

Article

End-to-End Autonomous Decision-Making Method for Intelligent Vehicles Based on ResNet-CBAM-BiLSTM

Yigao Ning , Xibo Fang, Xuan Zhao *, Shu Wang and Jianbo Zheng

School of Automobile, Chang'an University, Xi'an 710018, China; ningyigao@chd.edu.cn (Y.N.); fangxibo@chd.edu.cn (X.F.); shuwang@chd.edu.cn (S.W.); 2024122002@chd.edu.cn (J.Z.)

* Correspondence: zhaoxuan@chd.edu.cn

Abstract

To solve the difficulty of autonomous decision-making caused by the complex driving environment and changeable weather conditions, an end-to-end autonomous decision-making method based on residual network (ResNet), convolutional block attention module (CBAM) and bidirectional long short-term memory network (BiLSTM) is proposed for intelligent vehicles. Firstly, ResNet is used to extract spatial feature information contained in driving scene images. Then, CBAM is adopted to assign weights to each network channel and dynamically focus on important spatial regions in the image. Finally, BiLSTM is constructed to process the contextual features of continuous scenes, and the autonomous decision-making of intelligent vehicles is achieved through the fusion of spatial features and temporal information. On this basis, the proposed network model is trained using a real-world driving dataset and fully tested in various scenarios. Moreover, ablation experiments are conducted to verify the contribution of each module to the overall performance. The results show that the proposed method has better accuracy and stability compared with multiple existing methods, including PilotNet, FCN-LSTM, and DBNet, and its accuracy reaches 90.16% under clear weather conditions, as well as 81.29% under nighttime and snowy weather conditions.

Keywords: intelligent vehicles; end-to-end; decision-making; ResNet-CBAM-BiLSTM; multi-scenario verification

1. Introduction

The development and operation of intelligent vehicles are of great significance for improving traffic efficiency and reducing traffic accidents [1–3]. However, the driving scenarios of intelligent vehicles are complex, as they are not only affected by dynamic traffic environment in terms of their driving status, but also challenged in driving safety by various weather conditions, which makes their autonomous decision-making extremely difficult [4–6]. Traditional intelligent driving decision-making is based on pre-set relevant rules, by which the longitudinal and lateral control quantities of vehicles are determined. The combination of multiple sensors is required for data fusion perception, and multiple complex rule libraries need to be set. The computational complexity is high and the applicability is low, making it difficult for traditional methods to meet the requirements of complex driving scenarios, high driving safety, and traffic efficiency for intelligent vehicles [7–10]. To this end, an end-to-end intelligent driving decision-making method has been proposed by some scholars, in which the images captured by the car camera are directly taken as input, feature information is extracted from the images through neural



Academic Editor: Xiaozheng Jin

Received: 9 January 2026

Revised: 21 January 2026

Accepted: 28 January 2026

Published: 1 February 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

networks, and is then mapped to vehicle speed and steering wheel control variables. It does not require complex data processing processes and can avoid the setting of complex rules. This approach has become an important research direction in the field of intelligent driving [11–13].

NVIDIA [14] was the first company to propose an end-to-end autonomous driving model called PilotNet based on CNN, which receives front-facing images through three cameras configured in front of the vehicle and predicts the steering wheel angle of the vehicle. Subsequently, Bojarski et al. [15] proposed a region visualization method, explaining the learning process and decision-making principles of the PilotNet model. However, the PilotNet model did not take the prediction of vehicle speed into account. In actual driving, when the vehicle travels at a different speed according to the predicted steering angle, problems such as understeer or oversteer often occur. Therefore, it is necessary to consider the vehicle speed factor in end-to-end intelligent driving decision-making [16]. To this end, Kim et al. [17] proposed an intelligent driving decision model based on deep reinforcement learning, which uses the image of a single camera as the input of the model to achieve simultaneous decision-making of steering wheel angle and vehicle speed. However, with the increase in depth, traditional CNN models may encounter problems such as gradient explosion or vanishing due to the instability of gradients in backpropagation [18]. Moreover, the motion of cars is a series of time-dependent behaviors, and considering only the spatial information contained in the image is difficult to effectively meet the requirements of intelligent driving tasks for vehicles [19].

To simultaneously consider the temporal and spatial feature information during the driving process of intelligent vehicles, Xu et al. [20] proposed a neural network based on FCN-LSTM, which can effectively consider spatial and temporal information for decision-making of vehicle speed and steering angle. However, FCN-LSTM still exhibits insufficient perception of spatial depth information during training. In addition, Dai et al. [21] proposed the ST-LSTM network, which considers the complex spatial interactions and temporal relationships among vehicles, and improves the accuracy of trajectory prediction for intelligent vehicles in complex environments. But the ST-LSTM network needs to obtain the status information of the main vehicle and surrounding vehicles in advance, which is an extremely high requirement for communication technology and data processing algorithms. On this basis, Yan et al. [22] adopted two spatial attention mechanisms, context and channel, to model the interaction among vehicles, and process the temporal features of contexts through LSTM. However, traditional LSTM can only complete the propagation of information from historical states to current states and cannot backpropagate, which makes it difficult to fully utilize effective temporal information during vehicle motion. Thus, the further improvement of the ability for intelligent vehicles to combine spatial and temporal information processing is seriously restricted [23–25].

To fix the problem, an end-to-end autonomous decision-making method based on ResNet-CBAM-BiLSTM is established in this study, which takes the spatial information of the images captured by the onboard camera and the temporal information of the vehicle's continuous driving state as inputs to make decisions on vehicle speed and steering wheel angle.

The main contributions of this study are as follows:

1. An end-to-end autonomous decision-making method for intelligent vehicles based on ResNet-CBAM-BiLSTM is proposed, by which accurate decisions on motion control variables such as steering wheel angle and vehicle speed are made through driving scene images during vehicle operation.
2. By effectively combining the advantages of corresponding modules in image feature extraction, spatial attention optimization, and temporal information processing, the

problems of gradient explosion or vanishing, poor spatial feature extraction, and insufficient utilization of temporal information in existing end-to-end autonomous decision-making methods are addressed. As a result, the accuracy and stability of decision-making results are significantly improved.

3. The constructed network model is trained using a real-world driving dataset and is fully validated in representative scenarios such as daytime, nighttime, foggy, and snowy conditions. Mean Square Error (MSE) and accuracy are used to comprehensively assess the model's performance. Meanwhile, the performance of the proposed model is evaluated by comparison with three other representative end-to-end models using similar methods. Additionally, ablation experiments are designed to demonstrate the feasibility and advantages of the proposed method.

The remainder of this paper is structured as follows. In Section 2, an end-to-end autonomous decision-making model based on ResNet-CBAM-BiLSTM is established, and a dataset based on real driving scenarios was collected and used to train the established model. In Section 3, the proposed method was compared with three other representative end-to-end models in multiple scenarios with different features to fully validate the performance of the proposed method. Ablation experiments are designed to verify the performance improvement effect of different modules on the proposed model. Finally, some valuable conclusions are given in Section 4.

2. Methodology

2.1. End-to-End Autonomous Decision-Making Framework

An end-to-end autonomous decision-making method for intelligent vehicles based on ResNet-CBAM-BiLSTM is presented in this paper, where ResNet is used to extract spatial feature information, CBAM is adopted to assign weights to each network channel, and BiLSTM is constructed to process the contextual features of continuous scenes. On that basis, sufficient training is given for the established model using real driving data from existing dataset, thus the vehicle speed and steering wheel control quantities can be determined from images captured by forward-facing camera reasonably, and the difficulty caused by the complex driving environment and changeable weather condition can be solved effectively, as shown in Figure 1.

2.2. Construction of ResNet-CBAM-BiLSTM Model

The detailed architecture of the network model is shown in Figure 2. Firstly, ResNet extracts the spatial features of the image, and the CBAM calculates the weights of the channel dimension and spatial dimension separately to optimize the image features. Then, the processed convolutional feature maps are flattened into one-dimensional vectors as inputs to BiLSTM, and real driving data related to time are also used as inputs to BiLSTM. Finally, the steering wheel angle and vehicle speed are output from BiLSTM through a fully connected layer to achieve autonomous decision-making for intelligent vehicles.

2.2.1. Spatial Feature Extraction Module with ResNet

The safe driving of intelligent vehicles requires end-to-end autonomous decision-making models to exhibit high precision and strong robustness. To this end, a deeper network is required for the spatial feature extraction module. ResNet is based on traditional convolutional neural networks, which use residual structures to enable the output of a certain layer to directly skip one or more layers and connect to the input of subsequent layers, effectively avoiding the problems of gradient vanishing and exploding as depth increases [26,27]. Therefore, this study adopts ResNet as the spatial feature extraction module, and its architecture is shown in Figure 3.

The convolutional layer constitutes the core component of a convolutional neural network, where convolutional kernels are used to extract local spatial features from the input image, thereby enabling the transmission of feature information from one layer to the next. The convolution operation is computed as follows:

$$x_j^l = f_{\text{ReLU}}(\sum_i x_j^{l-1} k_{ij}^l + b_j^l) \quad (1)$$

where x_j^l is the j th feature map generated in the l th layer, f_{ReLU} is the activation function, k_{ij}^l are the convolutional kernels for the i th to j th feature maps, and b_j^l is the bias term.

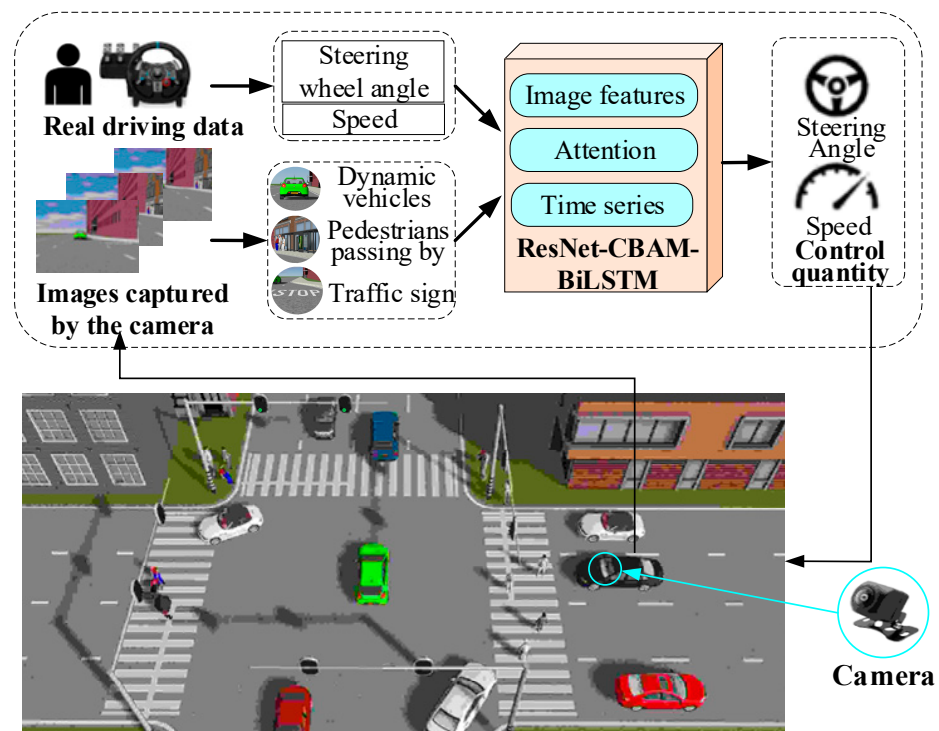


Figure 1. Scheme of the end-to-end autonomous decision-making.

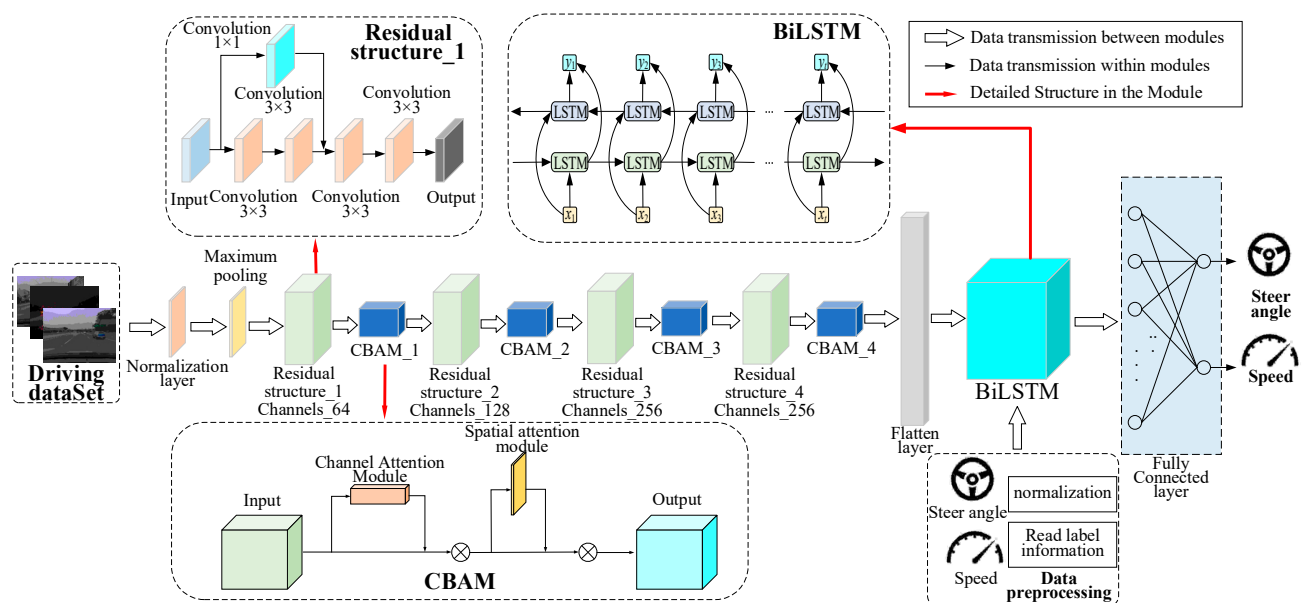


Figure 2. The structure of the ResNet-CBAM-BiLSTM model.

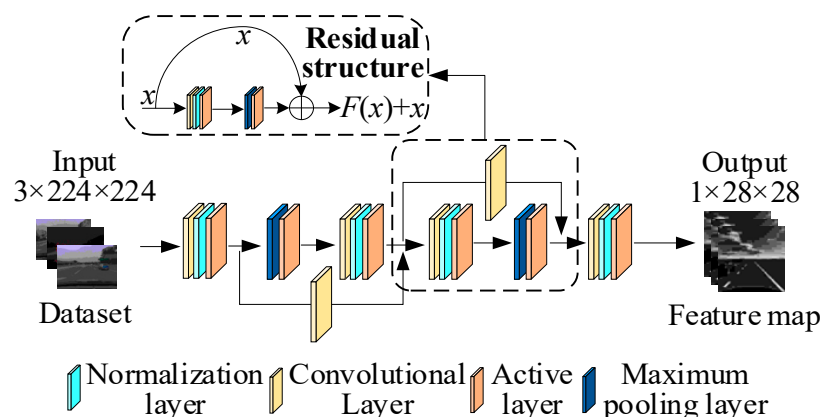


Figure 3. Architecture of ResNet.

Normalization layer, as a commonly used regularization method in deep learning, is employed to stabilize and accelerate the training process of neural network models. It also helps reduce the probability of overfitting and further improves the generalization capacity of the model. The normalization process is as follows:

$$y_i = \alpha \frac{x_i - \frac{\sum_{i=1}^n x_i}{n}}{\sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \frac{\sum_{i=1}^n x_i}{n})^2 + \beta}} + \gamma \quad (2)$$

where α is the scaling term in the normalization layer, γ is the bias term in the normalization layer, x_i is the i th input value in a training batch, y_i is the i th output value in a training batch, n is the sampling number, and β is the constant used to maintain the stability of normalization calculations.

The activation layer performs nonlinear transformations on the extracted features, enabling the neural network to capture complex and meaningful representations. The ReLU activation function is defined as follows:

$$f_{\text{ReLU}}(x) = \max(0, x) \quad (3)$$

where x is the input feature.

The pooling layer can select and filter spatial local features and further reduce the size of feature images. The pooling operation is as follows:

$$\begin{cases} h = \frac{h_x - h_k}{s} + 1 \\ w = \frac{w_x - w_k}{s} + 1 \end{cases} \quad (4)$$

where h and w are the height and width of the output feature map after the pooling operation, respectively; h_k and w_k are the height and width of the input feature map, respectively.

Taking the workload of dataset processing and experimental conditions into account, ResNet18 is adopted in this study as the primary network for image feature extraction. Thus, the extraction accuracy and speed of network training can be ensured, while the overfitting risk can be reduced, as is shown in Figure 3.

The total number of convolutional and fully connected layers in the network is 18, and the input image size is set to 224×224 . The symbol “/2” denotes that the number of convolutional channels is halved. During the identity-mapping process, the input is required to be forwarded directly across two convolutional layers. However, in certain

cases, the number of channels in the input and output feature maps may differ (as indicated by the dashed line in Figure 4), making the addition operation infeasible. To address this, an additional convolutional layer is introduced within the identity-mapping path to align the channel dimensions of the input and output feature maps.

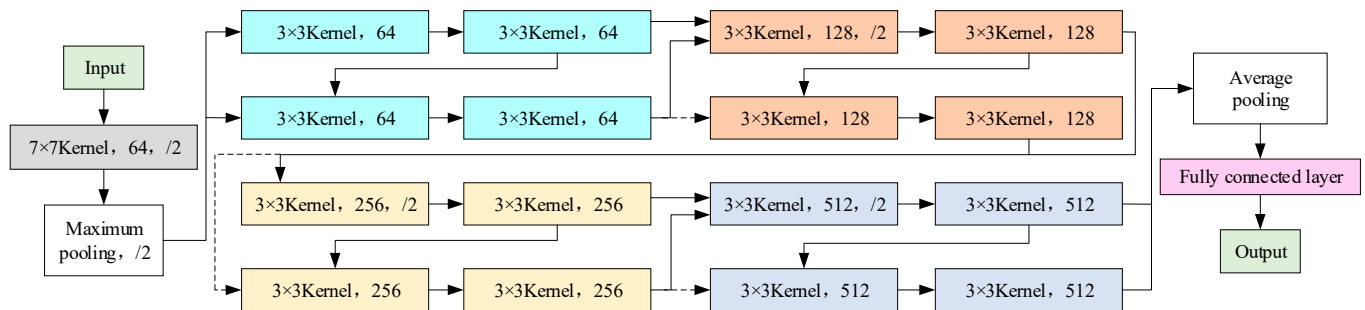


Figure 4. Structure of ResNet18.

2.2.2. Weight Calculation Module with CBAM

When intelligent vehicles drive in special weather conditions, such as snowy environments, a large number of snow features appear in the images captured by the camera. These irrelevant features may affect the decision-making values of the model. To this end, a lightweight attention mechanism module, CBAM, is introduced. Its structure is shown in Figure 5. By combining both channel and spatial attention mechanisms, CBAM enhances the representation of critical regions in the image, thereby improving the accuracy of decision-making [28,29].

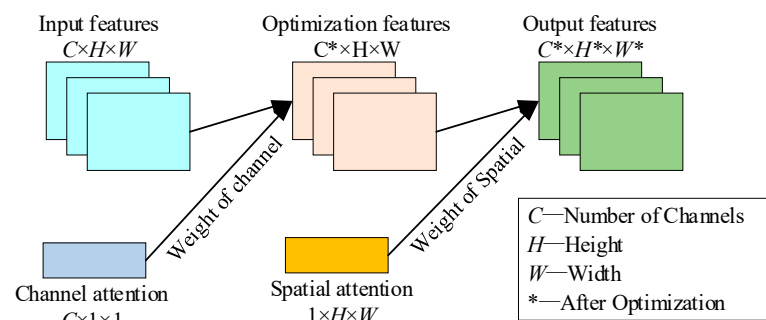


Figure 5. Modular structure of CBAM.

In this study, the steering wheel angle and vehicle speed of intelligent vehicles are classified into two decision-making tasks. To enhance the classification accuracy of the model, a channel attention mechanism is introduced. The residual convolutional neural network incorporating a channel attention mechanism is shown in Figure 6.

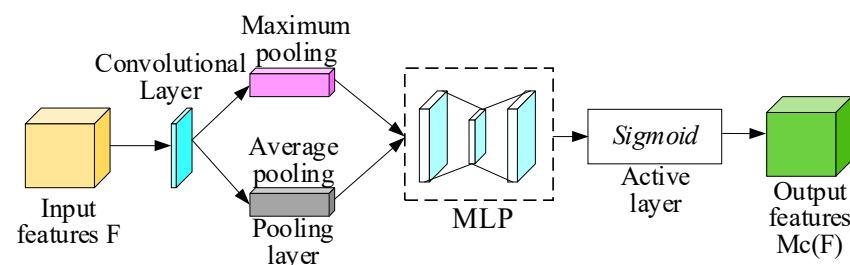


Figure 6. Mechanism of channel attention.

To avoid the loss of image feature information during pooling operations, both max pooling and average pooling operations are used. An MLP is employed to connect the fully connected layers, and the Sigmoid activation function is used to obtain the weight matrix of the feature map channel dimension, achieving the extraction of important features in the channel dimension. The output incorporating the channel attention mechanism is given as follows:

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \quad (5)$$

where F is the input feature map, $AvgPool$ is an average pooling operation, $MaxPool$ is the maximum pooling operation, and σ is Sigmoid activation function that can be described as the following formula:

$$y = \frac{1}{1 + e^{-x}} \quad (6)$$

where x is the input of the Sigmoid function, and y is the output of the Sigmoid function.

The structure of the spatial attention mechanism is shown in Figure 7. In this module, max pooling and average pooling operations are first applied to obtain two feature maps, which are then concatenated into a single feature map. The spatial attention mechanism can focus on different spatial positions of the feature map and weight the features by learning the importance of each position, making the network pay more attention to key features in the image.

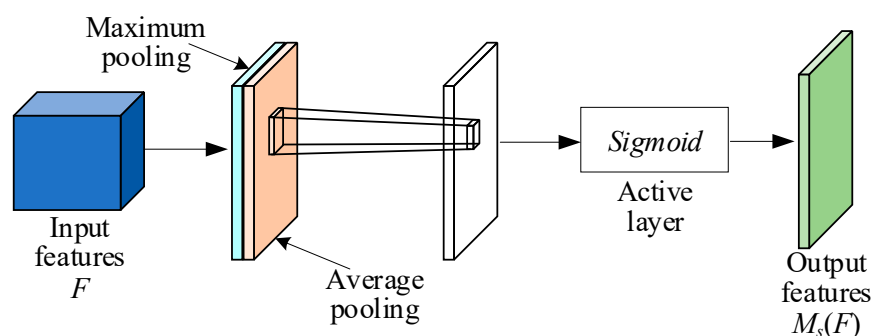


Figure 7. Mechanism of spatial attention.

The output with the spatial attention mechanism is as follows:

$$M_s(F) = \sigma(f^{7 \times 7}([AvgPool(F); MaxPool(F)])) \quad (7)$$

where $f^{7 \times 7}$ is the convolution operation, and the size of the convolution kernel is 7×7 .

2.2.3. Temporal Feature Processing Module with BiLSTM

Since vehicle behavior consists of a series of continuous states related to time, decision-making should not only rely on spatial feature extraction from images but also incorporate the temporal characteristics of motion states. BiLSTM, which integrates both forward and backward LSTM networks, possesses strong capabilities in memory retention and modeling temporal dependencies [30,31]. Its unique gating mechanism also helps to reduce the impact of gradient vanishing or exploding in classification tasks. The structure of traditional LSTM is shown in Figure 8, where f_t is the forget gate, m_t is the input gate, O_t is the output gate, h_t is the output of the hidden state of this unit, C_t is the feature information stored in this unit, $\tilde{C}_t$ is the feature information at the current time, x_t is the input at the current time, $\otimes$ represents the multiplication operation of matrix elements, and $\oplus$ represents the addition operation of matrix elements.

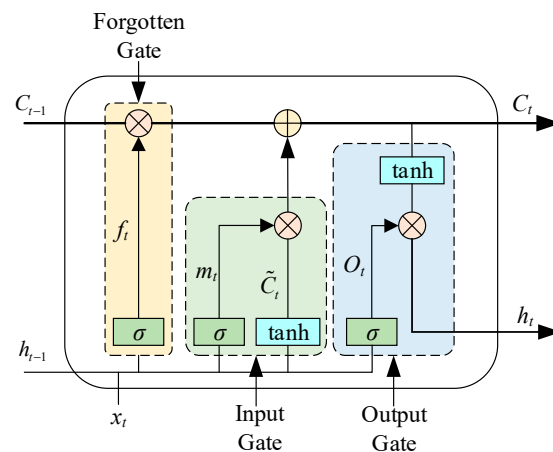


Figure 8. Structure of traditional LSTM.

The forget gate determines which information and how much information to forget from the unit state. Firstly, x_t is weighted and biased against h_{t-1} from the previous moment. The information in the current moment is passed through the activation function Sigmoid and multiplied by the unit state C_{t-1} in the previous moment, element by element. When the value of f_t approaches 1, the information from the previous moment is retained, and when it approaches 0, historical information is forgotten. The specific process is as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (8)$$

$$C_f = C_{t-1} \times f_t \quad (9)$$

where W_f is the weight matrix corresponding to the forget gate, b_f is the bias corresponding to the forget gate, and C_f is the information updated behind the forget gate.

The input gate consists of two functional units: the first uses the tanh activation function to determine changes in unit information; the second uses the Sigmoid activation function to generate m_t in the range of $[0, 1]$, which is used to control the amplitude of the current time information $\tilde{C}_t$ superimposed on the unit state. The larger the value, the higher the amplitude, indicating that the information is more important. Finally, the input gate calculates the result and adds the unit state processed by the forget gate element by element, to complete the feature information update. The specific update process is as follows:

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (10)$$

$$m_t = \sigma(W_m \cdot [h_{t-1}, x_t] + b_m) \quad (11)$$

$$C_t = C_f + \tilde{C}_t m_t \quad (12)$$

where W_c and W_m are the weight matrices corresponding to the input gates, b_c and b_m are the biases corresponding to the input gates, C_t is the updated information after entering the portal. The calculation formula for the tanh activation function is:

$$y = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (13)$$

The output gate can determine the value of the next hidden state. If LSTM has multiple layers, the gate output will serve as the input for the next layer, represented as:

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (14)$$

$$h_t = O_t \tanh(C_t) \quad (15)$$

where W_o is the weight matrix corresponding to the output gate, and b_o is the bias corresponding to the output gate.

BiLSTM is a network layer formed by combining traditional LSTM with reverse LSTM, and its structure is shown in Figure 9.

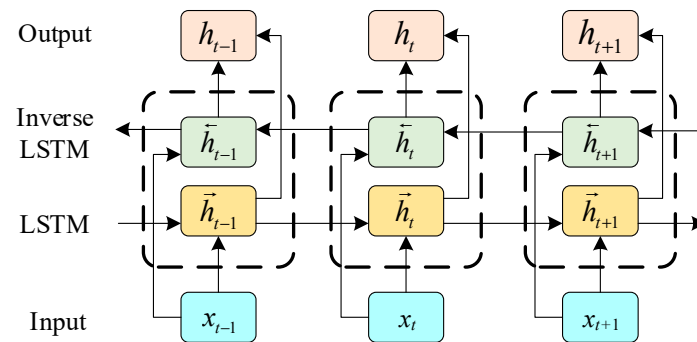


Figure 9. Network scheme of BiLSTM.

BiLSTM not only releases the temporal decision model from the constraint of prediction outputs solely based on historical information but also enables a better exploration of the complex relationships among the current, past, and future driving states of intelligent vehicles during the training process. It has better decision-making effects on steering wheel angle and vehicle speed with temporal characteristics.

2.3. Dataset and Model Training

2.3.1. Dataset

This study adopts the autonomous driving dataset Comma2k19 [32] to train and test the proposed method, since the dataset is a representative dataset designed for multi-sensor fusion and algorithm validation in autonomous driving. It was collected along a fixed 20 km segment of California State Route 280 between San Jose and San Francisco, accumulating over 33 h of data divided into 2019 one-minute segments. The data collection utilized two mainstream passenger vehicles, the Toyota RAV4 and Honda Civic, and the kinematic and dynamic characteristics of these two kinds of vehicles are highly similar. Particularly, the similarity in their steering mechanisms results in very close steering gear ratios between the Toyota RAV4 and the Honda Civic, which ensures that the mapping from steering wheel angle to front wheel angle is consistent.

The dataset fully reflects the motion characteristics of traffic flow in highway and urban commuting scenarios. The vehicle speed ranges from 0.0 to 34.0 m/s (0 to 122.4 km/h), exhibiting an overall right-skewed normal distribution. The mean speed is concentrated between 22 and 26 m/s (79–94 km/h), and the standard deviation is 4–6 m/s (14–22 km/h), which aligns with highway speed limits and commuting traffic characteristics. The Operational Design Domain (ODD) of this dataset focuses on bidirectional multi-lane scenarios. Road elements include structures such as straight sections, gentle curves, at-grade intersections, and traffic signals. Environmental conditions cover daytime, nighttime, and adverse weather such as snow and fog. The traffic flow is dominated by medium-to-high density commuting traffic, covering typical highway driving behaviors such as car following, lane changing, ramp merging/diverging, and congestion start-stop maneuvers. An example of data annotation for the Comma2k19 dataset is shown in Table 1.

Table 1. Annotated examples of the dataset.

Serial Number	Time Stamp	Speed (m/s)	Steering Wheel Angle (Deg)
1	311.2557	9.795	−1.8
2	311.2745	9.799	−1.7
3	311.2799	9.807	−1.6
4	105,601.5536	27.1278	−12.5
5	105,601.5645	27.1368	−12.4
6	105,601.5701	27.1257	−12.3

Based on the driving videos recorded in various environments from the Comma2k19 dataset, such as urban areas and highways in California, this study further filtered and categorized four typical verification scenarios: sunny daytime, nighttime, foggy, and snowy conditions. The image data for each scenario was directly sourced from the real driving records of the dataset. The specific data volume and partitioning ratios are as follows: 4800 images for sunny daytime environments (480 in the test set), 3840 images for nighttime environments (384 in the test set), 1440 images for foggy environments (144 in the test set), and 1210 images for snowy environments (121 in the test set). The selected scenarios cover both ideal driving conditions and severe weather conditions, enabling a comprehensive validation of the model's adaptability in practical applications. In the sunny daytime and nighttime environments, the vehicles traversed both urban and highway scenarios. In the foggy environment, the dataset includes only highway scenario, whereas in the snowy environment, the dataset includes only urban scenarios.

In summary, the dataset selected in this study covers diverse environments including daytime, nighttime, snowy, and foggy conditions, and includes both urban and highway scenarios. This satisfies the training and testing requirements of end-to-end autonomous decision-making models for vehicle driving scenarios and further enhances the generalization capability of the model.

2.3.2. Model Training

This study aims to use spatial features and continuous time series information perceived by onboard cameras to make decisions and plan for vehicle control variables. The model is trained through supervised learning, and the network weight parameters are continuously adjusted during the process to minimize the loss function between the model output variables and the corresponding real driving data. The specific loss function is defined as follows:

$$L = \frac{1}{N} \sum_{t=1}^N (v_{\text{decision}}^t - v_{\text{real}}^t)^2 + \frac{1}{N} \sum_{t=1}^N (\delta_{\text{decision}}^t - \delta_{\text{real}}^t)^2 \quad (16)$$

where $\delta_{\text{decision}}^t$ is the decision value of steering wheel angle at time t , δ_{real}^t is the actual steering wheel angle value at time t , v_{decision}^t is the decision value of the vehicle speed at time t , and v_{real}^t is the actual vehicle speed value at time t .

The images and corresponding real driving data should be preprocessed before being fed into the model. Crop the image to 224 pixels high by 224 pixels wide to meet the input layer requirements of the model. Normalizing the steering wheel angle and vehicle speed data can eliminate the influence of dimensionality and avoid certain features dominating the learning process of the model due to large numerical values. Normalized value $X_{t, \text{norm}}$

of the steering wheel angle or vehicle speed corresponding to the image sequence t is as follows:

$$X_{t,\text{norm}} = \frac{X_t - X_{\min}}{X_{\max} - X_{\min}} \quad (17)$$

where X_t is the raw data of steering wheel angle or vehicle speed corresponding to the image sequence t , $X_{\min}$ is the minimum data of steering wheel angle or vehicle speed in the dataset, and $X_{\max}$ is the maximum data of steering wheel angle or vehicle speed in the dataset.

The collected dataset was randomly divided into training and testing sets at a ratio of 9:1. After the random partitioning, the training set comprised 10,161 images and their corresponding data. During the training process, the learning rate for backpropagation was set to 0.0001, the batch size was set to 32, and a total of 100 iterations of forward and backward propagation over the entire dataset were performed. The training error curves of the proposed method and related comparison methods are presented in Figure 10.

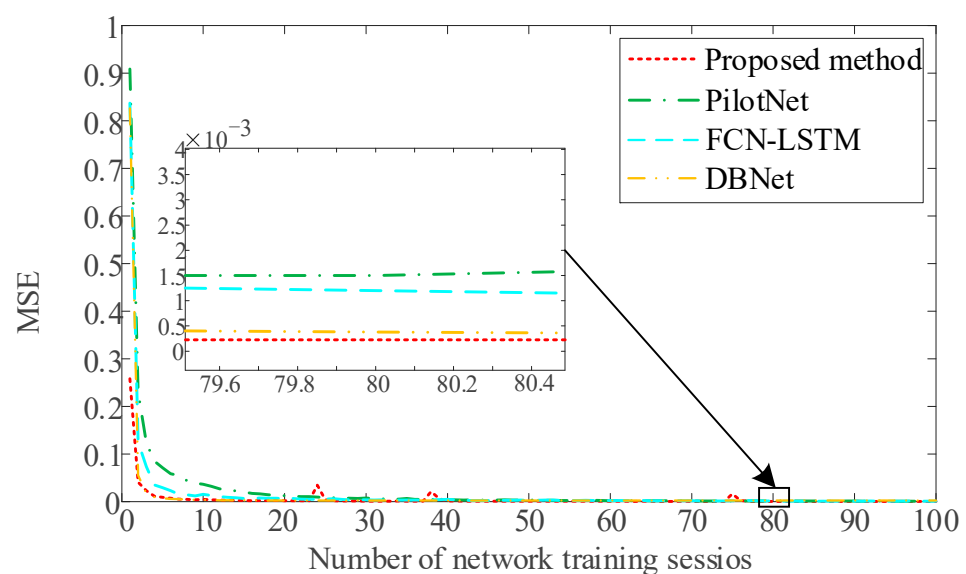


Figure 10. Result of model training.

From the graph, it can be observed that the errors of various models decrease continuously with the increase in training time. Among them, the model developed in this study stabilizes after approximately 80 training iterations, achieving a training error of 0.0003. By contrast, the PilotNet model has a training error of 0.0015, the FCN-LSTM model has a training error of 0.0012, and the DBNet model has a training error of 0.0005, all of which are greater than the model error in this study, which preliminarily verifies the feasibility and advantages of the proposed method.

3. Verification and Analysis in Multiple Scenarios

To further validate the effectiveness of the proposed end-to-end autonomous decision-making method for intelligent vehicles based on ResNet-CBAM-BiLSTM, multiple scenarios with significantly different features are selected to test the proposed method and compared with existing PilotNet models [14], FCN-LSTM models [20], and DBNet models [33] to fully verify the performance of the proposed method. This study utilizes both MSE and accuracy to quantitatively evaluate model performance. MSE can amplify the impact of large deviations by squaring the errors, thus the obvious deviation can be characterized. In addition, accuracy provides a straightforward reflection of the average precision between

the predicted and truth values. Therefore, the overall decision-making accuracy and the robustness to large local errors are simultaneously quantified with these two indicators.

3.1. Test Results of Clear Daytime Environment

The daytime clear environment dataset consists of 4800 images. After the dataset was divided into training and test set, the test set contained 480 images. Some of the test set images are shown in Figure 11a. The end-to-end autonomous decision-making method based on ResNet-CBAM-BiLSTM (proposed in this study) is compared with the autonomous decision-making methods based on PilotNet, FCN-LSTM, and DBNet (compared methods) in terms of difference between model outputs and real driving data, as shown in Figure 11b,c. Test results show that both the proposed and comparison methods can determine the steering wheel angle control quantity based on the test set images, and the main trend of decision-making values is the same as that of real driving data. However, the decision-making values generated by the proposed method exhibit higher consistency with the real driving data, while the values of comparative methods have deviations in multiple places. As for the vehicle speed, both the proposed method and comparative methods can achieve feasible decision results, but the decision-making value of the proposed method is significantly closer to real driving data, indicating that the decision-making effect is better than related comparative methods.

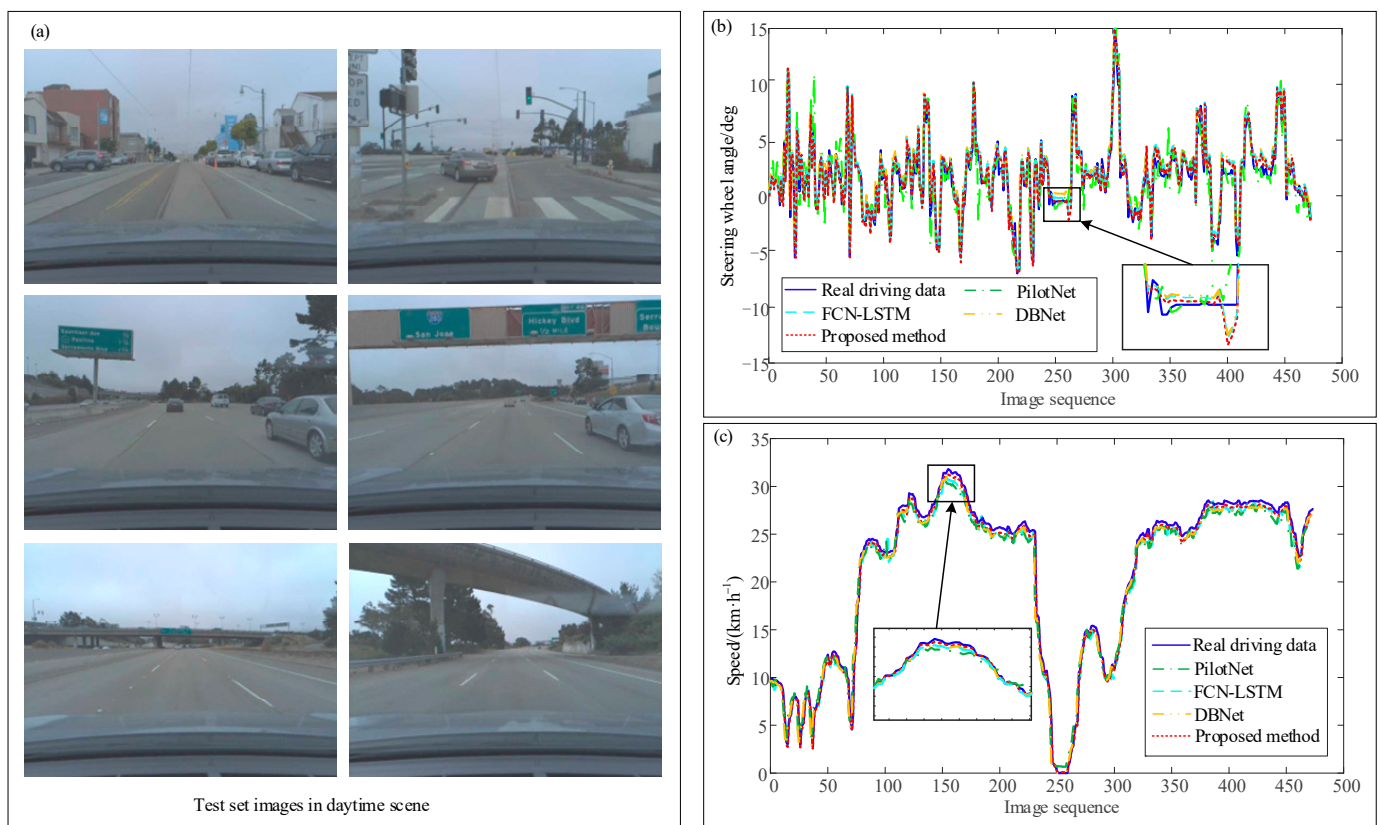


Figure 11. Test results of the daytime environment: (a) images from the test set; (b) steering wheel angle; (c) speed.

3.2. Test Results of Nighttime Environment

The dataset of the nighttime environment consists of 3840 images. After the training and testing sets are divided, the number of testing sets is 384, and some of the test set images are shown in Figure 12a. The nighttime environment is characterized by poor lighting conditions, and the significance level of road and traffic signal features decreases obviously

compared to daytime. This is a necessary testing scenario to verify the generalization and robustness of end-to-end autonomous driving methods. The output of the proposed method and comparative methods, as well as real driving data, are shown in Figure 12b,c. The experimental results show that the decision-making value of the proposed method has good consistency with real driving data. And due to the poor visibility of road features at night, such as lane markings, etc., the decision-making and value of comparison methods deviates from real driving data significantly. As for vehicle speed, the decision-making value of the proposed method is distinctly closer to real driving data, indicating that the decision-making effect is better than comparative methods.

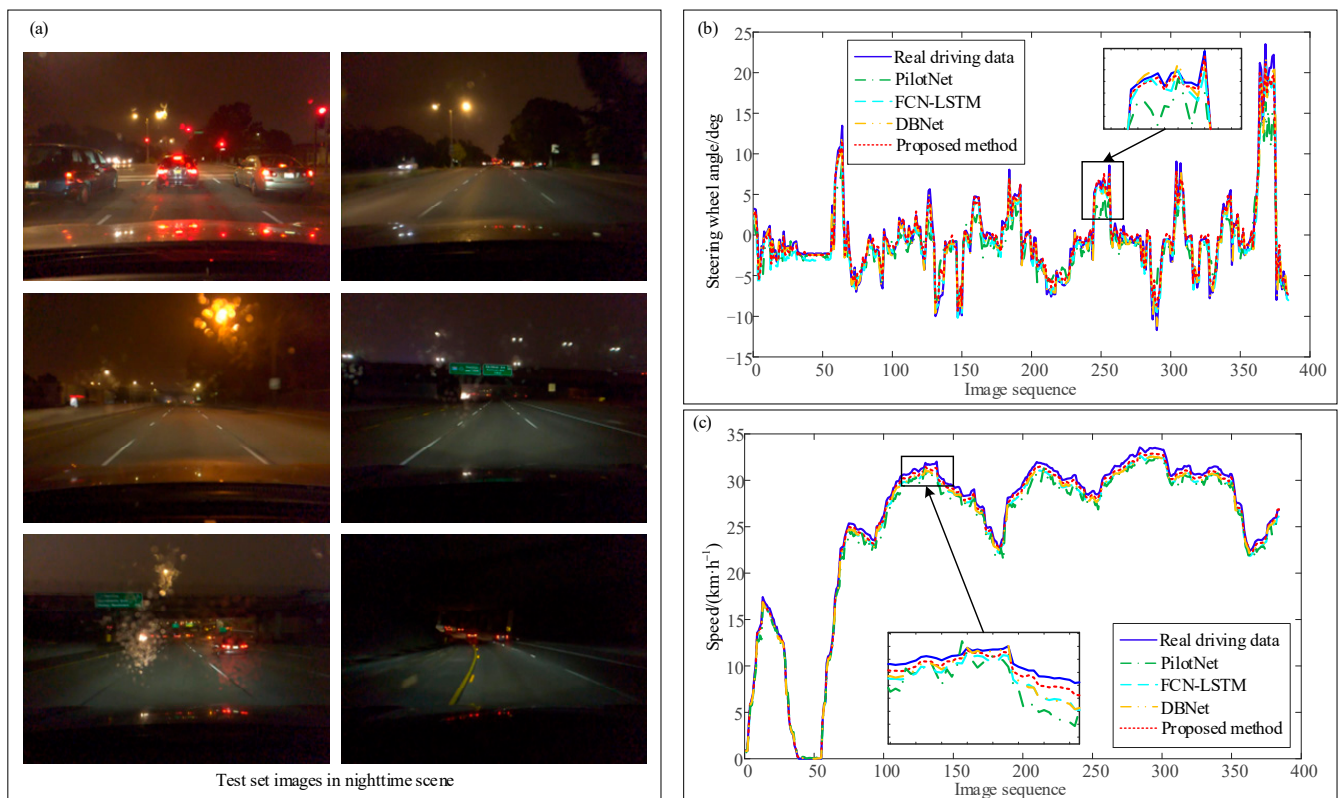


Figure 12. Test results of the nighttime environment: (a) images from the test set; (b) steering wheel angle; (c) speed.

3.3. Test Results of Fog Environment

The foggy environment dataset consists of 1440 images. After the training and testing sets are divided, the number of testing sets is 144, and some of the testing set images are shown in Figure 13a. The unclear features of long-distance road and traffic signal signs in foggy environments are an important testing scenario for verifying the generalization and robustness of end-to-end autonomous driving methods. The outputs of the proposed method and comparison methods, as well as real driving data, are shown in Figure 13b,c. The experimental results show that the steering wheel angle and vehicle speed as determined by the proposed method show high consistency with the real driving data and the curve changes smoothly, while the decision-making values of comparative methods show up and down shaking to some extent, indicating that the proposed method has better stability.

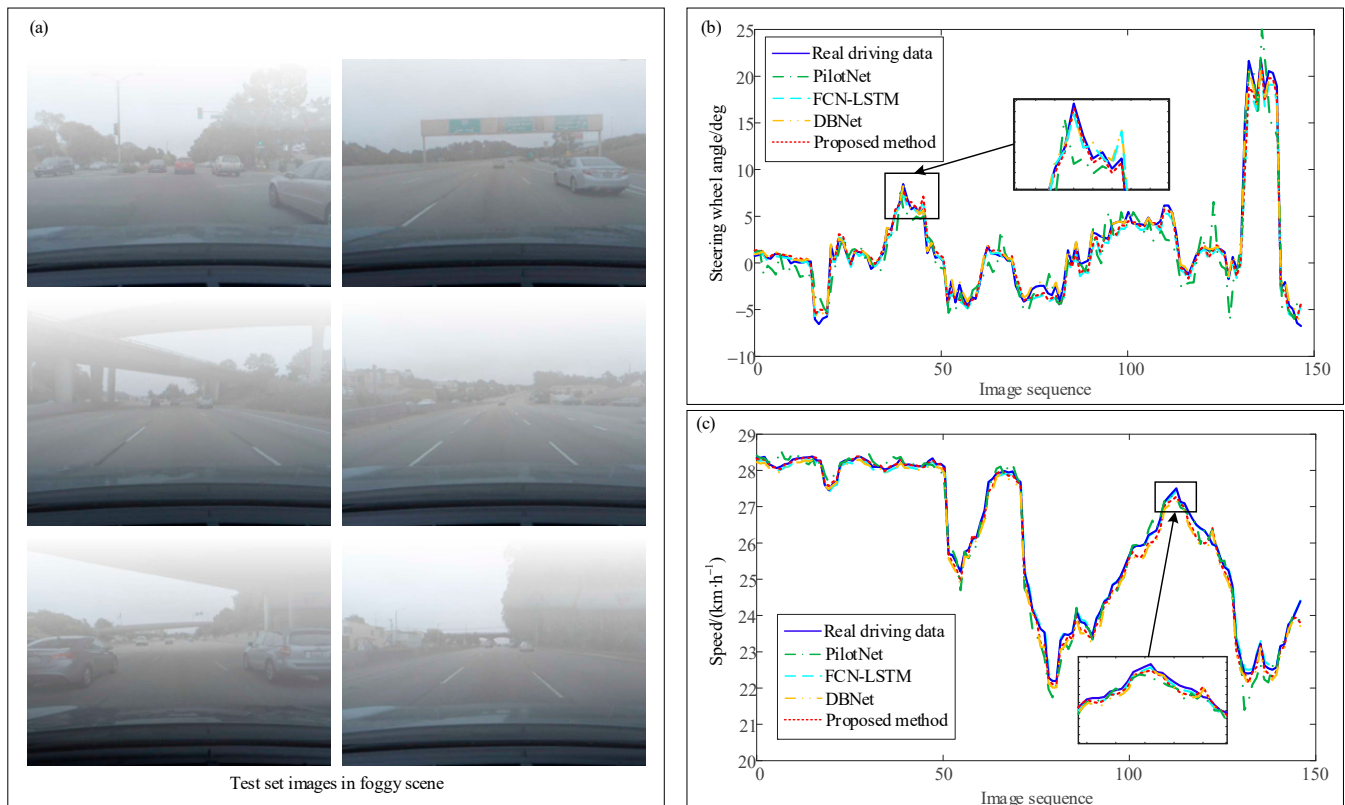


Figure 13. Test results of the foggy environment: (a) images from the test set; (b) steering wheel angle; (c) speed.

3.4. Test Results of Snowy Environment

The snowy environment dataset consists of 1210 images. After the training and testing sets are divided, the number of testing sets is 121, and some of the testing set images are shown in Figure 14a. Some road features can easily be obscured by snow accumulation in the snowy environment, and the decision-making results can easily be affected by the surrounding snow environment, making it an important testing scenario for verifying the generalization and robustness of end-to-end autonomous driving methods. The outputs of the proposed method and comparison methods, as well as real driving data, are shown in Figure 14b,c. The experimental results show that the steering wheel angle and vehicle speed as determined by the proposed method show higher consistency with the real driving data, and the curve changes are observed to be relatively stable. Moreover, due to the masking of road features such as lane lines in some snowy scenes, significant deviations from real driving data are exhibited by the decision-making values of comparison methods without an attention mechanism.

3.5. Quantitative Analysis of Comparative Test Results

To compare the performance of the proposed method and comparison methods in different testing scenarios more clearly, the quantitative data of testing error and decision accuracy from the proposed method and three comparison methods in different scenarios are listed in Table 2. It can be seen that under the same dataset and training period, different test results are exhibited by these methods. In the daytime testing scenario, the decision-making performance of related methods are relatively ideal. The test result error of 0.0006 is obtained by the proposed method, which is the lowest among related methods, and an accuracy of 90.16% in decision-making is achieved by the proposed method, which is the highest among related methods. Under nighttime and special weather conditions, there are

significant differences in results with related methods. Among them, the proposed method shows an obvious advantage in terms of testing result error and decision accuracy, where the maximum testing error is 0.0019 and the minimum decision accuracy is 81.29%. Next is the DBNet model and the FCN-LSTM model, which have a relatively small difference in decision performance. However, in snowy environments, the snow accumulation in the image has a major impact on the feature extraction of depth images, resulting in a significant decrease in the accuracy of DBNet's decision-making, namely only 69.84%. Finally, slightly higher decision accuracy than DBNet in the snowy scenario is obtained by the PilotNet model, but significantly lower accuracy than DBNet and FCN-LSTM in other scenarios is exhibited by PilotNet.

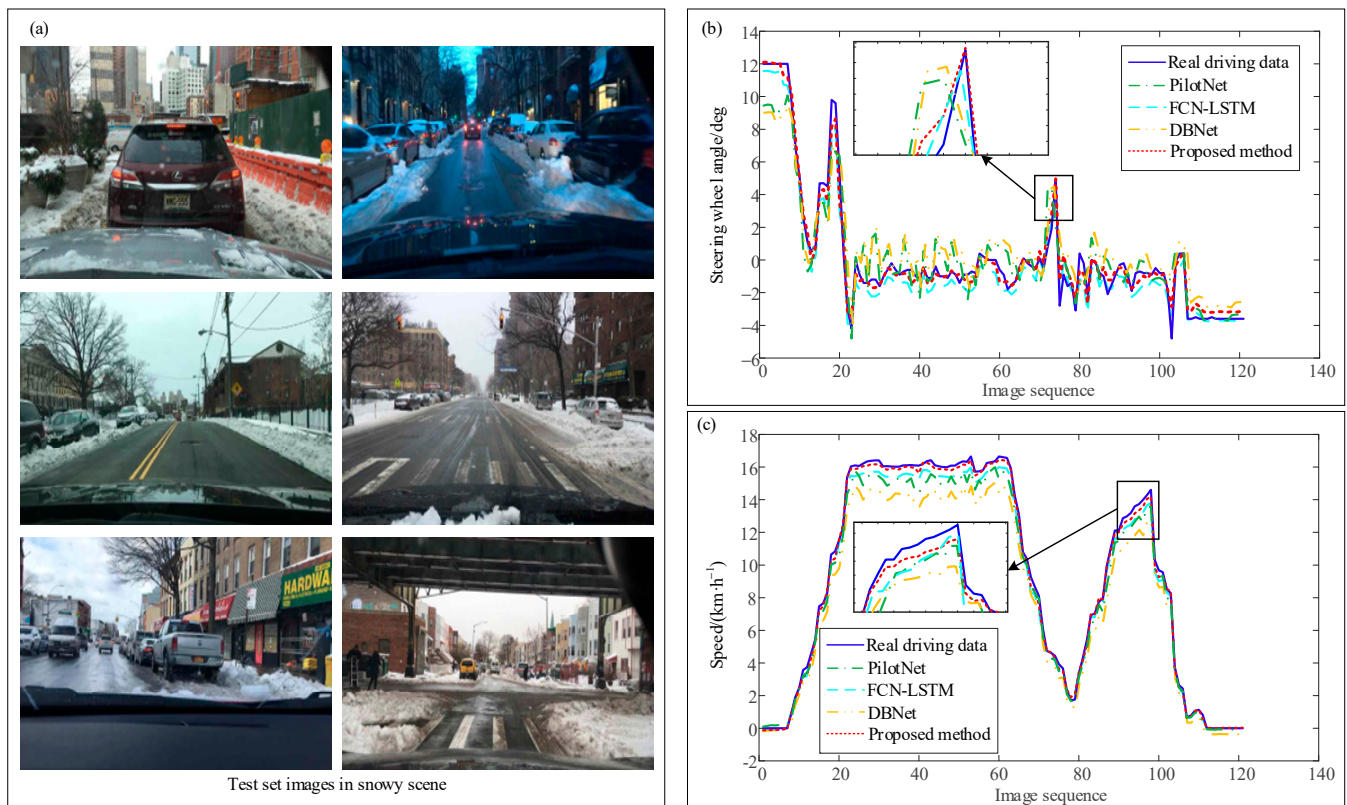


Figure 14. Test results of the snowy environment: (a) images from the test set; (b) steering wheel angle; (c) speed.

Table 2. Results of the comparative test.

Scene	Testing MSE				Accuracy/%			
	PilotNet	FCN-LSTM	DBNet	Our Model	PilotNet	FCN-LSTM	DBNet	Our Model
Daytime	0.0013	0.0009	0.0007	0.0006	88.17	88.41	90.01	90.16
Night	0.0025	0.0017	0.0014	0.0012	86.71	87.36	87.94	88.51
Foggy	0.0038	0.0021	0.0027	0.0017	79.33	81.09	79.94	84.65
Snowy	0.0054	0.0044	0.0061	0.0019	74.61	77.44	69.84	81.29

In addition, the model's complexity is also an important indicator, which, to some extent, determines its application potential, thus a balance must be struck between performance and complexity. In view of the fact that the complexity of neural network models

is mainly decided by the count of model parameters, thus the parameters count is used in this study to evaluate the corresponding complexity, as shown in Table 3. It can be found that the computational complexity of the proposed model ResNet-CBAM-BiLSTM is close to that of FCN-LSTM and is lower than that of DBNet. Since FCN-LSTM and DBNet are both existing feasible models in the field, the proposed model possesses significant potential for applications.

Table 3. Complexity of related models.

Model	Complexity	Model	Complexity
ResNet-CABM-BiLSTM	17,510,548	PilotNet	450,200
FCN-LSTM	12,489,794	DBNet	24,183,042

3.6. Analysis of Ablation Test Results

To verify the importance of different components of the model architecture for the model performance, ablation experiments are conducted on different components of the network model, and the foggy environment with unclear road features is selected as the test scenario. Ablation experiments are designed specifically for the image feature extraction module, the convolutional block attention module, and the temporal information modeling module, and the training loss and decision accuracy of related models are listed in Table 4, by which the following deductions can be drawn.

Table 4. Comparison of ablation test results.

Model	Structure of Model	Training MSE	Accuracy/%
Model 0	ResNet-CBAM-BiLSTM	0.0003	84.65
Model 1	CNN-CBAM-BiLSTM	0.0011	73.39
Model 2	ResNet-CBAM-LSTM	0.0005	81.94
Model 3	ResNet-BiLSTM	0.0031	64.90

- (1) Compared with the other three types of ablation test models, the model constructed in this study (Model 0) has the smallest training loss and the highest decision accuracy. This demonstrates that the model effectively integrates ResNet, CBAM, and BiLSTM to utilize spatial perception and temporal information of intelligent vehicles efficiently, resulting in more accurate and reasonable outputs for steering wheel angle and vehicle speed decisions.
- (2) It can be seen that when replacing ResNet with traditional CNN (Model 1), although the training loss remains at a small value 0.0011, the testing accuracy is only 73.39%. This indicates that due to the deepening of the image extraction network and the complexity of the training set data, the traditional CNN has overfitting, which seriously affects the accuracy of decision-making; After replacing the BiLSTM module with traditional LSTM (Model 2), the training loss slightly increased and the decision accuracy slightly decreased, indicating that the BiLSTM in the model has advantages over traditional LSTM. In addition, if CBAM is removed from the proposed model (Model 3), the training loss of the model increases and the accuracy decreases, observably. This indicates that the attention mechanism in the model can extract key features of the image when the road features are not obvious, thereby the decision accuracy of the model is improved. Therefore, the rationality and effectiveness of the model proposed in this paper are further demonstrated.

4. Conclusions

This study proposes a ResNet-CBAM-BiLSTM model aimed at the end-to-end autonomous decision-making of intelligent vehicles. It can directly map camera image information to decision-making variables, i.e., steering wheel angle and vehicle speed. By fully training with real-world driving datasets, it can effectively achieve end-to-end autonomous decision-making of intelligent vehicles.

The proposed method is compared and tested with existing methods based on PilotNet, FCN-LSTM, and DBNet in multiple scenarios. The results showed that the decision-making values output by the proposed method are more consistent with real driving data, with the highest accuracy of 90.16% and the lowest accuracy of 81.29% in test scenarios.

Ablation experiments are conducted for each component of the constructed model, and the results show that ResNet, BiLSTM, and CBAM all play an important role in improving model performance. Among them, CBAM has the greatest influence on final decision-making accuracy, while BiLSTM has the least impact.

Although the proposed method has shown good decision-making capabilities in multi-scenarios, there is still a gap between it and practical applications, especially in terms of decision-making accuracy in night, snowy and foggy environments. Future research will focus on enhancing recognition and decision-making capabilities, laying a solid foundation for the practical implementation of end-to-end autonomous driving.

Author Contributions: Conceptualization: X.Z. and Y.N.; methodology: X.F.; software: X.F.; datasets extraction and processing: J.Z.; validation: X.F.; writing—original draft preparation: Y.N. and X.F.; writing—review and editing: S.W.; resources: X.Z. and Y.N.; funding acquisition: X.Z. and Y.N. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China (52402492, 52372375), the Key Research and Development Program of Shaanxi (2024GX-YBXM-260), the Natural Science Basic Research Program of Shaanxi (2024JC-YBQN-0564).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zheng, L.; Liu, W.; Zhai, C. A dynamic lane-changing trajectory planning algorithm for intelligent connected vehicles based on modified driving risk field Model. *Actuators* **2024**, *13*, 380. [\[CrossRef\]](#)
2. Paniego, S.; Shinohara, E.; Josemaria, C. Autonomous driving in traffic with end-to-end vision-based deep learning. *Neurocomputing* **2024**, *594*, 127874. [\[CrossRef\]](#)
3. Xiao, Y.; Codevilla, F.; Gurram, A.; Urfalioglu, O.; Antonio, M. Multimodal end-to-end autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 537–547. [\[CrossRef\]](#)
4. Li, D.; Zhao, Y.; Wang, W.; Guo, L. Localization and Mapping Based on Multi-feature and Multi-sensor Fusion. *Int. J. Automot. Technol.* **2024**, *25*, 1503–1515. [\[CrossRef\]](#)
5. Sinha, A.; Bassil, D.; Chand, S.; Viridi, N.; Dixit, V. Impact of connected automated buses in a mixed fleet scenario with connected automated cars. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 11982–11993. [\[CrossRef\]](#)
6. Gao, Y.; Wang, X.; Huang, J.; Yuan, L. Adaptive Model Predictive Control for Intelligent Vehicle Trajectory Tracking Considering Road Curvature. *Int. J. Automot. Technol.* **2024**, *25*, 1051–1064. [\[CrossRef\]](#)
7. Yuan, T.; Zhao, X.; Liu, R.; Yu, Q.; Zhu, X.; Wang, S. Driving intention recognition and speed prediction at complex urban intersections considering traffic environment. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 4470–4488. [\[CrossRef\]](#)

8. Tampuu, A.; Aidla, R.; Van, G. Lidar-as-camera for end-to-end driving. *Sensors* **2023**, *23*, 2845. [[CrossRef](#)] [[PubMed](#)]
9. Chen, Z.; Huang, X. End-to-end learning for lane keeping of self-driving cars. In Proceedings of the IEEE Intelligent Vehicles Symposium, Los Angeles, CA, USA, 11–14 June 2017; pp. 1856–1860. [[CrossRef](#)]
10. Lin, H.; Shodiq, M.; Hsieh, M. Robot path planning based on three-dimensional artificial potential field. *Eng. Appl. Artif. Intell.* **2025**, *144*, 110127. [[CrossRef](#)]
11. Kim, D.; Kwon, J.; Nam, H. End-to-end learning-based self-driving control imitating human driving. In Proceedings of the International Conference on Information and Communication Technology Convergence, Jeju Island, Republic of Korea, 20–22 October 2021; pp. 1763–1765. [[CrossRef](#)]
12. Zhao, R.; Zhang, Y.; Huang, Z.; Yin, C. End-to-end spatiotemporal attention model autonomous driving. In Proceedings of the IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference, Chongqing, China, 12–14 June 2020; pp. 2649–2653. [[CrossRef](#)]
13. Chen, S.; Wang, M.; Song, W.; Yang, Y.; Li, Y.; Fu, M. Stabilization approaches for reinforcement learning-based end-to-end autonomous driving. *IEEE Trans. Veh. Technol.* **2020**, *69*, 4740–4750. [[CrossRef](#)]
14. Bojarski, M.; Testa, D.; Dowrakowski, D.; Fiener, B.; Flepp, B.; Goyal, P.; Jackel, P.; Monfort, M.; Muller, U.; Zhang, J.; et al. End-to-end learning for self-driving cars. *arXiv* **2016**, arXiv:1604.07316. [[CrossRef](#)]
15. Bojaeski, M.; Yeres, P.; Choromanska, A.; Choromanski, K.; Firner, B.; Jackel, L.; Muller, U. Explaining how a deep neural network trained with end-to-end learning steers a car. *arXiv* **2017**. [[CrossRef](#)]
16. Alex, K.; Jeffrey, H.; David, J.; Mazur, P.; Reda, D.; Allen, J. Learning to drive in a day. In Proceedings of the International Conference on Robotics and Automation, Montreal, QC, Canada, 20–24 May 2019; pp. 8248–8254. [[CrossRef](#)]
17. Kim, C.; Lee, M.; Hwang, K.; Ha, Y. End-to-end deep learning-based autonomous driving control for high-speed environment. *J. Supercomput.* **2021**, *78*, 1961–1982. [[CrossRef](#)]
18. Lv, S.; Li, X.; Liu, J.; Shi, M.; Chen, X. End-to-end optimization of semantic communication systems: Joint source-channel-host-tasks. *IEEE Trans. Veh. Technol.* **2025**, *74*, 7580–7593. [[CrossRef](#)]
19. Peng, C.; Yang, W.; Chang, Z.; Lv, J.; Guo, J. Trajectory prediction-based migration target selection method for vehicular network services. *Pervasive Mob. Comput.* **2025**, *110*, 102062. [[CrossRef](#)]
20. Xu, H.; Gao, Y.; Yu, F.; Darrell, T. End-to-end learning of driving models from large-scale video dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3530–3538. [[CrossRef](#)]
21. Dai, S.; Li, L.; Li, Z. Modeling vehicle interactions via modified LSTM models for trajectory prediction. *IEEE Access* **2019**, *7*, 38287–38296. [[CrossRef](#)]
22. Yan, J.; Peng, Z.; Yin, H.; Wang, J.; Wang, X.; Shen, Y.; Stechele, W.; Cremers, D. Trajectory prediction for intelligent vehicles using spatial-attention mechanism. *IET Intell. Transp. Syst.* **2020**, *14*, 1855–1863. [[CrossRef](#)]
23. Mohine, S.; Bansod, B.; Bhalla, R.; Basra, A. Acoustic modality based hybrid deep 1D CNN-BiLSTM for moving vehicle classification. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 16206–16216. [[CrossRef](#)]
24. Almars, A. Attention-based Bi-LSTM models for Arabic depression classification. *Comp. Mater. Con.* **2022**, *71*, 3091–3106. [[CrossRef](#)]
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *arXiv* **2015**. [[CrossRef](#)]
26. Fu, W.; He, Q.; Liu, S.; Feng, Q.; Gao, R. Detection of railway wheel flat based on CBAM-Enhanced ResNet for imbalanced data. *IEEE Sens. J.* **2025**, *25*, 18268–18276. [[CrossRef](#)]
27. Ashish, V.; Noam, S.; Niki, P.; Niki, P.; Jakob, U.; Llion Jones Aidan, N.; Kaiser, L.; Polosikhin, I. Attention is all your need. In Proceedings of the 31st International Conference on Neural Information Processing System, Red Hook, NY, USA, 4–9 December 2017; pp. 6000–6010.
28. Lu, G.; Liu, Y.; Wang, J.; Wu, H. CNN-BiLSTM-Attention: A multi-label neural classifier for short texts with a small set of labels. *Inf. Process. Manag.* **2023**, *60*, 103320. [[CrossRef](#)]
29. Lai, Z.; Thomas, B. End-to-end learning with memory models for complex autonomous driving tasks in indoor environment. *J. Intell. Robot. Syst.* **2021**, *107*, 37. [[CrossRef](#)]
30. Gong, J.; Qu, Z.; Zhu, Z.; Xu, H. Parallel timesnet-BiLSTM model for ultra-short-term photovoltaic power forecasting using STL decomposition and auto-tuning. *Energy* **2025**, *320*, 135286. [[CrossRef](#)]
31. Gao, G.; Wang, C.; Wang, J.; Lv, Y.; Li, Q.; Ma, Y.; Zhang, X.; Li, Z.; Chen, G. CNN-Bi-LSTM: A complex environment-oriented cattle behavior classification network based on CNN and Bi-LSTM. *Sensors* **2023**, *23*, 7714. [[CrossRef](#)] [[PubMed](#)]

32. Schafer, H.; Santana, E.; Haden, A.; Biasini, R. A commute in data: The comma2k19 dataset. *arXiv* **2018**, arXiv:1812.05752. [[CrossRef](#)]
33. Liao, M.; Zou, Z.; Wan, Z.; Yao, C.; Bai, X. Real-time scene text detection with differentiable binarization. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 11474–11481. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.