Multi-task Perception Network for Autonomous Driving Based on CSPDarkNet and Attention Mechanism

Yelin Weng
College of Intelligent Engineering Technology, Jiangsu Vocational College of Finance & Economics
Huaian 223001, China
E-mail: wengyelin11@126.com

Keywords: CSPDarkNet, attention mechanism, multi-task perception, autonomous driving, feature decoupling fusion module

Received: March 6, 2025

Aiming at the problems of environmental perception accuracy and multi-task collaborative processing in the process of autonomous driving, this paper proposes an Autonomous Driving Multi-Task (ADMT) perception model based on Cross Stage Partial DarkNet (CSPDarkNet) and attention mechanism. This model shares features and combines information by creating a feature decoupling and fusion module and optimizing the loss function design. This reduces the complexity of training and improves the collaborative effect among tasks. Dynamically weighting the spatiotemporal features of different tasks improves the performance of target detection, instance segmentation, and target tracking. The experimental results showed that, compared with YOLOv4 and YOLOv5, ADMT has achieved significant performance improvements on the KITTI and Cityscapes datasets. Among them, on the KITTI dataset, the F1score of the model reached 0.94, and APiou=0.5 was 0.92. The corresponding values of YOLOv4 and YOLOv5 were 0.93 and 0.91, respectively. These results indicate that ADMT effectively enhances the accuracy and efficiency of target recognition for autonomous driving systems in complex environments, providing strong technical support for future intelligent transportation systems.

Povzetek: Predstavljen je ADMT, model na osnovi CSPDarkNet in mehanizmov pozornosti, ki z razgradnjo in fuzijo značilk doseže bolj kvalitetno zaznavanje, segmentacijo in sledenje kot YOLOv4/YOLOv5.

1 Introduction

With the rapid development of artificial intelligence and machine learning technologies, Autonomous Driving (AD) technology has become one of the hotspots in global transportation research. AD is not only expected to improve traffic efficiency and reduce traffic accidents but also may completely change people's travel methods. Therefore, ensuring the safety and reliability of AD systems becomes particularly crucial. Environmental perception, as the core link of AD, involves vehicles obtaining information about the surrounding environment through various sensors (such as cameras, lidars, ultrasonic sensors, etc.) and conducting real-time analysis and processing [1]. This process aims to quickly identify important information such as surrounding obstacles, traffic signals, and pedestrians to assist vehicles in making safe decisions. However, environmental perception faces many challenges, including dynamically changing traffic scenes, complex climatic conditions, and the interference caused by light changes to visual sensors, etc. [2-3]. These factors pose higher requirements for the accuracy and robustness of sensing systems, especially when multiple sensing tasks must be processed simultaneously. Researchers have proposed the Multi-Task Perception Network (MTPN) to address these challenges. The goal of the MTPN is to implement the simultaneous processing of multiple perception tasks, such as object detection, semantic segmentation, and depth estimation, by constructing a unified network framework [4-5]. MTPN can share the feature extraction process, thereby reducing the consumption of computing resources and improving the collaborative effect among tasks [6-7]. The existing MTPN designs an independent loss function for each segmentation task, which affects the generalization ability of the model and also requires a lot of time to create and adjust the loss function and its parameters. Therefore, it is necessary to shorten the time for processing multiple tasks and optimize the loss function. The Cross-Stage Partial DarkNet (CSPDarkNet) is an improved network structure based on the deep learning DarkNet framework. It effectively enhances the feature extraction ability of the network through channel segmentation and aggregation. The Attention Mechanism (AM) simulates the focusing characteristics of human visual attention, which helps deep learning networks identify key features more accurately. In MTPN, AM helps the network identify and focus on targets in complex environments more accurately, improving perception accuracy [8]. Therefore, the research aims to clearly evaluate the performance of the multi-task perception system in AD, and how to effectively improve the accuracy and efficiency of multiple tasks such as target detection, instance segmentation, and target tracking in complex traffic environments. Meanwhile, the research explores how to

solve the problem of information interference in multitasking and optimize the loss function to improve the overall performance and reliability of the model. Against this background, the study proposes the Autonomous Driving Multi-Task (ADMT) perception model. The innovation point of this model lies in the use of the feature decoupling and fusion module to integrate features from different tasks, thereby reducing the mutual interference among various tasks. A new fusion loss function strategy is proposed. By comprehensively considering the correlations among various tasks, the sharing of the loss function is achieved. In the feature extraction stage, an adaptive attention module is introduced. ADMT can effectively suppress irrelevant background information in complex environments and enhance the recognition ability of small targets and occluded targets.

The research content mainly has four sections. Section 1 discusses the problems of AD and the related research results of the MTPN. Section 2 designs the ADMT. Section 3 analyzes the effectiveness of research methods. Section 4 discusses and summarizes the entire text.

2 Related works

AD technology is extensively utilized in transportation. To improve the effectiveness of AD, researchers have proposed some methods and strategies. Khan M A et al. proposed a building block technology to improve the level of vehicle driving automation. They discussed technologies and concepts such as sensors, mobile edge computing, machine learning, data analysis, and distributed learning, and mapped their roles to end-to-end solutions. The case analysis showed that this technology was feasible and provided ideas for different solutions for 5-level autonomy [9]. Chen L et al. proposed a parallel driving operating system based on parallel driving theory to address incompatibility between different AD algorithms and platforms. The system consisted of 4 structural layers: hardware layer, kernel layer, functional layer, and application layer. It could be derived into specific operating systems in 4 application scenarios: intelligent mining, warehousing, logistics, and ports. In the above testing scenarios, the parallel-driven operating system demonstrated reliability and high efficiency [10]. Li G et al. proposed a continuous decision-making method built on Deep Reinforcement Learning (DRL) to balance driving efficiency and comfort. This method utilized a convolutional neural network to map the relationship between traffic images and vehicle operation, established an end-to-end decision-making framework, and used a deep deterministic policy gradient algorithm to solve the problems in the decision-making process, obtaining the optimal driving strategy. This method provided an effective strategy for AD at intersections while balancing driving comfort, and ensuring driving safety and efficiency [11]. Wang H et al. proposed a novel anchor-free detection network and average boundary model to address occlusion issues in driving scenarios. The backbone network of this network used structural reparameterization technology to locate targets using boundary feature information. This algorithm outperformed CenterNet in both speed and precision, with an accuracy of 55.6%, meeting the requirements for speed and accuracy in driving scenarios [12]. Yang K et al. proposed a robust decision-making framework for AD on highways to improve driving safety and constructed a reinforcement learning strategy based on deep deterministic policy gradients. This strategy directly mapped observations to actions and evaluated model uncertainty based on deep deterministic strategies at runtime to quantify the reliability of the strategy and identify unknown scenarios. The proposed framework had good performance [13].

Some experts have also conducted research on MTPN. Ji Y et al. proposed a multi-task context-aware recommendation method to assist product design in a more intelligent way. By pre-processing work, multi-task knowledge requirement perception, and recommendation engine, the problem of mutual interference of contextual information in different tasks has been solved. This method outperformed traditional methods in terms of effectiveness and performance [14]. Ye H et al. proposed a sequential greedy pruning strategy to optimize the objective of global channel pruning task mismatch. They developed a performance-sensitive criterion to assess the filtersensitivity to each task and retain the filters that are globally most task-related. Experiments on multiple multitasking datasets have shown that the algorithm reduced parameters by over 60% without significant performance degradation, and achieved 1.2-3.3 times acceleration on cloud and mobile platforms [15]. Choudhry A et al. proposed a multi-tasking framework for identifying fake news and rumors on the internet. This framework trained various deep learning models in both single-task and multi-task settings for more comprehensive comparisons. Whether in domain or cross domain settings, this model consistently outperformed single-task models in accuracy, precision, recall, and other aspects [16]. Yang E et al. proposed a task adaptive learning rate method to balance different tasks on each parameter. This method measured the task dominance of a parameter by overall updates made to that parameter by each task, to separate the cumulative gradient in the adaptive learning rate method. Comprehensive experiments on recommendation system datasets have shown that this method optimizes the performance of the dominant task [17]. Pei Y et al. proposed a multi-task DRL method for controlling voltage regulation in distribution systems through photovoltaic intelligent inverters. This method encoded the topology as an additional state for multi-task DRL and utilized a multi-task learning scheme to jointly learn all task control strategies. The comparison conducted on the improved node system showed that this method had good robustness [18].

Xudong Yu et al. proposed an intelligent driving simulation test platform based on a six-degree-of-freedom motion platform for the safety and comfort of AD. This platform included four parts: the motion platform, the ultra-fast data acquisition system, the visual simulation and projection system, and the real-time simulation system. It had A fast response, wide-angle range, and

excellent driving ability, capable of simulating various dynamic driving scenarios. The research results showed that in the absence of monitoring requests, the time it took for drivers to return their palms to the steering wheel was significantly longer than in the situation with a request strategy. Monitoring requests effectively improved the takeover efficiency. Meanwhile, different prompt sounds also had an impact on the driver's takeover performance [19]. Yang T et al. proposed a design method based on the convex program for the Cooperative Adaptive Cruise Control (CACC) problem of Connected and Automated Vehicles (CAVs), aiming to synthesize distributed attack monitors and H∞ CACC controllers. The goal of this method was to minimize the combined impact of covert pseudo-data injection attacks and system interference on fleet dynamics while ensuring that designated trains adhere to performance [20]. Wang X et al. proposed a trajectory prediction model named Multi-Dimensional Spatio-Temporal Feature Fusion (MDSTF) for the problem of trajectory prediction of traffic participants in AD, aiming to accurately capture complex spatiotemporal features. The experimental results on the ApolloScape trajectory dataset showed that this method outperformed other advanced methods in the Weighted Average Displacement Error (WSADE) and Weighted Final Displacement Error (WSFDE) metrics, reducing the errors by 4.37% and 6.23% respectively compared to the best benchmark model S2TNet [21].

In summary, although many scholars have designed some models to enhance the recognition accuracy of AD, the existing models have low recognition accuracy. In view of this, this study attempts to use CSPDarkNet and AM to construct ADMT, providing certain technical support for improving the security of AD and the accuracy of target recognition.

3 Design of ADMT

3.1 Construction of CSPDarknet structure

During the process of AD, vehicles need to detect and recognize surrounding targets. In small target detection in remote sensing images, the YOLOv4 algorithm cannot achieve very accurate detection results [22]. CSPDarkNet is the core Feature Extraction Network (FEN) of YOLOv4. To enhance the extraction of small targets, the paper adds a backbone network to the Context Converter (Cot) module of YOLOv4 and constructs a Cot-CSPDarknet. The Cot module is shown in Figure 1.

In Figure 1, the Cot module is generated by "query", "key", and "value" through three independent 1×1 convolutional layers of the input feature map A. The parameters of the convolutional layers are not shared to learn different context representations. The convolution kernel size of each branch is 1×1, the stride is 1, and the fill is 0, aiming to extract task-related features through channel dimension reduction while maintaining the spatial resolution unchanged. The "key" further encodes the Static Context (SC) through 3×3 convolution kernels to generate the SC, which is designed to capture the local spatial correlation. Subsequently, the attention weight matrix is generated through two 3×3 convolution operations and multiplied by the "values" to obtain the Dynamic Context (DC). The expression for the attention matrix is shown in equation (1).

$$A=[K_1,Q]W\tau W\phi \quad (1)$$

In equation (1), A is the attention matrix. K_1 is an SC. $W\phi$ and $W\tau$ represent convolutions with and without sigmoid activation functions. The DC formula for input is shown in equation (2).

$$K_2 = V \times A \quad (2)$$

In equation (2), K_2 is the DC. The fusion of SC and DC can serve as the Cot module's output. The most basic structure in Cot-CSPDarknet is the Cot module. The convolution kernel in Layer 1 can lower the dimensionality of the Feature Matrix (FM) and decrease the number of parameters. The Layer 2's Cot module combines the residual structure in Cot-CSPDarknet to optimize the extraction of contextual feature information

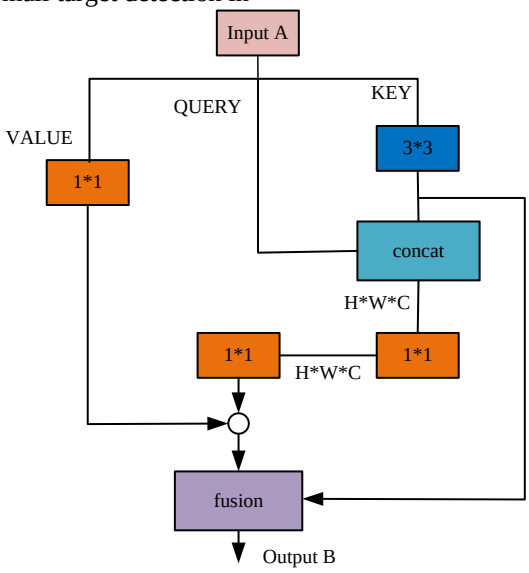


Figure 1: Cot module diagram.

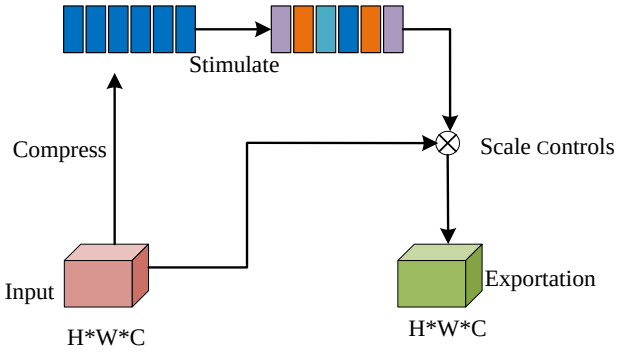


Figure 2: SE module network structure.

near small targets. The function of the 1×1 convolution kernel in Layer 3 is to increase the dimensionality of the FM and double the number of channels in the FM. The number of channels of the Cot module is determined through the combination of empirical dimensionality reduction and experimental tuning. To balance the computational efficiency and the feature expression ability, the number of input feature map channels C is uniformly dimensionally reduced to C/4 through 1×1 convolution (for example, 64 channels are output when C=256). To pay more attention to small objects in remote sensing, this study uses Squeeze-and-Excitation (SE) Attention module in the early stage of FEN downsampling operation. The SE module achieves selective attention to features through two main steps: compression and excitation. In the compression step, the SE module first performs global average pooling on the input feature map to obtain the global feature description of each channel. In the excitation step, the SE module learns the interrelationships between channels through the fully connected layer and generates a channel weight vector. Finally, the SE module multiplies the channel weight vectors obtained through the above learning by the original feature map and re-weights the input features, thereby enhancing the important features and suppressing unnecessary shallow features. This process ensures that the model can focus more on the information useful for the actual tasks when dealing with downstream tasks. The reason for researching the supplementary SE module is that the model can enhance the emphasis on key channels and effectively filter out shallow features irrelevant to the task. The SE module accelerates the convergence of the network by reducing information redundancy and reduces the computational burden during the training process. The addition of the SE module promotes feature sharing among multiple tasks. It enables each task to more effectively apply the learned feature information within the model to different tasks, such as object detection, instance segmentation, object tracking, etc., thereby improving overall performance. Figure 2 shows the SE structure.

In Figure 2, the SE includes compression and excitation operations, which optimize the network's expressive power by explicitly modeling the interdependence between feature map channels. The SE first compresses the features from input U . This operation utilizes the spatial dimension M×N to generate channel features, then performs incentive operations to learn the connections between each channel, and finally multiplies the feature U to generate the output of the SE, as shown in equation (3).

$$\begin{cases} s = F_{ex}(z,V) = \sigma\left(g(z,V)\right) = \sigma\left(V_2 \kappa(V_1, z)\right) \\ Z_c = F_{sq}\left(u_c\right) = (1/(M \times N)) \cdot \sum_{(i=1)}^{M} \sum_{(j=1)}^{N} u_c(i,j) \\ \overline{x}_c = F_{scale}\left(u_c, s_c\right) = u_c \times s_c \end{cases} \tag{3}$$

In equation (3), F_{sq} is the compression operation. F_{ex} is an incentive operation. u_c is a feature from U . κ is the ReLU activation function. F_{scale} is the feature of multiplying the activation values of each channel learned through compression and excitation operations by U . $\overline{x}_c$ is the result of compression and excitation operations on input U . The sub-pixel convolution operation is achieved by inserting a rearrangement step after the convolution layer. In this process, a feature map of a higher dimension is first generated using a convolutional layer, and then the channel information of the feature map is reorganized into a higher-resolution output through a rearrangement operation. The magnification factor adopted is 2, which means that the width and height of the output image are both twice that of the input feature map. The transformation relationship of sub-pixel convolution is shown in equation (4).

$$I^{LR}_{xlr,ylr,c \times r^2} \rightarrow I^{SR}_{x,y,c} \tag{4}$$

In equation (4), I^{LR} and I^{SR} are feature maps with low resolution and high resolution. x and y both represent the size of the feature map. c is the amount of channels in the feature map. By converting low-resolution feature maps into high-resolution one, small targets can be better detected. The fully convolutional structure is adopted, and two convolutional layers are used to change the number of channels and accommodate the feature data of the network model. Two consecutive convolution operations are used to achieve classification and regression of detection targets. Classification is used to predict the confidence level of each anchor, and regression

is used to predict the offset between the anchor and the bounding box at each location. The expression for calculating the predicted box position based on the offset is shown in equation (5).

$$\begin{cases} b_x = \sigma(t_x) + c_x \\ b_y = \sigma(t_y) + c_y \\ b_w = p_w \times e^{t_w} \\ b_h = p_h \times e^{t_h} \end{cases} \tag{5}$$

In equation (5), c_x and c_y are the coordinates of the points on the current grid. p_w and p_h are the width and height of the anchors. b_x and b_y are the adjusted center coordinates of the predicted box. b_w and b_h are the width and height of the adjusted prediction box. To improve the performance of the prediction branch, this study introduces a residual module to supersede the first convolutional layer of the prediction branch with a ResBlock, and optimizes the feature maps input for candidate box regression and classification tasks to achieve the performance of the prediction branch. The network structure of Cot-CSPDarknet feature extraction is shown in Figure 3.

In Figure 3, this study first optimizes the backbone network by introducing the Cot module that focuses more on contextual information, thereby constructing a new Cot-CSPDarknet. CSPDarkNet is an improved deep learning network architecture derived from the original DarkNet framework, specifically designed to enhance feature extraction capabilities. It divides the feature mapping into multiple parts through the Cross-Stage Partial Network (CSP) technology and conducts different feature learning in each part, thereby enhancing the model's ability to capture complex features. To more effectively utilize shallow feature information, an AM-SE module is introduced in the feature fusion stage after adding features extracted by downsampling at different magnifications. This helps reduce interference from irrelevant features such as background. This study utilizes clustering algorithms to regenerate prior boxes that are more suitable for small targets in the dataset. To further improve the accuracy of upsampling, the original nearest neighbor interpolation method has been replaced with sub-pixel convolution. Finally, to enhance the performance of the prediction layer, the first convolutional block of the prediction layer is used to replace the residual unit.

In the above content, the overall structure of the ADMT model as an MTPN is introduced, emphasizing that its design purpose is to achieve multiple perception tasks simultaneously through a unified network structure. These tasks include object detection, semantic segmentation, depth estimation, etc. In the complex AD environment, these tasks are interdependent and require comprehensive consideration by the network to improve the accuracy and efficiency of perception. By sharing the feature extraction process, the model can effectively reduce the consumption of computing resources and improve the collaborative effect among various tasks. After constructing the overall framework of the ADMT model, the next step will be to explore its internal structure and modules in detail, in order to achieve collaborative processing of multiple tasks. Especially the feature decoupling and fusion module, and how to effectively allocate feature weights for subtasks such as object detection, instance segmentation, and object tracking.

3.2 Environment perception decoupling fusion algorithm based on AM

Cot-CSPDarknet can enhance the recognition of small targets, but it cannot solve the problem of mutual interference between tasks in multitasking environments. Cot-CSPDarkNet cannot effectively solve the problem of mutual interference among tasks in a multi-task environment. The main reason is that the feature sharing strategy adopted may result in different focus on feature information for each task, leading to the information of some tasks being overwhelmed by other tasks. Furthermore, the gradient influence of the loss functions of different tasks on the shared features can lead to interference in the optimization process, making it difficult for the model to balance the learning requirements of multiple tasks. Finally, the lack of feature decoupling ability limits the adaptability of the model in complex environments and reduces the efficiency of full utilization of information among tasks. AM can reduce information overload by focusing on key information, enabling neural networks to process input data more effectively and improve task processing efficiency and accuracy [23]. To solve the multitasking problem in AD, this study proposes an environment aware decoupling fusion algorithm based on AM, and its module structure is displayed in Figure 4.

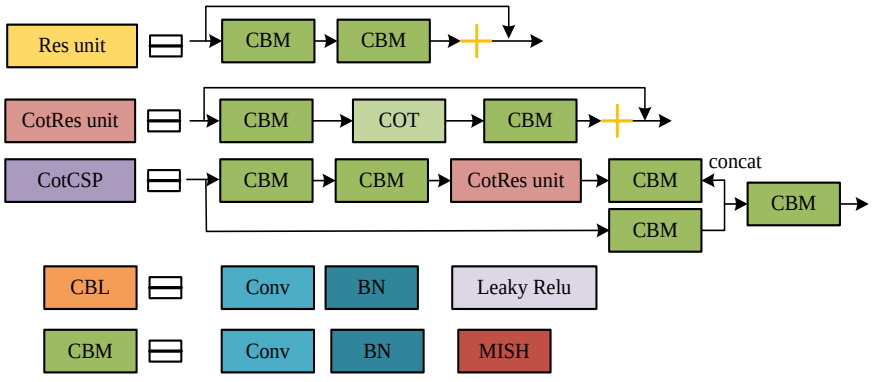


Figure 3: Cot module and its convolutional layer structure.

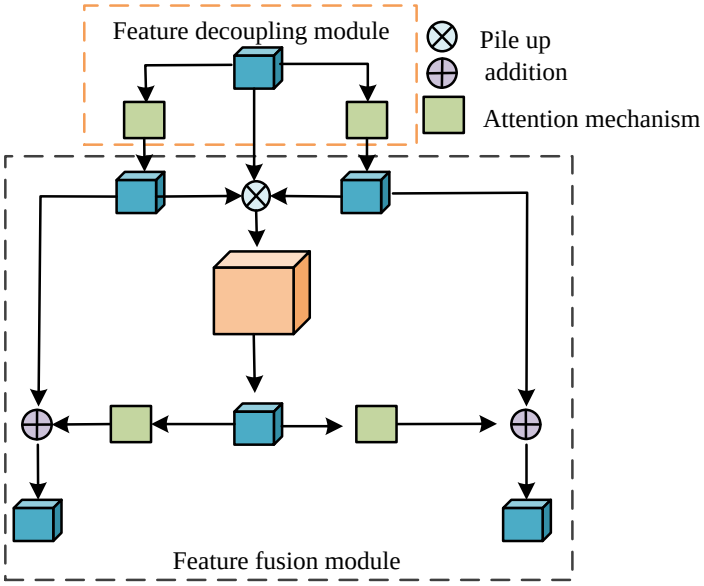


Figure 4: Block diagram of feature decoupling and fusion module.

In Figure 4, the spatiotemporal features are input into the feature decoupling module and weighted through AM to obtain the features of the detection object. To capture the intrinsic connections between features, it is necessary to use feature pools. The feature pool is the fused spatiotemporal features, and AM is applied during feature selection. Then, decoupled features are fused to obtain features suitable for various tasks. This study employs Efficient Channel Attention (ECA). ECA proposes a cross channel interaction method that does not require dimensionality reduction, and replaces the fully connected layer in the SE with a 1D convolutional structure. This not only simplifies the model structure but also improves network performance. The ECA module uses the softmax function for feature mapping, and its calculation is shown in equation (6).

$$R = \text{softmax}\left(W_2 \,\text{ReLU}\left(W_1 y\right)\right) \tag{6}$$

In equation (6), R is the feature map. W_1 and W_2 are hyperparameters. y is the result of global average pooling of image features. ReLU is the activation function. The flowchart of the feature decoupling module is exhibited in Figure 5.

In Figure 5, after being processed by a dual AM, spatiotemporal features are weighted for application in detecting objects. The features processed by the decoupling fusion unit are assigned different weights and then sent to three fully connected networks to perform specific sub-tasks. When performing feature decoupling, the object detection network first performs heatmap prediction to determine the approximate position, and then estimates the offset to lift the localization accuracy. Finally, the target size is estimated based on the model to collect detailed information about the target, ensuring comprehensive and accurate object detection. The instance segmentation module uses the offset calculated by the extraction module for data matching, and finally generates the predicted target trajectory. To balance the training speed of multiple tasks, this study uses dynamic weighted averaging to optimize the network as a whole. "Dynamic weighted averaging" is achieved by assigning a dynamic weight to each input feature, and these weights are adjusted according to the importance of the feature and the context information. Firstly, the initial weights of each feature are calculated, and then these weights are dynamically updated through learning mechanisms such as neural networks or AM to reflect the relative importance of specific tasks or data moments. Ultimately, the calculation process of the dynamic weighted average is to multiply the eigenvalues by their corresponding weights and then sum them up. Finally, it is normalized and divided by the sum of weights to obtain an accurate average value. The calculation of minimizing the loss function is shown in equation (7).

$$l_{total} = \varpi_t\left(t\right)\cdot l_{td} + \varpi_{is}\left(t\right)\cdot l_{is} + \varpi_{tf}\left(t\right)\cdot l_{tf} \tag{7}$$

In equation (7), l_{total} is the total loss function. The total loss function summarizes the losses of the model on different tasks and is used to evaluate the overall performance of the model, ensuring that all tasks receive reasonable attention and optimization. l_{td}, l_{is}, and l_{tf} are the loss function for object detection, instance segmentation, and target tracking. The object detection loss function measures the model's ability to recognize objects in an image, typically including the accuracy of the detection box and the prediction accuracy of the corresponding object category. The instance segmentation loss function evaluates the ability of the model to distinguish different instances of the same type of target and perform precise segmentation. The target tracking loss function is used to measure the model's ability to maintain the tracking target in the video stream. $\varpi\left(t\right)$ represents the weight of the loss function at time *t*. These weights dynamically adjust the contribution of each task to the

total loss function based on the importance and performance of each task. The multitasking algorithm framework is shown in Figure 6.

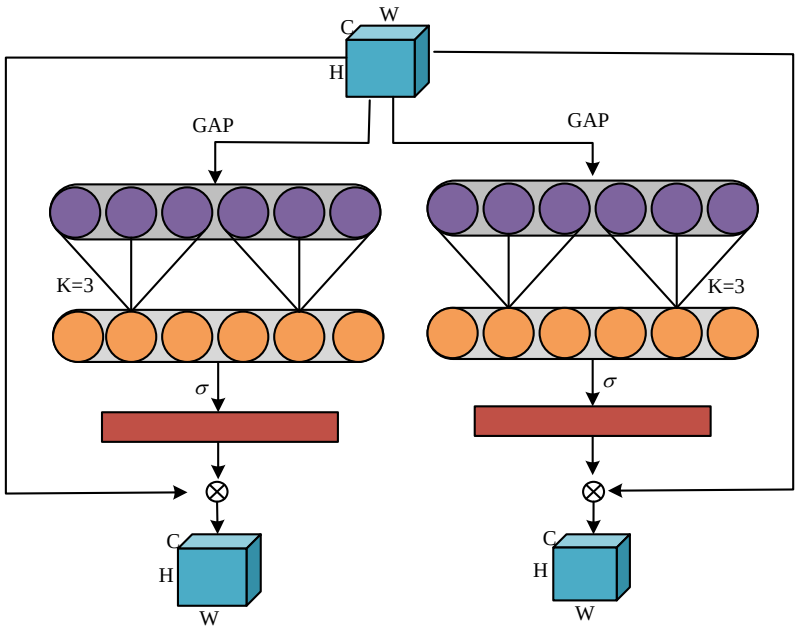


Figure 5: Feature decoupling module.

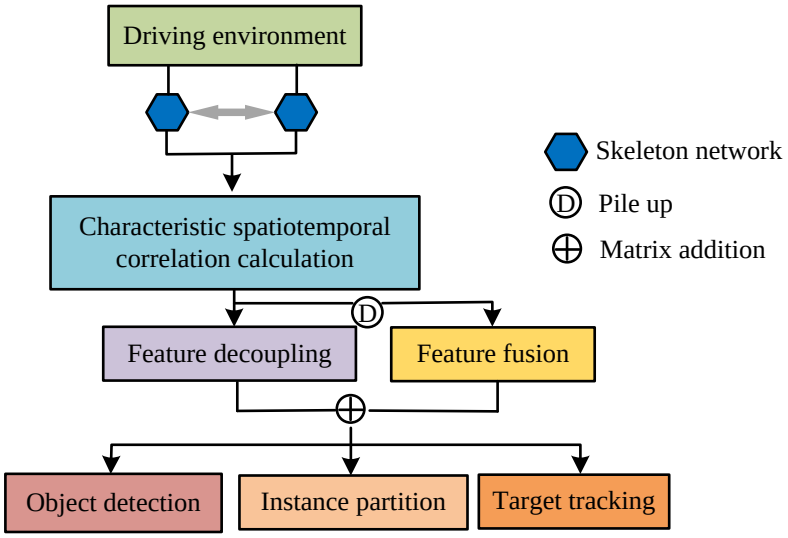


Figure 6: Multi-task learning framework.

In Figure 6, first, feature spatiotemporal correlation is performed based on the driving environment, then spatiotemporal features are decoupled and fused, and finally three tasks are performed: object detection, instance segmentation, and object tracking. The corresponding weights of the loss functions for these three tasks are shown in equation (8).

$$\varpi_i(t)=\frac{\mathrm{N}\cdot\exp\left(r_i(t-1)/T\right)}{\sum_n \exp\left(r_n(t-1)/T\right)} \tag{8}$$

In equation (8), r_i is the training speed, i is the task, $(t-1)$ is a certain moment, and N is the number of tasks. T stands for temperature scaling. Its value can adjust the competition intensity among tasks. A smaller value will enhance the difference in training speed and accelerate the convergence of simple tasks. Larger values alleviate the imbalance of learning rates among tasks and prevent certain tasks from being completely ignored. The dynamic weighted average method is achieved by evaluating the weight of each task's loss function, with a lower weight indicating a faster training process for that task. The calculation process of training speed is expressed in equation (9).

$$r_n(t-1)=\frac{l_n(t-1)}{l_n(t-2)} \tag{9}$$

In equation (9), l_n is the corresponding loss function for a certain iteration cycle. The research adopts the design

of the Feature Decoupling Fusion Module (FDFM) based on the feature decoupling theory of multi-task learning and the collaborative optimization principle of the AM. The theoretical core lies in explicitly separating the task-specific features from the shared features, and using the local cross-channel interaction of the ECA module and the global channel recalibration of the SE module to enhance the positioning accuracy of small targets and suppress background noise respectively. The decoupling stage uses a gating mechanism to generate sparse features of the task. In the fusion stage, cross task self attention is introduced to model collaboration between tasks, and multi task losses are balanced through dynamic uncertainty weighting to avoid bias in manual parameter adjustment. This architecture alleviates task conflicts and enhances the feature saliency of occlusions and distant small targets through hierarchical feature optimization and adaptive weight allocation.

4 ADMT effectiveness analysis

4.1 Multi-task perception model performance testing

To analyze the performance of ADMT at runtime, this study uses the KITTI and Cityscapes datasets as test data. For the KITTI dataset, the main tasks include object detection and depth estimation, as this dataset is typically used for object recognition and 3D reconstruction in traffic scenes. For the Cityscape dataset, the main task is semantic segmentation, with a focus on image understanding and pixel level classification in urban environments. To optimize the hyperparameters of the model and improve its performance, the study adopts the Bayesian optimization strategy to automate the hyperparameter search process. The research systematically optimizes key hyperparameters such as the learning rate, batch size, and momentum by constructing the objective function of hyperparameters. The Bayesian optimization method uses the previous experimental results to guide the subsequent search, thereby effectively exploring the hyperparameter space. In the experiment, the study sets the initial value range of the learning rate from 0.0001 to 0.01. The batch size ranges from 16 to 64, and the momentum parameter ranges from 0.8 to 0.99. Through this strategy, the optimal combination of hyperparameters can be quickly found, resulting in the optimal setting of a learning rate of 0.001, batch size of 32, and momentum of 0.9, thereby improving the convergence speed and performance of the model. The research adopts the standard division strategies of training set, validation set, and test set to ensure the generalization ability of the model on different data. The KITTI and Cityscapes datasets are divided in the proportions of 70% for training, 15% for validation, and 15% for testing. The training set is used for the learning and optimization of the model, while the validation set regularly evaluates the model performance during the training process to achieve the best effect. The test set is used to ultimately evaluate the generalization ability and performance of the model. It will not be viewed or adjusted during the testing phase. Table 1 shows the experimental environment and equipment.

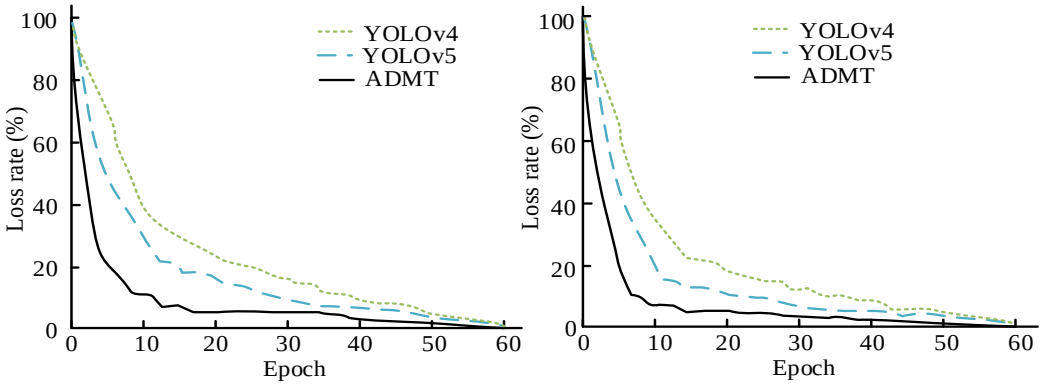
To test the performance of ADMT, this study conducts training and testing. After 60 iterations, the ADMT model is compared and analyzed with YOLOv4 [24] and YOLOv5 [25]. The experiment compares the losses of three methods under different iterations. The loss values include the KITTI dataset and the Cityscapes dataset. These losses are measured by the cross-entropy loss function. The specific calculation method is to compare the category probabilities predicted by the model with the real labels, thereby quantifying the performance of the model during the training process. The experimental results are shown in Figure 7.

Figure 7 shows the trend of loss variation with iteration times for three different methods. As the Number of Iterations (NoI) increases, the loss values of all methods show a decreasing trend, indicating that the model gradually learns and optimizes during the training process. When the NoI is relatively small, especially within 10 iterations, the loss reduction curve is steeper, indicating that in the early phases of training, the model's loss reduction rate is faster and the learning efficiency is higher. However, when the NoI exceeds 10, the curve of loss reduction becomes relatively flat, indicating that as the training progresses, the rate of loss reduction in the model gradually slows down. Until after 60 iterations, the loss value approaches 0, indicating that the model is approaching convergence. In Figure 7, at iteration times of 10 and 20, the losses of YOLOv4 and YOLOv5 are 38% and 20%, 28% and 12%, while the losses of ADMT are 10% and 5%, significantly lower than the comparison algorithm. This indicates that the ADMT model can reduce losses during iterations. This study compares the MOTA, F1 score, and APiou=0.5 scores of different algorithms on KITTI and Cityscapes, as shown in Figure 8.

Table 1: Engineering project setting conditions.

Name	Parameter
Simulation program	Matlab2020
Computer	Intel (R) Core (TM) i5-6200U CPU@ 2.30GHz
Dataset	KITTI, Cityscapes
Operating system	Ubuntu 16.04
CPU	Intel.Xeon, E5_2 660.v3
GPU	NVIDIA TESLA V100

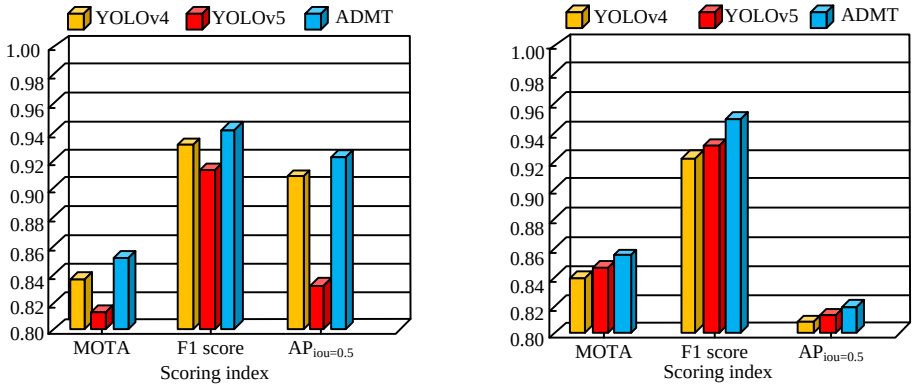
Video memory	16 GB
Deep learning framework	Tensorflow



(a) The loss of the model on the test set of the training phase

(b) The loss of the model on the test set of the Test phase

Figure 7: Loss comparison of different methods in training and testing phases.



(a) Score on KITTI data set

(b) Scores on the Cityscapes dataset

Figure 8: Comparisonof existing methods tested on KITTI and Cityscapes datasets.

Figure 8 shows the performance between the ADMT model and the other two methods on the KITTI and Cityscapes datasets. On KITTI, the MOTA, F1 score, and APiou=0.5 scores of the ADMT model are 0.85, 0.94, and 0.92. These scores are the highest among the three comparison methods, demonstrating the superior performance of the ADMT model in multi-target tracking and object detection tasks. In contrast, the corresponding scores of YOLOv4 and YOLOv5 are 0.834, 0.93, 0.904 and 0.81, 0.91, 0.83, all lower than the ADMT model. These results indicate that the ADMT model has higher accuracy and robustness in handling object detection and tracking tasks in complex road environments. On Cityscapes, the MOTA, F1 score, and APiou=0.5 scores of the ADMT model are 0.852, 0.946, and 0.818. These scores also demonstrate the good performance of the ADMT model in urban scenarios. These datasets contain complex scenes of urban streets, including targets of different sizes and shapes, as well as various occlusion and truncation situations. Therefore, these scores further validate the effectiveness and applicability of the ADMT in practical AD scenarios.

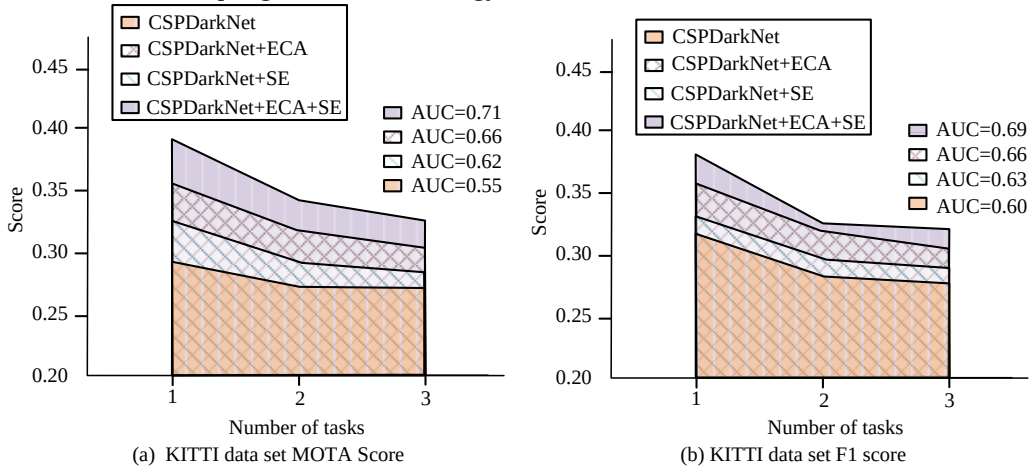
4.2 Application analysis of multi-task perception model

To validate the application effectiveness of each module used, the study conducts ablation experiments. The datasets used in the experiment are KITTI and Cityscapes datasets, as shown in Figure 9.

The ablation experiment in Figure 9 compares the independent and collaborative contributions of each component to the model performance by gradually introducing different modules into the system. The results show that when CSPDarkNet is used alone as the base network, the MOTA and F1 scores on the KITTI dataset are 0.55 and 0.60, respectively, indicating its basic feature extraction ability as a backbone network. When the ECA module is introduced, the MOTA's value increases to 0.66 (+20%), and the F1 score rises to 0.66 (+10%). This is attributed to the fact that ECA achieves cross-channel interaction through lightweight 1D convolution, optimizes the dynamic selection ability of key features, and particularly enhances the positioning accuracy of small targets. When the SE module is added alone, MOTA and F1 scores reach 0.62 (+12.7%) and 0.63 (+5%),

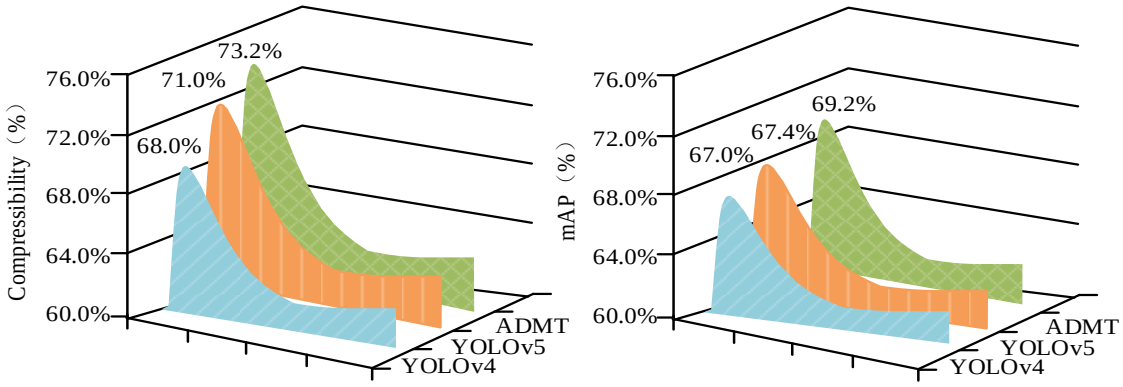
respectively. Its channel AM suppresses redundant background features by explicitly modeling the channel dependency relationship. When the ECA and SE modules are used in combination, MOTA and F1 scores further increase to 0.71 (+29.1%) and 0.69 (+15%), respectively, revealing the complementarity of the two AMs: ECA focuses on local cross-channel feature interaction, while SE enhances semantic importance through global channel re-calibration. The synergy of the two significantly improves the feature representation ability of multi-scale targets in complex scenarios. This ablation experiment not only verifies the effectiveness of each module design but also quantifies the hierarchical optimization mechanism of the AM in the feature decoupling and fusion strategy, providing an empirical basis for the interpretability of the model components. The results of comparing the compression ratio and mean Average Precision (mAP) of different methods are shown in Figure 10.

In Figure 10, the compression rate of ADMT is 73.2%, which is higher than the compression rates of 71.0% for YOLOv5 and 68.0% for YOLOv4. The mAP of ADMT is 69.2%, higher than the 67.4% of YOLOv5 and 67.0% of YOLOv4. This indicates that ADMT has good compression performance in object detection and is more stringent on target boxes detected by the network. Figure 11 shows the percentage of targets identified using ADMT on KITTI and Cityscapes.



(a) KITTI data set MOTA Score (b) KITTI data set F1 score

Figure 9: Comparative experiment of decoupling modules in KITTI and Cityscapes datasets.



(a) Compression ratio comparison of different detection methods (b) mAP comparison of different detection methods

Figure 10: Compression ratio and mAP comparison of different detection methods.

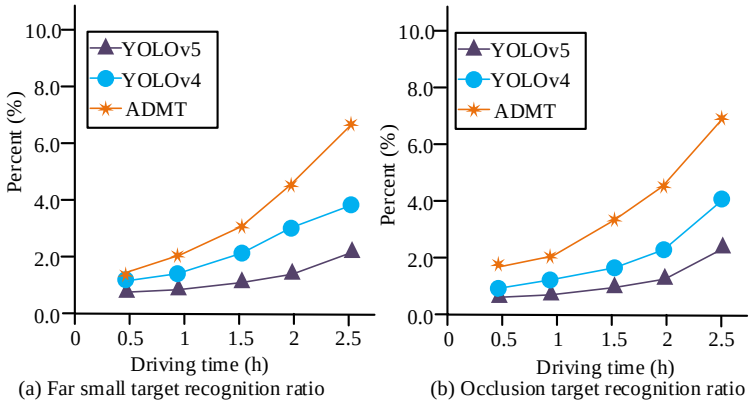


Figure 11: The proportion of targets identified at different driving times.

Figure 11 shows the comparison of the recognition percentages of far small targets and occluding targets by ADMT and YOLOv4 and YOLOv5 on the KITTI and Cityscapes datasets. The x-axis in the figure represents "Driving time (h)", which represents the cumulative testing time of the model in continuous driving scenarios, used to simulate the changes in target recognition performance of AD vehicles when driving for a long time in real road environments. Figure 11(a) shows the recognition of far small targets. The recognition rate of ADMT within a 2.5-hour driving time is 6.6%, which is higher than that of YOLOv5 (2.2%) and YOLOv4 (3.8%). This advantage stems from the enhanced extraction of context features by Cot-CSPDarkNet and the refined reconstruction ability of low-resolution features by sub-pixel convolution. Figure 11(b) shows the recognition of occlusion targets. The recognition rate of ADMT reaches 7.0%, while that of YOLOv5 and YOLOv4 is 2.4% and 4.1%, respectively. This verifies the effectiveness of the FDFM module in dynamically decoupling the occluded area from the background noise through attention weights. Meanwhile, it indicates that the synergistic effect of the ECA and SE modules can improve the fusion accuracy of local features and global semantics. Although the above results have improved the recognition accuracy of small objects and occlusions, on the whole, they are still relatively low. The main reasons lie in the deviation of data distribution, the limitations of model structure, and the boundaries of multi-task optimization. In terms of data, the proportion of far small targets and severely occluded samples in the KITTI and Cityscapes datasets is less than 5%, resulting in insufficient learning of extreme scenes by the model. The limitation of sensor resolution causes the details of small targets to be blurred. In terms of model structure, the channel compression loss of the Cot module is small for the shallow texture features of the target. The sub-pixel convolution is insufficient for the reconstruction of low-resolution features. The AM is prone to interference from similar backgrounds in densely occluding areas. In terms of task conflicts, the dynamic weighting strategy leans towards tasks with fast training speeds and ignores the refinement requirements of instance segmentation for occlusion boundaries. The visualization diagram of the research method is shown in Figure 12.

Several key elements are involved in Figure 12, including the detected target box, the corresponding category label, and the confidence score. Boxes of different colors represent the model's recognition of different types of targets, demonstrating the model's ability to handle complex traffic scenarios, such as effectively distinguishing pedestrians, vehicles, and other obstacles during dynamic driving. In addition, the instance segmentation part reflects the precise boundary division of the model on the target, thus more clearly depicting the shape and boundary of the target, which is conducive to improving the tracking accuracy of subsequent tasks. Through these visualization effects, Figure 12 not only reveals the effectiveness of ADMT but also provides important analytical basis and visualization feedback for further optimizing the model and field applications. To further verify the feasibility of the proposed method, the Waymo dataset is adopted for more in-depth verification. The Waymo dataset is a large-scale AD dataset released by Waymo to support advanced AD research. This dataset contains over 250,000 scenarios, covering various complex urban and highway environments, with a focus on traffic participants such as pedestrians, cyclists, and vehicles. The dataset provides high-quality Lidar and camera data, supporting a variety of perception tasks, including object detection, instance segmentation, and tracking. Meanwhile, the diversity of the Waymo dataset, such as different weather conditions, illumination changes, and occlusion situations, provides a good basis for the robustness test of the algorithm. The comparative methods adopted in the research are the intelligent driving simulation test platform proposed by Xudong Y U et al. [19], the CACC method based on convex programming proposed by Yang T et al. [22], and the MDSTF trajectory prediction model proposed by Wang X et al. [21]. The specific comparison results of the experiment are shown in Table 2.

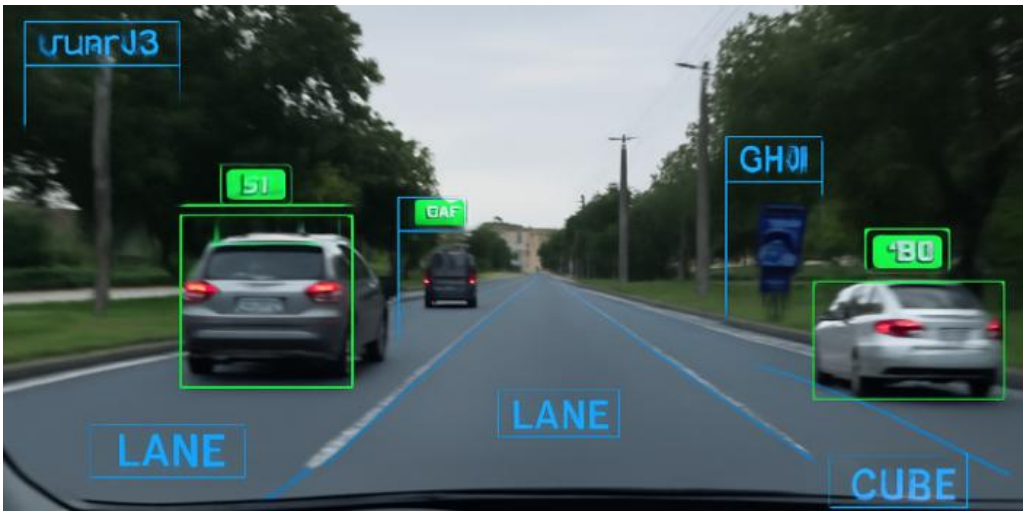


Figure 12: Study the visualization results of the model.

Table 2: Performance comparison results of different models.

Indicator	ADMT	Xudong Y U et al.[19]	Yang T et al. [20]	Wang X et al. [21]
Inference Speed (FPS)	45 FPS	30 FPS	38 FPS	35 FPS
Parameter size (MB)	10.5 MB	12.2	9.8	11.3
FLOPs	25.4 B	30.2 B	22.8 B	27.0 B
IoU	0.88	0.81	0.82	0.82
Accuracy rate of occlusion situation recognition	75.2%	60.6%	70.4%	65.4%
Recognition accuracy rate of light changes	80.5%	65.3%	72.4%	68.9%

Table 2 shows that the proposed ADMT model outperforms other comparison methods in multiple key performance indicators. Firstly, in terms of inference speed, the 45 FPS of ADMT is significantly higher than the 30 FPS of Reference [19] and the 35 FPS of Reference [21], demonstrating the advantage of ADMT in processing speed and helping to meet the real-time requirements of AD systems. Secondly, in terms of parameter size and FLOPs, the parameter count (10.5 MB) and FLOPs (25.4 B) of ADMT are more optimized compared to other methods, indicating that it is more efficient in terms of computing resource consumption. Furthermore, ADMT achieves 0.88 in the IoU index, surpassing the performance of other methods and proving its stronger ability in target detection and tracking accuracy. Especially in the task performance in complex environments, the accuracy rate of ADMT in recognizing occlusion situations (75.2%) and recognition accuracy under illumination changes (80.5%) is significantly better than that of other models. This indicates that ADMT has stronger robustness and adaptability under adverse conditions. These results indicate that the ADMT model not only performs well in standard perception tasks but also has higher reliability and accuracy when dealing with the challenges of complex driving environments.

5 Discussion

The ADMT model based on CSPDarkNet and the AM proposed in the research is significantly superior to the existing mainstream methods, such as YOLOv4 and YOLOv5, in multiple tasks. On the KITTI and Cityscapes datasets, ADMT achieved 0.94 and 0.92 respectively in performance metrics such as F1score and AP_{iou}=0.5, demonstrating its efficiency in object detection and tracking tasks and robustness in complex environments. Through the feature decoupling and fusion module, ADMT effectively and dynamically allocates and weights the features of different tasks, reduces information interference, and enhances the focus of key features of each task. This innovative design enhances the generalization ability of the model, enabling it to perform more accurately in diverse traffic situations. The introduced AM enables the model to focus on key information, improves the recognition ability of small targets and occluded targets, and enables ADMT to maintain efficient perception performance under changing environmental conditions. In similar studies, the method based on DRL proposed by Khan MA et al. has achieved good results in balancing driving efficiency and comfort. However, this method still faced challenges in the accuracy of target detection in complex environments [11]. In contrast, ADMT significantly improves the

perceptual performance in complex scenarios by comprehensively considering multi-task features and utilizing the AM. The anchor-free detection network proposed by Wang H et al. performed relatively well in the detection of occlusive objects. However, its model had a high complexity and might face the bottleneck of real-time processing in practical applications [12]. By optimizing the network structure and introducing an efficient feature decoupling mechanism, ADMT has achieved a better detection rate and response speed in the case of target occlusion, demonstrating stronger real-time processing capabilities. The AD decision-making framework proposed by Yang K et al. utilized reinforcement learning algorithms to enhance the safety of highway driving, but it lacked systematic methods in multi-task processing [13]. In contrast, ADMT demonstrates its ability to cope with different driving decision-making tasks in complex environments through centralized multi-task learning and feature fusion of task relevance, indicating the superiority of its own method. To sum up, ADMT outperforms the existing related methods in multiple performance indicators, demonstrating its broad application potential and technical value in the field of AD.

6 Conclusion

This study designed an ADMT aimed at improving the safety of AD and the precision of target recognition. This model established a FEN by integrating the Cot module, which optimized the prediction branch of YOLOv4 using the full convolution structure. In addition, the model also incorporated an AM SE module to perfect the recognition capacity of the target. To address potential conflicts between multiple tasks, the model specifically added FDFM. In the experiment, the compression rate of the model reached 73.2%, with an mAP of 69.2%. The model outperformed YOLOv5 and YOLOv4 algorithms in terms of compression rate and average accuracy. When the driving time was 2.5 hours, the proportion of ADMT recognizing far and small targets was 6.6%, and the proportion of recognizing occluded targets was 7.0%. These data indicated that the model not only had good compression performance in object detection but also had stricter requirements for target boxes detected by the network, effectively increasing the recognition of occluded truncated targets and far small targets. The results show that the Cot-CSPDarkNet constructed by the study enhances the context feature extraction ability of small targets through the context transformation module and the convolution of sub-pixels. The FDFM designed in the research combines a dual-pathway AM to dynamically separate task-specific features from shared features, alleviating multi-task conflicts. A task uncertainty weighting strategy based on temperature scaling is proposed to balance the differences in training speeds of detection, segmentation, and tracking tasks. However, ADMT still has limitations. The negative migration effect between tasks is not explicitly modeled, and the feature decoupling in extreme occlusion scenarios may fail. The dynamic weighting strategy relies on a fixed temperature parameter T and has insufficient adaptability to dynamic scenes such as sudden changes in illumination. The proportion of far small targets in the experimental dataset is relatively low, which affects the generalization ability of the model. Future work will focus on three areas. One is to introduce task association aware AM and clarify the decoupling process that constrains task features. The second is to design an adaptive temperature parameter strategy, combined with online learning to dynamically adjust the T value to adapt to complex road conditions. The third is to jointly generate adversarial networks to integrate high-density small targets and occlusion data, optimizing the robustness of the model under long tail distribution.

Fundings

The research is supported by: Research Project on Higher Education Reform in Jiangsu Province: Research and Practice of Teaching Reform for High-Quality Courses in Computer Science Majors Driven by Generative AI, (No.2025JGZD037).

References

[1] Manzoor Ahmed Khan, Hesham El-Sayed, Sumbal Malik, Muhammad Talha Zia, Muhammad Jalal Khan, Najla Alkaabi, and Henry Ignatious. Level-5 autonomous driving-are we there yet? A review of research literature. ACM Computing Surveys (CSUR), 55(2):1-38, 2022.https://doi.org/10.1145/3485767

[2] Sushila Umesh Ratre, and Bharti Joshi. Trajectory and motion prediction of autonomous vehicles driving assistant system using distributed discriminator based bi-directional long short term memory model. International Journal of Intelligent Transportation Systems Research, 23(1):245-258, 2025.https://doi.org/10.1007/s13177-024-00447-8

[3] Hebing Liu, Jinhong Sun, Heshou Wang, and Ka Wai Eric Cheng. Comprehensive analysis of adaptive soft actor-critic reinforcement learning-based control framework for autonomous driving in varied scenarios. IEEE Transactions on Transportation Electrification, 11(1):3667-3679, 2025.https://doi.org/10.1109/TTE.2024.3429186

[4] Wujie Zhou, Shaohua Dong, Jingsheng Lei, and Lu Yu. MTANet: Multitask-aware network with hierarchical multimodal fusion for RGB-T urban scene understanding. IEEE Transactions on Intelligent Vehicles, 8(1):48-58, 2022.https://doi.org/10.1109/TIV.2022.3164899

[5] Wujie Zhou, Shaohua Dong, Jingsheng Lei, and Lu Yu, MTANet: Multitask-aware network with hierarchical multimodal fusion for RGB-T urban scene understanding. IEEE Transactions on Intelligent Vehicles, 8(1):48-58, 2022.https://doi.org/10.1109/TIV.2022.3164899

[6] Menghao Guo, Tianxing Xu, Jiangjiang Liu, Zhengning Liu, Pengtao Jiang, Taijiang Mu, Songhai Zhang, Ralph R. Martin, Mingming Cheng, and Shimin Hu. Attention mechanisms in computer

vision: A survey. Computational Visual Media, 8(3):331-368, 2022.https://doi.org/10.1007/s41095-022-0271-y

[7] Tianyu Yang, Yeqiang Qian, Weihao Yan, Chunxiang Wang, and Ming Yang. AdaptiveOcc: Adaptive octree-based network for multi-camera 3D semantic occupancy prediction in autonomous driving. IEEE Transactions on Circuits and Systems for Video Technology, 35(3):2173-2187, 2025.https://doi.org/10.1109/TCSVT.2024.3492289

[8] Harshpreet Kaur, and Munish Bhatia. Digital-twin-driven performance assessment for IoT-integrated autonomous driving systems. IEEE Internet of Things Journal, 12(4):4078-4085, 2025.https://doi.org/10.1109/JIOT.2024.3481501

[9] Manzoor Ahmed Khan, Hesham El-Sayed, Sumbal Malik, Muhammad Talha Zia, Muhammad Jalal Khan, Najla Alkaabi, and Henry Ignatious. Level-5 autonomous driving-are we there yet? A review of research literature. ACM Computing Surveys (CSUR), 55(2):1-38, 2022.https://doi.org/10.1145/3485767

[10] Long Chen, Yunqing Zhang, Bin Tian, Yunfeng Ai, Dongpu Cao, and Fei-Yue Wang. Parallel driving OS: A ubiquitous operating system for autonomous driving in CPSS. IEEE Transactions on Intelligent Vehicles, 7(4):886-895, 2022.https://doi.org/10.1109/TIV.2022.3223728

[11] Guofa Li, Shenglong Li, Shen Li, and Xingda Qu. Continuous decision-making for autonomous driving at intersections using deep deterministic policy gradient. IET Intelligent Transport Systems, 16(12):1669-1681, 2022.https://doi.org/10.1049/itr2.12107

[12] Hai Wang, Yansong Xu, Zining Wang, Yingfeng Cai, Long Chen, and Yicheng Li. Centernet-auto: A multi-object visual detection algorithm for autonomous driving scenes based on improved centernet. IEEE Transactions on Emerging Topics in Computational Intelligence, 7(3):742-752, 2023.https://doi.org/10.1109/TETCI.2023.3235381

[13] Kai Yang, Xiaolin Tang, Sen Qiu, Shufeng Jin, Zichun Wei, and Hong Wang. Towards robust decision-making for autonomous driving on highway. IEEE Transactions on Vehicular Technology, 72(9):11251-11263, 2023.https://doi.org/10.1109/TVT.2023.3268500

[14] Yongjun Ji, Zuhua Jiang, Xinyu Li, Yongwen Huang, and Fuhua Wang. A multitask context-aware approach for design lesson-learned knowledge recommendation in collaborative product design. Journal of Intelligent Manufacturing, 34(4):1615-1637, 2023.https://doi.org/10.1007/s10845-021-01889-7

[15] Hancheng Ye, Bo Zhang, Tao Chen, Jiayuan Fan, and Bin Wang. Performance-aware approximation of global channel pruning for multitask cnns. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(8):10267-10284, 2023.https://doi.org/10.1109/TPAMI.2023.3260903

[16] Donggun Park, Yushin Lee, and Yong Min Kim. Effects of autonomous driving context and anthropomorphism of in-vehicle voice agents on intimacy, trust, and intention to use. International Journal of Human-Computer Interaction, 40(21/22):7179-7192, 2024.https://doi.org/10.1080/10447318.2023.2262271

[17] Enneng Yang, Junwei Pan, Ximei Wang, Haibin Yu, Li Shen, Xihua Chen, Lei Xiao, Jie Jiang, and Guibing Guo. Adatask: A task-aware adaptive learning rate approach to multi-task learning. Proceedings of the AAAI Conference on Artificial Intelligence, 37(9):10745-10753, 2023. https://doi.org/10.1609/aaai.v37i9.26275

[18] Yansong Pei, Junbo Zhao, Yiyun Yao, and Fei Ding. Multi-task reinforcement learning for distribution system voltage control with topology changes. IEEE Transactions on Smart Grid, 14(3):2481-2484, 2023.https://doi.org/10.1109/TSG.2022.3233766

[19] Xudong Yu, Guang Yin, Lie Guo, Jian Huang, and Jian Zhao. Establishment and application research of a six-degree-of-freedom autonomous driving simulation test platform. Experimental Technology and Management, 41(10):150-156, 2024.https://doi.org/10.16791/j.cnki.sjg.2024.10.019

[20] Tianci Yang, Carlos Murguia, Dragan Nešić, and Chau Yuen. Toward crash-free autonomous driving: Anomaly detection and control for resilience to stealthy sensor attacks. IEEE Internet of Things Journal, 12(1):276-287, 2025.https://doi.org/10.1109/JIOT.2024.3459590

[21] Xing Wang, Zixuan Wu, Biao Jin, Mingwei Lin, Fumin Zou, and Lyuchao Liao. Correction to: MDSTF: A multi-dimensional spatio-temporal feature fusion trajectory prediction model for autonomous driving. Complex & Intelligent Systems, 10(5):7419-7420, 2024.https://doi.org/10.1007/s40747-024-01548-3

[22] V. Arulalan, and Dhananjay Kumar. Efficient object detection and classification approach using HTYOLOV4 and M2RFO-CNN. Computer Systems Science and Engineering, 44(2):1703-1717, 2023. https://doi.org/10.32604/csse.2023.026744

[23] Zhixian Zeng, Jianjun Cao, Nianfeng Weng, Zhen Yuan, and Xu Yu. Cross-modal entity resolution for image and text integrating global and fine-grained joint attention mechanism. Journal of Shanghai Jiaotong University (Science), 28(6):728-737, 2023.https://doi.org/10.1007/s12204-022-2465-y

[24] Mingsheng Liu, Liang Wan, Bo Wang, and Tingting Wang. SE-YOLOv4: shuffle expansion YOLOv4 for pedestrian detection based on PixelShuffle. Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies, 53(15):18171-18188, 2023.https://doi.org/10.1007/s10489-023-04456-0

[25] Shanshan Wang, Weiwei Tan, Tengfei Yang, Liang Zeng, Wenguang Hou, and Quan Zhou. High-voltage

transmission line foreign object and power component defect detection based on improved YOLOv5. Journal of Electrical Engineering & Technology, 19(1):851-866, 2024.https://doi.org/10.1007/s42835-023-01625-6