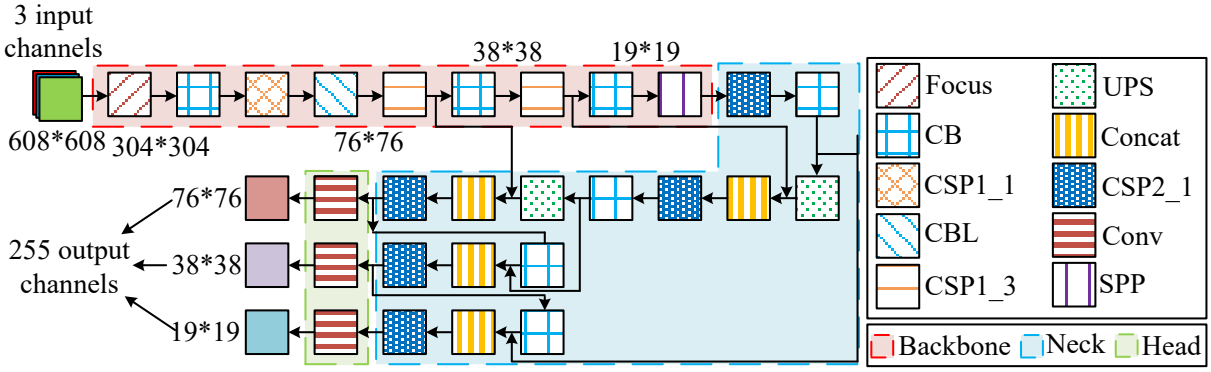


Figure 1 – Structure of YOLOv5

In *Figure 1*, YOLOv5 has three parts: backbone, neck and head [25]. The backbone network extracts rich feature information from input images to ensure efficient feature extraction and computational performance. The neck network, also known as the feature enhancement module, is applied to fuse multi-scale features,

enhance small object detection capabilities, and transmit richer and more hierarchical features to the head. The head performs final bounding box prediction and category classification based on these features, generating multiple bounding boxes and their confidence and category probabilities for each feature map position. These network structures consist of operations such as **upsampling (USP),convolution (Conv), concatenation (Concat) and focus,** as well as **conv-batch normalisation (CB), CB-Leaky ReLU (CBL), cross-stage partial network (CSP) and spatial pyramid pooling (SPP)**. Among them, the CB module consists of convolutional layers and batch normalisation layers, while the CBL module adds an activation function layer on the basis of the CB module. Focus enhances the ability to capture detailed information by rearranging the input imagepixels, as displayed in *Figure 2*.

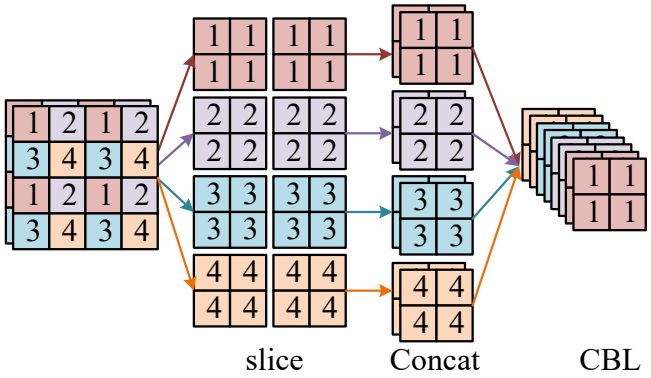


Figure 2 – The process of the focus operation

In *Figure 2*, the focus operation achieves a balance between spatial downsampling and channel expansion by dividing the input image into 2*2 localised regions via space-to-depth and stitching the pixels within each region along the channel dimension. Assuming that the image input size is h ∗ w and the number of channels is c, the height and width of the output image of the focus operation are halved, and the number of channels is expanded to 4 times the original. In YOLOv5, its input image size is 608*608, the number of channels is 3, and the 3-channel pixel values of each 2*2 neighbourhood are spliced along the channels to generate a 3*4=12-channel feature map, and the output size is 304*304 (i.e. 0.5*608*0.5*608). The focus operation balances computational efficiency with feature expressiveness by reducing the spatial resolution of the input image and extending the channel dimension [26].

The CSP module splits input features into two pathways: one remains unchanged to preserve raw information, while the other undergoes partial processing via residual blocks (CSP1_x) or CBL layers (CSP2_x). This split-merge design enhances gradient propagation, mitigates information loss and reduces redundant computations. In the backbone, CSP1_x leverages stacked residuals for deep feature extraction; in the neck, CSP2_x employs lightweight CBL modules (Conv+BN+LeakyReLU) to optimise multi-scale fusion and real-time inference efficiency. And the structures of CSP1_x and CSP2_x are shown in *Figure 3*.

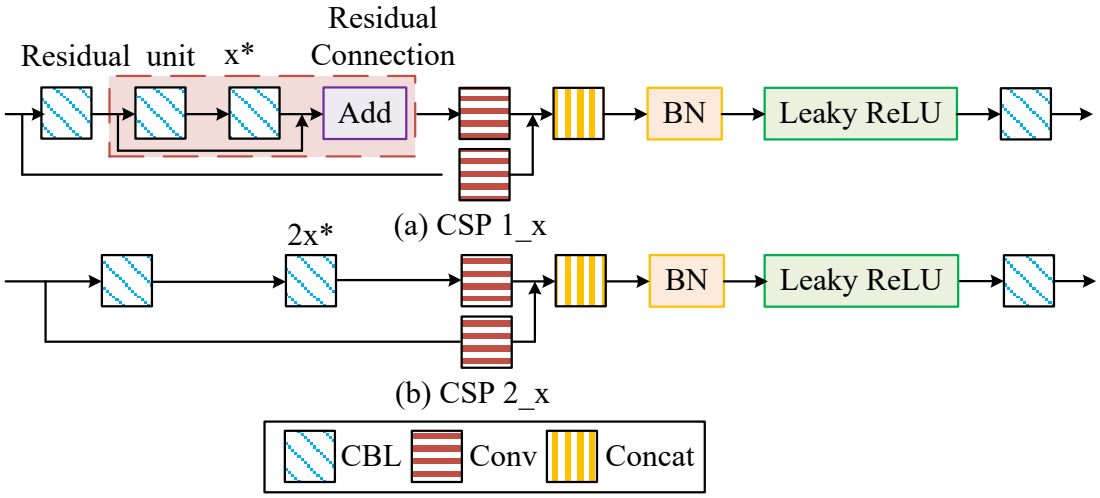


Figure 3 – The structure of CSP 1_x and CSP 2_x

In *Figure 3a*, the CSP 1_x module adopts a relatively simple layered design aimed at reducing computational complexity while maintaining feature richness [27]. The design explicitly incorporates BN and LeakyReLU operations to stabilise feature distributions and alleviate gradient vanishing, as defined by the LeakyReLU function. The input feature map has two branches. The main branch first passes through CBL, where BN normalises channel-wise activations using mini-batch statistics, followed by LeakyReLU for non-linear transformation. It then goes through x residual units, each containing a shortcut connection to fuse the original and processed features. The auxiliary branch directly convolves the original features with a 1×1 kernel, followed by BN to maintain consistent feature scaling. Two branches are concatenated in the channel dimension, activated by the BN function to recalibrate fused feature magnitudes, and finally integrated into a CBL output. As shown in *Figure 3b*, the structures of CSP2_x and CSP 1_x are the same, with the difference being that CSP 1_x replaces the residual units in CSP 1_x with 2x CBLs, where each CBL includes sequential Conv-BN-LeakyReLU operations to prioritise lightweight multi-scale fusion. The former is applied in the neck network [28].

In addition, the SPP converts input feature maps of any size into fixed-size output vectors through multi-scale max pooling operations, enhancing the model's detection ability [29]. The SPP in YOLOv5 is displayed in *Figure 4*.

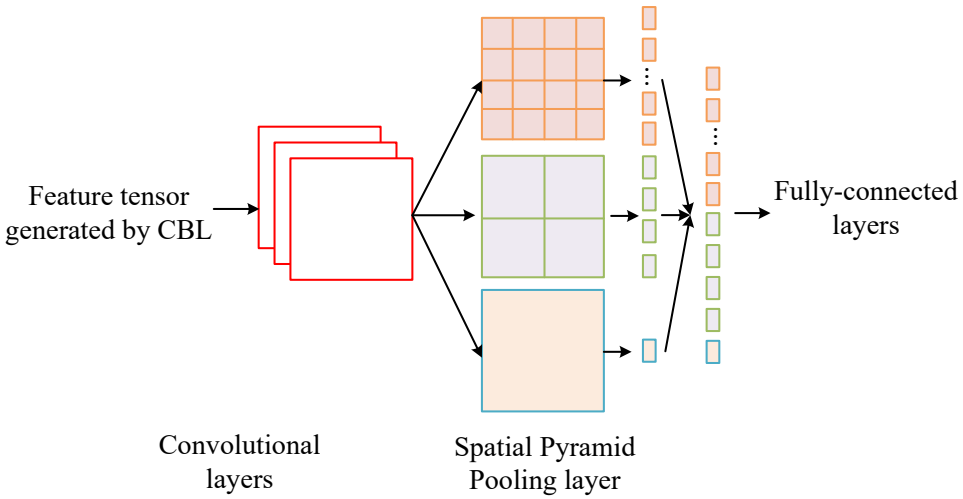


Figure 4 – The structure of the SPP module

In *Figure 4*, SPP separates the input feature map into multiple sub-regions of different sizes and performs max pooling on each sub-region to extract the most representative feature values. These pooling results are ultimately concatenated together to form a fixed feature vector. The SPP module fuses features through parallel 5×5, 9×9 and 13×13 multi-scale maximal pooling, and the pyramid structure synchronises the capture of global context and local details to enhance scale invariance and keep the output dimension fixed. Therefore, SPP can process images of any input size and enhance the detection accuracy for large, medium and small targets by fusing multi-scale information. In addition, YOLOv5 considers the aspect ratio influence factor V to address issues such as the relative position relationship between the predicted and the target bounding boxes, as well as differences in aspect ratios [30]. Its calculation is shown in *Equation (1)*.

$$V = \frac{4}{\pi^2}\left(arctan\left(\frac{A}{B}\right) - arctan\left(\frac{C}{D}\right)\right)^2 \tag{1}$$

In *Equation (1)*, A and B signify the width and height of the ground truth box. C and D signify the width and height of the predicted bounding box. Therefore, YOLOv5 uses the complete intersection over union loss (CIoU_Loss) as the bounding box regression loss function for the head [31], as shown in *Equation (2)*.

$$CIoU\_Loss = 1 - U + \frac{\rho^2}{\rho^2(d)} + V \cdot \alpha \tag{2}$$

In *Equation (2)*, Urepresents the IoU, which is applied to measure the overlap between the predicted and the ground truth boxes. $\rho^2(P)$ *s*ignifies the squared Euclidean distance between the centroids of these two boxes. $\rho^2(d)$ represents the square of the diagonal length of the smallest bounding rectangle containing two boxes. V is the aspect ratio influencing factor applied to measure the consistency of the aspect ratio between these two. α is the equilibrium factor. According to the dynamic adjustment of the relationship between U and V, it can be represented as shown in *Equation (3)* [32].

$$\alpha = \frac{V}{1 - U + V}. \tag{3}$$

CIoU_Loss provides more accurate bounding box regression by combining IoU, centre point distance and aspect ratio consistency, improving the model performance in handling targets of various scales and shapes.

3.2 Autonomous driving road surface guideline detection model based on E-YOLOv5

YOLOv5 can provide efficient and accurate road surface guideline detection results. However, YOLOv5 exhibits notable limitations in detecting fragmented or occluded guidelines under complex road conditions, frequently leading to missed detections and localisation inaccuracies. As a lightweight attention module, CBAM can optimise the model's attention to important features, reduce the influence of environmental noise, and enhance the temporal nature of feature information [33]. Therefore, the study combines CBAM to improve YOLOv5. This study integrates CBAM into YOLOv5 at three critical positions: 1) after the 4th CSP1_3 block in the backbone to amplify high semantics, 2) before upsampling layers in the neck to prioritise cross-scale channel-spatial correlations, and 3) prior to the final CSP2_3 block in the head to refine localisation-sensitive features. An autonomous driving road surface guide detection model on the basis of E-YOLOv5 is built. Among them, the CBAM is displayed in *Figure 5*.

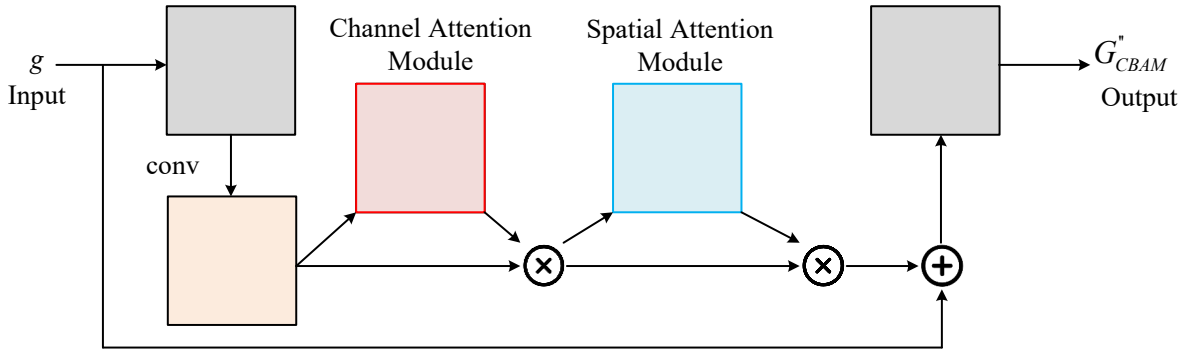


Figure 5– The structure of CBAM

In *Figure 5*, CBAM has a channel attention module and a spatial attention module. The input feature map g involves two branches. The main branch performs a simple convolution operation on the image to obtain the feature map G. Then, the channel attention module performs convolution processing to obtain $M_z(G)$ $(1 * 1 * c)$, M_z is the channel weight vector output by the channel attention module instead of the direct feature map. $M_z(G)$is multiplied (element-wise multiplication) by G to obtain the first attention weighted feature map $G' = M_z(G) \otimes G$. *Next,* G' is taken as the input of the spatial attention module for convolution processing to obtain$M_y(G')$ $(w * h * 1)$, M_y is the spatial weight matrix output by the spatial attention module and $M_z(G)$ is multiplied by G to output $G'' = M_y(G') \otimes G'$. Finally, G'' is added to the original feature map g of another branch to obtain the final output G''_{CBAM}.

CBAM can be integrated into existing basic network architectures without the need for significant modifications to the basic network structure [34]. The spatial attention and channel attention are shown in *Figure 6.*

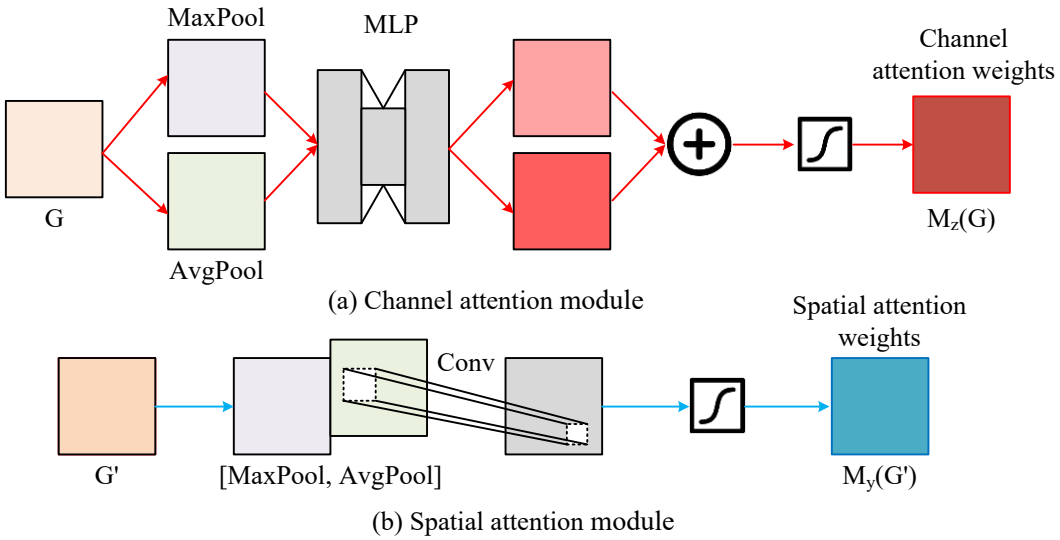


Figure 6 – The structure of the channel attention module and the spatial attention module

In *Figure 6a*, the channel attention module processes the input feature map G through two parallel max pooling layers and average pooling layers, changing the feature map dimension from $c * h * w$ to$c * 1 * 1$. Subsequently, the feature map is channel compressed and extended, and two activated features are obtained using the ReLU activation function, where the channel compression operation is implemented by sharing the fully connected layer of the multilayer perceptron (MLP) [35]. After adding the two output features element by element, the channel attention weight $M_z(G)$ is generated through the Sigmoid activation function, as shown in *Equation (4)*.

$$M_z(G) = \sigma(MLP_{avg}\big(AvgPool(G)\big) + MLP_{max}(MaxPool(G))) \tag{4}$$

In *Equation (4)*, σ represents the Sigmoid. In *Figure 6b*, the spatial attention module receives G' as input and first obtains two feature maps of $1 * h * w$through max pooling and average pooling [36]. The two feature maps are merged through concatenation, and a single channel feature map is obtained by convolving with a kernel size of $7 * 7$. After passing through the Sigmoid, the spatial attention weight $M_y(G')$ is generated, as shown in *Equation (5)*.

$$M_y(G') = \sigma(f_{7*7}([AvgPool(G'); MaxPool(G')])) \tag{5}$$

In *Equation (5)*, σ signifies the Sigmoid activation function. $[AvgPool(G'); MaxPool(G')]$ signifies concatenating the two features obtained from max pooling and average pooling. f_{7*7} is convolution op with$7 * 7$ kernel. In addition, the anchor box parameters (anchor box parameters) define the predefined size and scale of the candidate box in the target detection model, which is used to generate the initial prediction box for targets of different scales. Appropriate anchor box parameters can significantly improve the recognition performance for small-sized markings, ensuring that the model can accurately capture small features even in complex road conditions. Regarding the anchor-free approach, although it avoids the bias of the preset anchor frames, the real-time requirements of YOLOv5 and the multi-scale prediction structure are more suitable for the anchor frame-based recognition approach. Therefore, the study adjusts the anchor box parameters of the model. The study analyses the target size distribution based on K-means clustering (threshold set IoU = 0.6) and adds a fourth set of small-size anchor frames (3.2*3.2, 4.8*4.8, 6.4*6.4) for the three sets of anchor frames of the original YOLOv5 (min. 10*13), whose parameters are determined by the cluster centres of mass of the small target subsets. To ensure the effectiveness of the new anchor box, the model head structure is adjusted. The original size of the last feature map layer is 80*80, and based on the data distribution of the small target share, a bilinear interpolation is used to upsample to 160*160 to align it with the size of the second feature map layer of the backbone network (160*160), and to fuse the multiscale details through channel splicing. This size choice balances small target sensitivity (high resolution) with computational overhead. It was then concatenated and fused with the second layer feature map in the backbone network to generate a larger and

more detailed feature map specifically for small object detection. Finally, by running the "Text to YOLO" script, a new dataset path folder is generated to ensure that all adjusted parameters and configurations are correctly applied to the training process. The "Text to YOLO" script auto-generates configuration files (dataset paths, anchor parameters) that align with structural modifications (e.g. CBAM, 160×160 feature fusion) and training hyperparameters (learning rates, loss weights), ensuring consistent deployment of new anchors (3.2×3.2, etc.) and avoiding manual errors. In summary, the training and detection flow of the proposed E-YOLOv5-based autopilot pavement guidance detection model is shown in *Figure 7*.

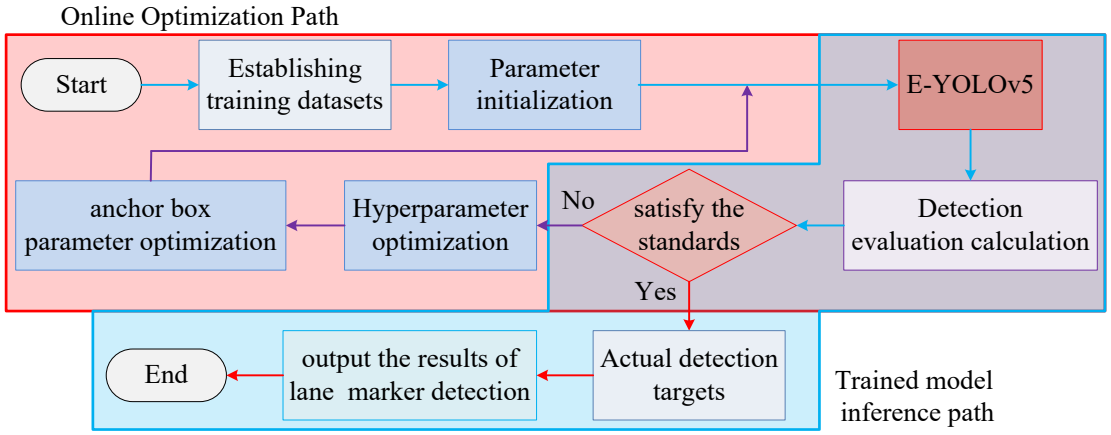


Figure 7 – Dual-phase optimisation of E-YOLOv5

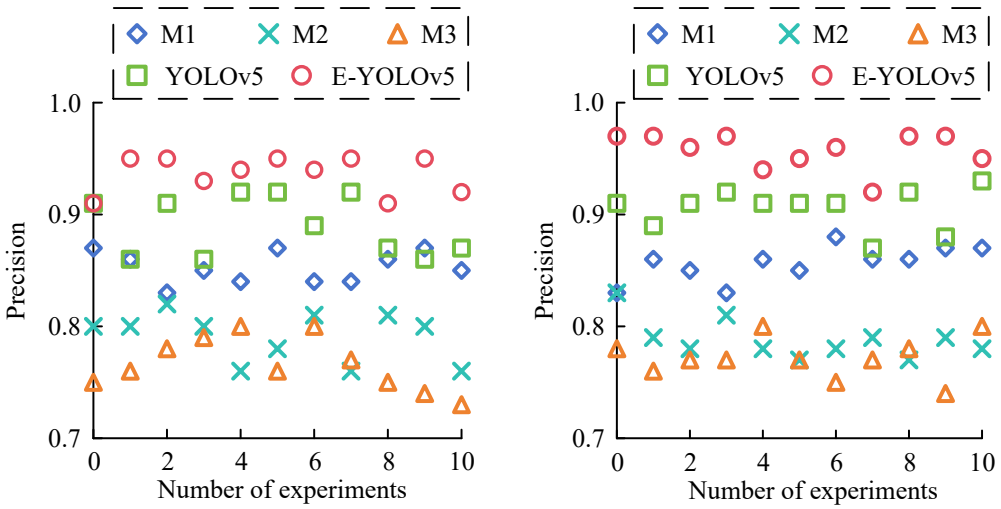
In *Figure 7*, the model first constructs a training set of road surface guidelines and then sets the initial parameters of the model. Afterwards, the E-YOLOv5 is trained using the training set, and the detection target evaluation value of the training output is calculated. Next, whether the evaluation value meets the preset criteria is determined. If not met, the hyperparameters and anchor box parameters of the model are optimised and adjusted, and training continues. If the evaluation value meets the preset criteria, the actual road surface guideline detection target is output for actual detection. Finally, the road surface guideline detection result is output. The whole process ensures the high accuracy and reliability of the model under complex road conditions, provides key visual inputs for the autonomous driving system, and effectively improves the safety and efficiency of road driving.

4. RESULTS

To verify the effectiveness and superiority of the E-YOLOv5, simulation experiments and actual model performance experiments are conducted based on the theoretical foundation and algorithm analysis mentioned above. The results are analysed in detail, and the performance of abnormal behaviour detection accuracy and real-time performance is compared.

4.1 Simulation operation experiment

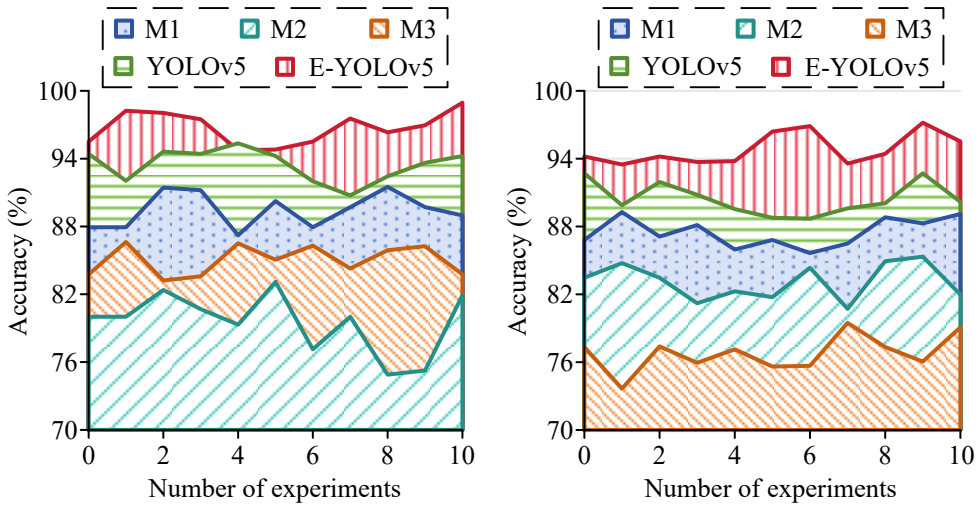
In the simulation experiment, Windows 10 is chosen as the operating system and the Silicon Valley Innovation Centre's autonomous driving simulator software [37] is used as the platform to simulate the autonomous driving road environment. The study compares the single-stage object detection algorithm, feature pyramid network, U-shaped detection network and traditional YOLOv5, and names them M1, M2, M3 and YOLOv5, respectively. The proposed model is named E-YOLOv5. The study trains all algorithms on the Cityscapes dataset and uses a cosine annealing scheduler to set the initial learning rate to 0.01. The Cityscapes dataset includes 5,000 high-res images (2048*1024) from 50 urban scenes, featuring complex lighting, occlusions and multi-scale objects (9.3% small markings). Its pixel-semantic/instance annotations enable robust validation of autonomous driving models in real-world scenarios. After 300 iterations, the learning rate gradually decreased to 10% of the initial learning rate according to the cosine decay curve. Cosine annealing is a learning rate decay mechanism designed to optimise training stability and model accuracy. After training, the study first verified the model's detection precision in detecting left arrows (LA) and straight arrows (SA) in the simulated environment, as displayed in *Figure 8*.



(a) Detection precision for LA (b) Detection precision for SA

Figure 8 – The detection precision difference for LA and SA detections

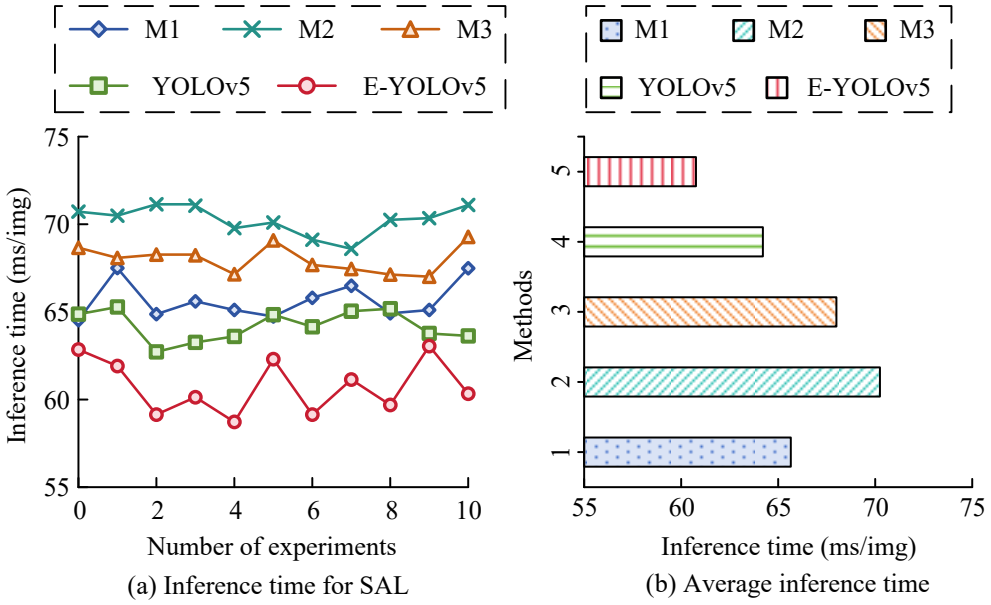
According to *Figure 8a*, when detecting LA, the detection precision of M3 was the lowest, between 0.73-0.80. Next was M2, with a precision of 0.76-0.82. The average precision of M1 and YOLOv5 was 0.85 and 0.89, respectively. The precision range of E-YOLOv5 was 0.91-0.95. As shown in *Figure 8b*, E-YOLOv5 had the highest precision with an average precision of 0.96 when detecting SA. Next was YOLOv5, with an average precision of 0.91. The precision of M1 and M2 was between 0.83-0.88 and 0.77-0.83, respectively. The precision of M3 was the lowest, with an average precision of only 0.77. E-YOLOv5 has significantly higher precision in detecting LA and SA compared to other algorithms. Subsequently, the research verified the accuracy in detecting right arrows (RA) and straight and left (SAL) arrows. The experimental results are shown in *Figure 9*.



(a) Detection accuracy for RA (b) Detection accuracy for SAL

Figure 9 – The detection accuracy discrepancy for RA and SAL

According to *Figure 9a*, E-YOLOv5 had the highest accuracy in identifying and detecting RA, ranging from 94.74% to 98.95%. Next was YOLOv5, with an accuracy range of 90.75% to 95.36%. The average accuracy of M1 and M3 was 89.43% and 85.02%, respectively. M2 had the lowest accuracy. The range was between 74.89% and 83.10%. As shown in *Figure 9b*, M3 had the lowest accuracy in identifying and detecting SAL, with an average accuracy of 76.79%.The second was M2, with an average accuracy of 83.11%. The accuracy of M1 and YOLOv5 ranged from 85.65% to 89.28% and 88.70% to 92.71%, respectively. The average accuracy of E-YOLOv5 was 94.85. E-YOLOv5 has better accuracy than other algorithms in recognising and detecting RA, SAL arrows. Afterwards, the research verified the detection inference time of the model for straight and right (SAR) arrows, as displayed in *Figure 10*.



(a) Inference time for SAL (b) Average inference time

Figure 10 – Inference time discrepancy for detecting SAL

As shown in *Figure 10a*, M2 had the longest inference time for SAR detection and recognition, ranging from 68.6 ms/img to 71.1 ms/img. Next was M3, whose inference time ranged from 67.0 ms/img to 69.3 ms/img. The inference time of M1 and YOLOv5 was between 64.5 ms/img to 67.5 ms/img and 62.7 ms/img to 65.3 ms/img, respectively. The inference time of E-YOLOv5 was between 58.7 ms/img and 63.1 ms/img. According to *Figure 10b*, E-YOLOv5 had the shortest average inference time, reaching 60.8 ms/img. Next was YOLOv5, with an average inference time of 64.2 ms/img. The average recommendation speed of M1 and M3 was slightly longer, at 68.7 ms/img and 68.0 ms/img, respectively. M2 had the longest inference time, with an average speed of only 70.2 ms/img. The detection speed of E-YOLOv5 is much better than traditional detection methods when detecting SAR turn arrows. On this basis, ablation experiments are performed to verify the impact of the network modules and anchor box optimisation used in the study on the overall detection performance. The study sequentially extracted modules such as focus, CSP, SPP, CBAM and anchor box optimisation measures from the E-YOLOv5 model and replaced them with ordinary convolutions, as displayed in *Table 1*.

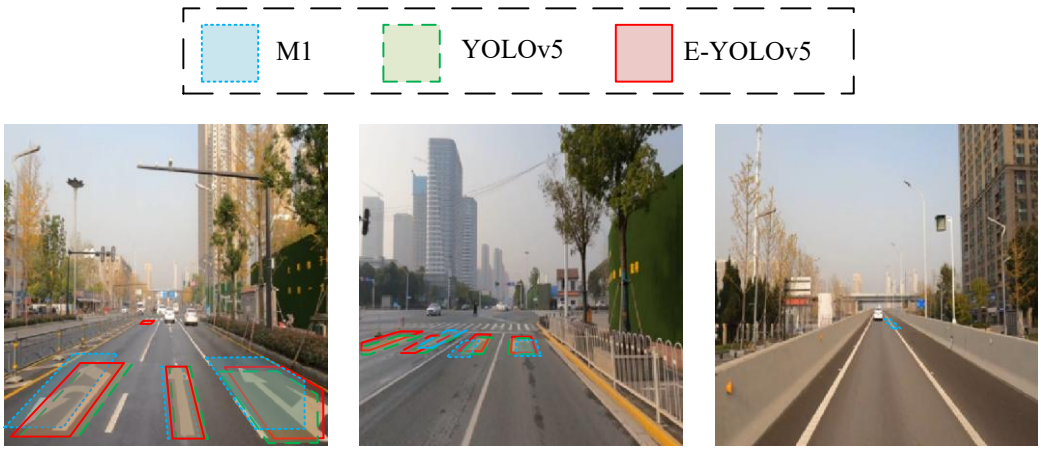
Table 1 – Module ablation experiment

	Focus	**CSP**	**SPP**	**CBAM**	**Anchor box optimisation**	**Precision**	**Accuracy(%)**	**Inferencetime(ms/img)**
M1	×	×	×	×	×	0.85	87.31	65.3
M2	×	×	×	×	×	0.81	82.92	69.9
M3	×	×	×	×	×	0.76	82.10	69.0
YOLOv5	√	√	√	×	×	0.86	89.93	64.7
E-YOLOv5	√	√	√	√	√	0.95	95.23	63.3
E-YOLOv5	×	√	√	√	√	0.90	94.23	64.9
E-YOLOv5	√	×	√	√	√	0.91	92.79	66.3
E-YOLOv5	√	√	×	√	√	0.91	93.81	65.0
E-YOLOv5	√	√	√	×	√	0.87	91.91	67.0
E-YOLOv5	√	√	√	√	×	0.85	89.88	66.6

According to *Table 1*, in the ablation experiment, the detection precision of E-YOLOv5 was 0.95, the accuracy was 95.23%, and the inference time was 63.3ms/img. When the model lost the focus operation, its precision dropped to 0.90. When the model lost the CSP module, its accuracy dropped to 92.79%. When the model lost the SPP module, its inference time dropped to 65.0m/s/img. In addition, when the model lost CBAM, its precision and accuracy were decreased by 0.08% and 3.32%, respectively. When the model did not optimise the anchor box parameters, its inference time increased by 3.3ms/img. At this time, the three indicators of other algorithms were between 0.76–0.86, 89.93%–82.10% and 64.7–69.9, respectively. The optimisation and improvement measures for YOLOv5 have improved the detection performance for road surface guidelines.

4.2 Actual model performance experiment

The operation status of the road surface guideline detection model in simulation is an important criterion for measuring the performance of the model. However, due to the influence of uncontrollable factors on the audience, the road surface guideline detection model often differs from simulation in practical applications. Therefore, the study chose a closed road section as the actual experimental environment. Based on the results of simulation experiments, M1 and YOLOv5 were taken as comparison algorithms, and E-YOLOv5 was taken as the research object. The potential and promotion value of the model in practical applications was verified through the detection performance of road guidelines under actual conditions. The study first conducted actual detection experiments during the process of manual safe driving, as displayed in *Figure 11*.



(a) Close-up (b) Longer shot (c) None

Figure 11 – Autonomous detection in autonomous driving

In *Figure 11a*, the vehicle was located in the centre of the three lanes, and M1, YOLOv5 and E-YOLOv5 could all detect the guidelines on the road. However, compared to the actual lead position, YOLOv5 had a slight deviation, while M1 was more significantly offset. The detection box of E-YOLOv5 had the highest overlap rate with the actual guideline position, and only E-YOLOv5 detected the guideline arrow in the far left front. As shown in *Figure 11b*, the vehicle was currently located in the rightmost right turn lane of the four lanes. YOLOv5 and E-YOLOv5 could detect all directional arrows in four lanes, but YOLOv5 still exhibited a deviation phenomenon. The detection box of M1 was severely in deviation and could not detect the guide arrow of the leftmost lane. As shown in *Figure 11c*, the vehicle was currently in a single lane with no directional arrows on the road surface. However, in this section, M1 was detected incorrectly, while YOLOv5 and E-YOLOv5 showed no abnormalities. In actual detection, the E-YOLOv5 has the highest performance and strong anti-interference ability across different environments. To further validate the actual detection performance, experiments are conducted on the actual detection precision on road sections with a wide field of view and appropriate lighting, as displayed in *Figure 12*.

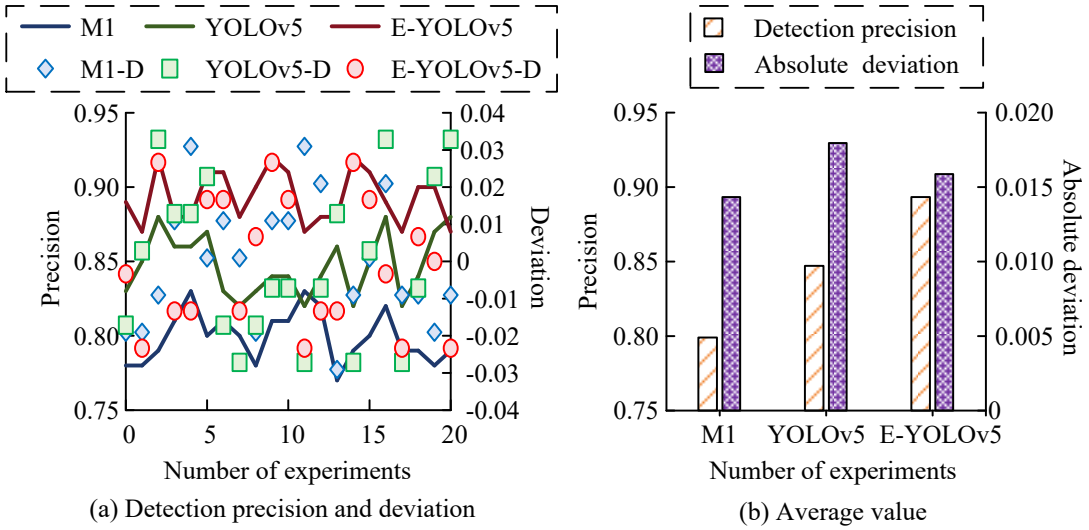


Figure 12 – Actual model detection precision and deviation

According to *Figure 12a*, the actual detection precision of M1 was the lowest, between 0.77–0.83, with a deviation of -0.029 and 0.031. The precision of YOLOv5 ranged from 0.82 to 0.88, with a deviation of -0.027 to 0.033. The confidence of E-YOLOv5 was higher than the first two, with a range of 0.87–0.92 and a deviation of -0.023 to 0.028. According to *Figure 12b*, the average precision of M1, YOLOv5 and E-YOLOv5 was 0.80, 0.85 and 0.89, respectively. The average precision absolute deviation of M2 was the largest, at 0.018. Next was E-YOLOv5, with an average deviation of 0.016. The average deviation of M1 was the smallest, only 0.014. In actual detection, E-YOLOv5 has the highest precision and is relatively stable. On this basis, experiments are conducted on the accuracy of model detection, and the results are shown in *Figure 13.*

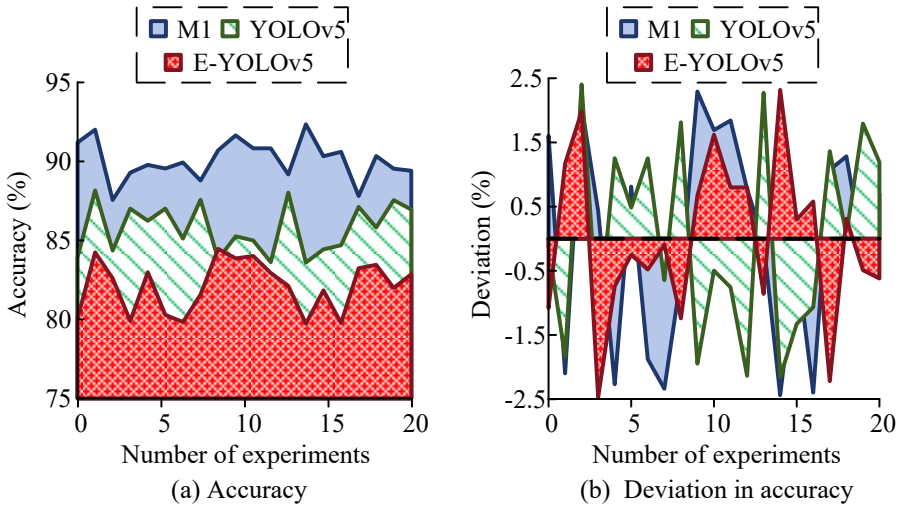


Figure 13 – Actual model accuracy and deviation

According to *Figure 13a*, in actual detection, E-YOLOv5 had the highest accuracy, ranging from 87.56% to 92.35%, with an average accuracy of 90.03%. The accuracy of YOLOv5 was 83.60%–88.16%, with an average accuracy of 85.76%. The accuracy of M1 was relatively low, ranging from 79.73% to 84.46%, with an average accuracy of only 82.17%. According to *Figure 13b*, E-YOLOv5 had the most stable performance in terms of accuracy, with a deviation between -2.47% and 2.32%, and an average absolute deviation of 1.00%.The stability of M1 and YOLOv5 was relatively close, with deviations ranging from -2.44% to 2.29% and -2.16% to 2.40%, respectively. The average deviations of the two were 1.39% and 1.32%, respectively. E-YOLOv5 has high actual detection accuracy, stable performance, and can undertake the task of detecting actual road surface guidelines. Finally, the inference time of the actual detection is verified, as presented in *Table 2*.

Table 2 – Actual model inference time and deviation

Number of experiments	Inference time (ms/img)			Deviation (ms/img)		
	M1	YOLOv5	E-YOLOv5	M1	YOLOv5	E-YOLOv5
2	73.6	73.3	67.2	-0.4	1.4	-1.0
4	74.2	70.9	68.5	0.2	-1.0	0.3
6	74.0	70.7	68.5	0.0	-1.2	0.2
8	73.5	73.4	68.2	-0.5	1.5	-0.1
10	74.2	70.6	68.6	0.2	-1.3	0.3
12	73.9	71.7	67.1	-0.1	-0.2	-1.2
14	73.7	71.9	68.9	-0.3	0.0	0.6
16	75.1	71.4	68.6	1.1	-0.5	0.4
18	73.6	72.8	69.2	-0.3	0.9	0.9
20	74.1	72.2	68.0	0.1	0.3	-0.3
Absolute mean	74.0	71.9	68.3	0.3	0.8	0.5

According to *Table 2*, in actual road surface guideline detection, M1 had the longest inference time, ranging from 73.5 ms/img to 75.1 ms/img, with an average inference time of 74.0 ms/img. The inference time of YOLOv5 was slightly shorter, ranging from 70.6 ms/img to 73.4 ms/img, with an average inference time of 71.9 ms/img. E-YOLOv5 had the shortest inference time, ranging from 67.1 ms/img to 69.2 ms/img, with an average time of only 68.3 ms/img.In addition, M1 had the most stable performance in this aspect, with a deviation between -0.5 ms/img and 1.1 ms/img and an average absolute deviation of 0.3 ms/img. Next was E-YOLOv5, with an inference time deviation between -1.2 ms/img and -0.9 ms/img and an average absolute deviation of 0.5 ms/img. The inference time performance of YOLOv5 was the most unstable, with a deviation between -1.3 ms/img and -1.5 ms/img and an average absolute deviation of 0.8 ms/img. The experimental results indicate that E-YOLOv5 hashigher actual detection efficiency and has the potential for promotion and application. YOLOv5, as the basic skeleton of the model, combined with the CBAM module and optimised anchor box parameters, is effective in improving the efficiency and real-time performance of road surface guideline detection.

5. DISCUSSION

The E-YOLOv5 proposed in this paper demonstrates significant advantages in the road marking detection task. Compared with the traditional single-stage detection algorithm (M1), the average accuracy is improved by 9.7%, and the inference speed is accelerated by 17.3% compared with the feature pyramid network (M2). Compared with the base YOLOv5, the LA/SA detection accuracy is improved by 6.7%/5.5%, and the SAR arrow recognition speed is accelerated by 8.4% through the introduction of the CBAM attention mechanism and the anchor frame optimisation strategy, respectively. In real scenario tests, the detection frame overlap rate of this model is improved by 11.2% compared with the benchmark algorithm, the false detection rate is reduced by 3.8%, and the average accuracy fluctuation under complex lighting is reduced by 42%, which is significantly better than the 15.3% accuracy fluctuation of the existing U-shaped network (M3). These improvements validate the effectiveness of multi-module co-optimisation for autonomous driving environment sensing, especially reaching the industry-leading level in real-time and small target detection.

The research significantly improves detection performance in complex road conditions but still faces limitations in generalisation and deployment efficiency. (1) Insufficient coverage of extreme weather and dense occlusion scenarios in training data restricts model stability in real-world complex environments. (2) Added multi-scale optimisation modules increase computational demands, challenging deployment on low-power automotive hardware. (3) Dynamic parameter adjustment relies on preset criteria, lacking autonomous

adaptability for closed-loop optimisation, limiting continuous learning in unknown environments. To address these issues: (1) enhance scene representation via multi-modal data fusion and cross-sensor information; (2) explore lightweight architecture co-design to reduce complexity using plug-and-play modules and dynamic sparse computation; (3) develop self-driven parameter optimisation frameworks with online learning for autonomous model updates. (4) Optimise CIoU_Loss via dynamic weight allocation to adaptively balance centre-point and bounding box regression; (5) introduce confidence intervals and hypothesis testing to quantify model performance fluctuations and increase data comparability and credibility.

6. CONCLUSION

In response to the low accuracy in traditional road surface guide detection models, YOLOv5 was combined with CBAM, and the anchor box parameters were optimised. Finally, an autonomous driving road surface guide detection model on the basis of E-YOLOv5 was designed, which improved the performance by optimising the extraction and fusion efficiency of small target image feature information. In the simulation experiment, the E-YOLOv5 model proposed by the study had detection precisions of 0.91–0.95 and 0.92–0.97 for LA and SA, respectively, while other algorithms had a precision of 0.73–0.93 for both. The detection accuracy of E-YOLOv5 for RA and SAL was 94.74%–98.95% and 93.49%–97.18%, respectively, while the accuracy of other algorithms for both was between 73.66% and 95.36%. When detecting SAR, the inference time of E-YOLOv5 was between 58.7ms/img and 63.1 ms/img, while the inference time of other algorithms was between 62.7 ms/img and 71.1 ms/img. In the ablation experiment, the precision and accuracy of E-YOLOv5 were 0.95 and 95.23%, respectively. When it lost focus, its precision decreased to 0.90. When it lost CBAM, its accuracy decreased to 93.81%. In addition, in actual performance experiments, the detection box of E-YOLOv5 was most in line with the actual position of the guideline, while other algorithms exhibited phenomena such as deviation, missed detection and false detection. Further research found that the actual precision of E-YOLOv5 was between 0.87 and 0.92, while other algorithms were between 0.77 and 0.88. Moreover, the actual accuracy deviation of E-YOLOv5 ranged from -2.47% to 2.32%, while other algorithms ranged from -2.43% to 2.40%. Meanwhile, the average inference time and deviation of E-YOLOv5 were 68.3mm/img and 0.5mm/img, respectively, while the average inference time and deviation of other algorithms were 72.9m/img and 0.6m/img, respectively. To sum up, the research has practical application value in improving the accuracy and real-time performance of road guideline detection.

REFERENCES

[1] Mao J, et al. 3D object detection for autonomous driving: A comprehensive survey. *International Journal of Computer Vision*. 2023;131(8):1-55. DOI: 10.1007/s11263-023-01790-1.

[2] Huang Y, et al. A survey on trajectory-prediction methods for autonomous driving. *IEEE Transactions on Intelligent Vehicles*. 2022;7(3):652-674. DOI: 10.1109/TIV.2022.3167103.

[3] Ghanem S, et al. Lane detection under artificial colored light in tunnels and on highways: an IoT-based framework for smart city infrastructure. *Complex & Intelligent Systems*. 2023;9(4):3601-3612. DOI: 10.1007/s40747-021-00381-2.

[4] Biparva M, et al. Video action recognition for lane-change classification and prediction of surrounding vehicles. *IEEE Transactions on Intelligent Vehicles.* 2022;7(3):569-578. DOI: 10.1109/TIV.2022.3164507.

[5] Alam A, et al. Distance-based confidence generation and aggregation of classifier for unstructured road detection. *Journal of King Saud University-Computer and Information Sciences*. 2022;34(10):8727-8738. DOI: 10.1016/j.jksuci.2021.09.020.

[6] Yu X, Hanks TD, Geng JJ. Attentional guidance and match decisions rely on different template information during visual search. *Psychological Science*. 2022;33(1):105-120. DOI: 10.3758/s13414-022-02478-3.

[7] Romanengo C, Biasotti S, Falcidieno B. Hough transform for detecting space curves in digital 3D models. *Journal of Mathematical Imaging and Vision*. 2022;64(3):284-297. DOI: 10.1007/s10851-021-01066-8.

[8] Ren Z, et al. State of the art in defect detection based on machine vision. *International Journal of Precision Engineering and Manufacturing-Green Technology*. 2022;9(2):661-691. DOI: 10.1007/s40684-021-00343-6.

[9] Hasanvand M, et al. Machine learning methodology for identifying vehicles using image processing. *Artificial Intelligence and Applications*. 2023;1(3):170-178. DOI: 10.47852/bonviewAIA3202833.

[10] Chalvatzaras A, Pratikakis I, Amanatiadis AA. A survey on map-based localization techniques for autonomous vehicles. *IEEE Transactions on Intelligent Vehicles*. 2022;8(2):1574-1596. DOI: 10.1109/TIV.2022.3192102.

[11] Qin Z, Zhang P, Li X. Ultra fast deep lane detection with hybrid anchor driven ordinal classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2022;46(5):2555-2568. DOI: 10.1109/TPAMI.2022.3182097.

[12] Dewangan DK, Sahu SP. Lane detection in intelligent vehicle system using optimal 2-tier deep convolutional neural network. *Multimedia Tools and Applications*. 2022;82(5):7293-7317. DOI: 10.1007/s11042-022-13425-7.

[13] Dong Y, et al. A hybrid spatial–temporal deep learning architecture for lane detection. *Computer-Aided Civil and Infrastructure Engineering*. 2022;38(1):67-86. DOI: 10.1111/mice.12829.

[14] Xu Z, et al. csBoundary: City-scale road-boundary detection in aerial images for high-definition maps. *IEEE Robotics and Automation Letters*. 2022;7(2):5063-5070. DOI: 10.1109/LRA.2022.3154052.

[15] Lee DH, Liu JL. End-to-end deep learning of lane detection and path prediction for real-time autonomous driving. *Signal, Image and Video Processing*. 2022;17(1):1-55. DOI: 10.1007/s11760-022-02222-2.

[16] Sun Y, et al. Adaptive multi-lane detection based on robust instance segmentation for intelligent vehicles. *IEEE Transactions on Intelligent Vehicles*. 2022;8(1):888-899. DOI: 10.1109/TIV.2022.3158750.

[17] Zhang L, et al. DNet-CNet: A novel cascaded deep network for real-time lane detection and classification. *Journal of Ambient Intelligence and Humanized Computing*. 2022;14(8):10745-10760. DOI: 10.1007/s12652-022-04346-2.

[18] Gao H, et al. CAMRL: A joint method of channel attention and multidimensional regression loss for 3D object detection in automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*.2023;24(8):8831-8845. DOI: 10.1109/TITS.2022.3219474.

[19] Lin C, et al. Mobile LiDAR deployment optimization: Towards application for pavement marking stained and worn detection. *IEEE Sensors Journal*. 2022;22(4):3270-3280. DOI: 10.1109/JSEN.2022.3140312.

[20] Öztürk G, Eldoğan O, Köker R. Computer vision-based lane detection and detection of vehicle, traffic sign, pedestrian using YOLOv5. *Sakarya University Journal of Science*. 2024;28(2):418-430. DOI: 10.16984/saufenbilder.1393307.

[21] Liu Y, et al. Reinforced attention method for real-time traffic line detection. *Journal of Real-Time Image Processing*. 2022;19(5):957-968. DOI: 10.1007/s11554-022-01236-w.

[22] Dewangan DK, Sahu SP. Towards the design of vision-based intelligent vehicle system: methodologies and challenges. *Evolutionary Intelligence*. 2022;16(3):759-800. DOI: 10.1007/s12065-022-00713-2.

[23] Muhammad K, et al. Vision-based semantic segmentation in scene understanding for autonomous driving: Recent achievements, challenges, and outlooks. *IEEE Transactions on Intelligent Transportation Systems*. 2022;23(12):22694-22715. DOI: 10.1109/TITS.2022.3207665.

[24] Wang J, et al. Improved YOLOv5 network for real-time multi-scale traffic sign detection. *Neural Computing and Applications*. 2022;35(10):7853-7865. DOI: 10.1007/s00521-022-08077-5.

[25] Wang Y, et al. Pavement marking incorporated with binary code for accurate localization of autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*. 2022;23(11):22290-22300. DOI: 10.1109/TITS.2022.3173656.

[26] Lu Z, et al. An improved YOLOv5 algorithm for obscured target recognition. *Mathematical Models in Engineering*. 2023;9(1):25-39. DOI: 10.21595/mme.2023.23126.

[27] Ye C, et al. Steering angle prediction YOLOv5-based end-to-end adaptive neural network control for autonomous vehicles. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*. 2022;236(9):1991-2011. DOI: 10.1177/09544070211053677.

[28] Chen Y, et al. Dw-yolo: an efficient object detector for drones and self-driving vehicles. *Arabian Journal for Science and Engineering*.2022;48(2):1427-1436. DOI: 10.1007/s13369-022-06874-7.

[29] He X. Vehicle target detection algorithm based on yolov5. *Frontiers in Computing and Intelligent Systems*. 2023;3(1):56-59. DOI: 10.54097/fcis.v3i1.6024.

[30] Zheng Z, et al. Distance-IoU loss: Faster and better learning for bounding box regression//Proceedings of the AAAI conference on artificial intelligence. 2020;34(07):12993-13000. DOI: 10.1609/aaai.v34i07.6999.

[31] Liu X, et al. Object detection method based on CIoU improved bounding box loss function. Chinese Journal of Liquid Crystals and Displays, 2023;38(5):656-665. DOI: 10.37188/CJLCD.2022-0282.

[32] Tian Y, et al. Recent advances on loss functions in deep learning for computer vision. *Neurocomputing*. 2022;497: 129-158. DOI: 10.1016/j.neucom.2022.04.127.

[33] Zhang L, et al. A real-time lane detection network using two-directional separation attention. *Computer-Aided Civil and Infrastructure Engineering*. 2023;39(1):86-101. DOI: 10.1111/mice.13051.

[34] Xiong L, et al. Detection and threshold-adaptive segmentation of farmland residual plastic film images based on CBAM-DBNet. *International Journal of Agricultural and Biological Engineering*. 2024;17(8):231-238. DOI: 10.25165/j.ijabe.20241705.8069.

[35] Sun X, et al. DMA-Net: DeepLab with multi-scale attention for pavement crack segmentation. *IEEE Transactions on Intelligent Transportation Systems*. 2022;23(10):18392-18403. DOI: 10.1109/TITS.2022.3158670.

[36] Mozaffari S, et al. Early lane change prediction for automated driving systems using multi-task attention-based convolutional neural networks. *IEEE Transactions on Intelligent Transportation Systems*. 2022;7(3):758-770. DOI: 10.1109/TIV.2022.3161785.

[37] Han Y, et al. Simulation and test of vehicle global routing planning algorithm based on autonomous driving simulator. *2022 34th Chinese Control and Decision Conference (CCDC)*. 2022:604-609. DOI: 10.1109/CCDC55256.2022.10033527.