

Article

# Monovision End-to-End Dual-Lane Overtaking Network without Map Assistance

Dexin Li <sup>1,2,†</sup> and Kai Li <sup>1,2,\*,†</sup>

<sup>1</sup> School of Cyberspace Security and Computer, Hebei University, Baoding 071002, China; lidexin@stumail.hbu.edu.cn

<sup>2</sup> Institute of Intelligence Image and Document Information Processing, Hebei University, Baoding 071002, China

\* Correspondence: likai@hbu.edu.cn; Tel.: +86-1363-338-0065

† These authors contributed equally to this work.

**Abstract:** Overtaking on a dual-lane road with the presence of oncoming vehicles poses a considerable challenge in the field of autonomous driving. With the assistance of high-definition maps, autonomous vehicles can plan a relatively safe trajectory for executing overtaking maneuvers. However, the creation of high-definition maps requires extensive preparation, and in rural areas where dual two-lane roads are common, there is little pre-mapping to provide high-definition maps. This paper proposes an end-to-end model called **OG-Net** (Overtaking Guide Net), which accomplishes overtaking tasks without map generation or communication with other vehicles. OG-Net initially evaluates the likelihood of a successful overtaking maneuver before executing the necessary actions. It incorporates the derived probability value with a set of simple parameters and utilizes a Gaussian differential controller to determine the subsequent vehicle movements. The Gaussian differential controller effectively adapts a fixed geometric curve to various driving scenarios. Unlike conventional autonomous driving models, this approach employs uncomplicated parameters rather than RNN-series networks to integrate contextual information for overtaking guidance. Furthermore, this research curated a new end-to-end overtaking dataset, **CarlaLanePass**, comprising first-view image sequences, overtaking success rates, and real-time vehicle status during the overtaking process. Extensive experiments conducted on diverse road scenes using the Carla platform support the validity of our model in achieving successful overtaking maneuvers.



**Citation:** Li, D.; Li, K. Monovision End-to-End Dual-Lane Overtaking Network without Map Assistance.

*Appl. Sci.* **2024**, *14*, 38. <https://doi.org/10.3390/app14010038>

Academic Editors: Mehmet Aydin, Yong Yue and Xiaohui Zhu

Received: 25 October 2023

Revised: 11 December 2023

Accepted: 12 December 2023

Published: 20 December 2023



**Copyright:** © 2023 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) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** overtaking; dual-lane; automatic drive; feasible probability

## 1. Introduction

In recent years, with the continuous advancements in hardware, the field of autonomous driving has experienced remarkable growth in both research and practical applications. Overtaking is a pervasive behavior observed during driving, typically occurring when a preceding vehicle is traveling at a slow pace. However, assessing the environment to successfully execute overtaking maneuvers, particularly in situations where two lanes face each other and oncoming vehicles are present, has received scarce research attention.

Modular autonomous driving systems utilize environmental information generated by perception and prediction modules to obtain a safe overtaking trajectory through numerical optimization, curve fitting, reinforcement learning, and other techniques [1–3]. During the overtaking process, the overtaking route may resemble a classic function curve like a sigmoid or tangent function. Huang et al. [4] proposed using a simulated sigmoid curve for overtaking path planning. The Baidu Apollo team introduced the EMPlanner algorithm [2], a planning method primarily designed for obstacle avoidance in urban driving scenarios. However, EMPlanner typically does not make autonomous overtaking decisions and is better suited for dealing with stationary obstacles or extremely slow-moving objects. The Qingzhou team has developed a spatio-temporal coordinated planning algorithm [1,5]

that can be used for overtaking in scenarios where there is a leading vehicle in the same lane or oncoming vehicles in a dual-lane environment. This method projects the predicted trajectories of surrounding vehicles or obstacles at time  $t$  onto a spatio-temporal coordinate system, allowing for the planning of the most suitable vehicle trajectory within feasible spatio-temporal regions. In the highway-env simulation platform, Brito et al. [6] and Chen et al. [7] successfully employed reinforcement learning to achieve overtaking and lane changing with relatively stable outcomes. Ghimire et al. [8] and Yuan et al. [9] utilized the DQN [10] algorithm to train their model policies and optimize the selection of the best driving strategies using teacher–student networks.

In addition to trajectory research, significant attention has been devoted to studying the driver's overtaking intentions. Rasch et al. [11] and Liu et al. [12] employed statistical approaches and designed Bayesian networks to classify overtaking intentions based on the self-information of the overtaking vehicle. Eysenbach et al. [13] established upper-bound expectations on a Markov chain and adjusted the driving behavior of vehicles to choose between conservative and aggressive overtaking methods. Furthermore, Mo et al. [14] and Hegde et al. [15] achieved overtaking through information sharing via vehicular networks. However, these studies heavily relied on high-precision maps, connected vehicles, and complex autonomous driving systems.

In contrast, end-to-end methods typically do not depend on high-resolution maps [16]. At least two of the following three components are required as inputs to directly output ideal trajectory or control signals: radar signals, image data, and information about the vehicle itself. Rhinehart et al. [17] proposed "Imitative Models" to combine the benefits of IL and goal-directed planning. Imitative Models are probabilistic predictive models of desirable behavior that are able to plan interpretable expert-like trajectories to achieve specified goals. Han et al. [18] embedded spatial and temporal attention mechanisms based on ConvLSTM and SE-Net modules into appropriate layers to focus on key visual information and long-term and short-term memory in visual sequences in order to predict the steering angle of the ego vehicle. This contributed to subsequent research on the mapping of raw inputs and control signals. Chen et al. [19] proposed the Openpilot model, capable of achieving L2-level autonomous driving functions. Perumal et al. [20] proposed an innovative uncertainty-aware framework for end-to-end control, utilizing the probabilistic control barrier function (CBF) to enforce safety constraints. They experimented on a small race car and achieved better cornering performance. The TCP model [21] developed by the Shanghai Artificial Intelligence Lab utilizes only first-person view images and vehicle information as inputs. It can guide vehicle control using generated trajectories and achieved first place in the 2022 Carla leaderboard. Currently, these end-to-end models are capable of achieving Level 2 and Level 3 autonomous driving functions. Some research has employed end-to-end models to generate data required for overtaking, such as trajectory prediction. Perumal et al. [22] developed IOAS, which provides overtaking suggestions to drivers. This system predicts the speed of the leading vehicle by combining VNet and TTC-Net networks, ensuring high accuracy. Similarly, Mandal et al. [23] utilized the tail lights of the leading vehicle and the headlights of oncoming vehicles to determine the speed information of relevant vehicles in night-time environments, providing overtaking recommendations.

Nevertheless, the majority of studies in the field concentrate on conservative driving tasks [21]. These models are trained based on correct control signals or planned trajectories as labels, but the data generated by traffic accidents is actually equally important. Including these data in model training can help improve the ability of self-driving cars to avoid dangers. Currently, there is no denying that the end-to-end model is capable of executing simple driving tasks on certain simulation platforms. However, its ability to handle more complex and risky maneuvers, such as overtaking on dual-lane roads, is still a thought-provoking issue.

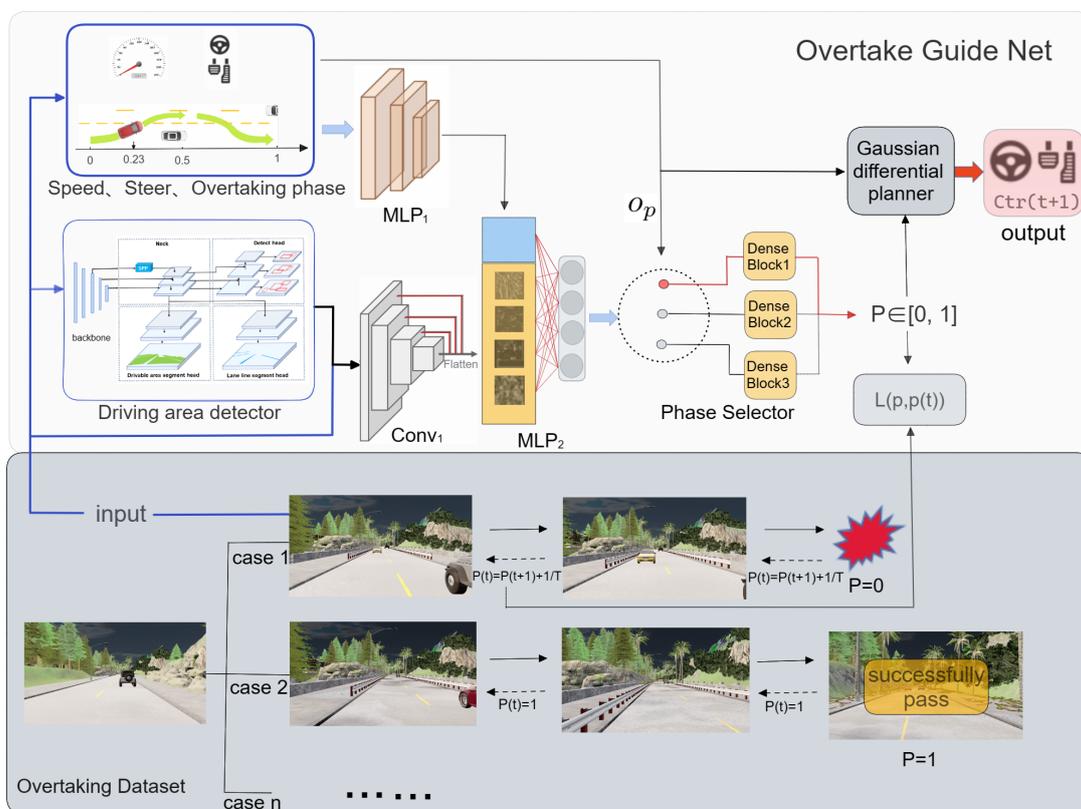
Thus, an effective end-to-end model for overtaking on roads should have the capacity to perform overtaking maneuvers when safe and quickly return to the main lane when potential hazards are detected. In light of these considerations, this paper presents a model capable of estimating the probability of safe overtaking in real time using the information

readily available from the vehicle (e.g., image, speed, and steering wheel angle). By utilizing this probability to make an overtaking decision, an overtaking control scheme is generated. In accomplishing this, our contributions are as follows:

- This paper proposes an innovative method to address the relatively perilous challenge of overtaking in autonomous driving, substantiating the viability of the end-to-end approach in tackling this task.
- This paper abandons the traditional tracking control method, and we design a control planner that can drive an overtaking curve with fixed lateral displacement at different speeds using geometric curves.
- This study established a dataset for dual-lane overtaking scenarios called CarlaLanePass, encompassing both successful and unsuccessful overtaking situations. CarlaLanePass offers a distinctive perspective that contributes to the realm of autonomous driving research.

## 2. Overtaking Guide Net

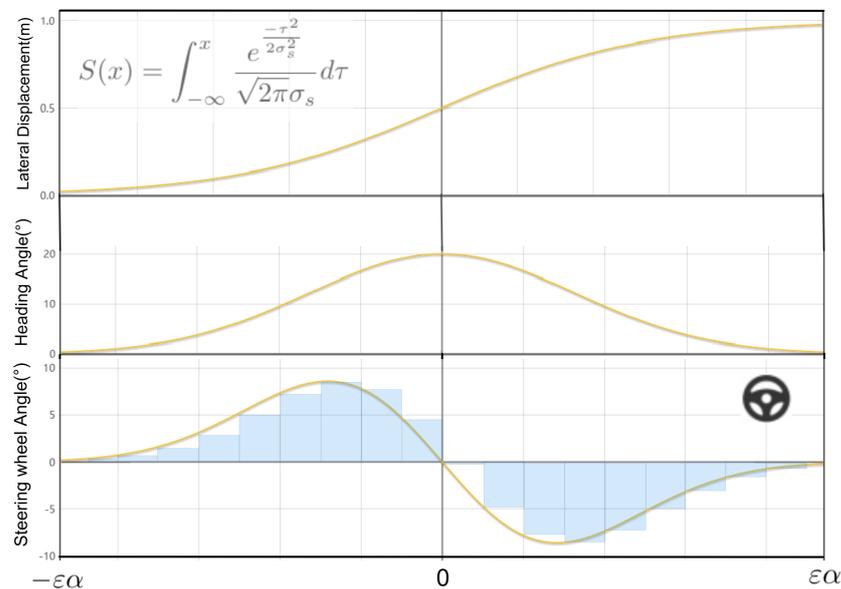
Our model aims to enable overtaking using a vision module and a set of simple and readily available parameters. The framework comprises an OFP (Overtake Feasibility Predictor) and a GDCP (Gaussian Differential Control Planner), as shown in Figure 1. Initially, we utilize the Gaussian differential controller to generate a dataset called CarlaLanePass, which includes the probability of safe overtaking in the current situation on the Carla platform. Subsequently, we train our OFP on this dataset. Finally, based on the feasible probability generated by this network and other information, our model inputs these values to the GDCP. The controller can generate the vehicle’s control signal for the next moment, thus accomplishing the overtaking maneuver or returning to the original lane.



**Figure 1.** OG–Net. The input image is processed through the network, which outputs the safety probability of continuing the overtaking maneuver under the current state. The output probability is combined with the vehicle state information and inputted into the GDCP for the next control operation.

### 2.1. Gaussian Differential Control Planner

This paper adopts the primitive function of the Gaussian distribution as the trajectory for overtaking by an AV. It takes the form of a smooth S-shaped curve, as shown in the first part of Figure 2. The first derivative of this function represents the angle between the vehicle’s body and the road in the vertical direction at the current moment, while the second derivative precisely corresponds to the steering wheel angle at the vehicle’s position. The aim of this study was to design a *GDCP* that can generate corresponding control signals by incorporating the current safety probability, ultimately achieving lane-changing overtaking. The overall model flow is illustrated in Figure 1. This paper proposes the overtaking phase parameter  $o_p$ , combined with the vehicle speed  $v$ , to replace the coordinate information input in the control planning function. When the vehicle’s speed changes, it leads to variations in the longitudinal displacement of the vehicle. To address this, a velocity adaptation coefficient  $\alpha$  and a reference speed  $v_0$  are introduced, as shown in Equation (1). The overtaking trajectory, vehicle heading angle, and steering wheel angle at  $v_0 = 20$  m/s are depicted in Figure 2.



**Figure 2.** Vehicle overtaking trajectory, heading angle, and steering wheel angle. During the entire overtaking phase, the size of the blue shaded area is the degree to which the vehicle changes its orientation every time the steering wheel controls the vehicle. The black pattern in the lower right corner of the image is the logo of the steering wheel.

As the vehicle speed increases, a higher value of  $\alpha$  results in the reduced curvature of the overtaking curve. The smoothing parameter  $\sigma_s$  of the overtaking trajectory function increases, and the stretching factor  $\sigma_s$  is defined as shown in Equation (1).

$$\sigma_s = \alpha \sigma_{s_0} = \frac{v}{v_0} \sigma_{s_0} \tag{1}$$

where  $\sigma_{s_0}$  is the smoothing parameter that can complete overtaking when  $v = v_0$ . Since the variation range of  $x$  is different when the vehicle speed is different, this paper defines a range  $[-\epsilon\alpha, \epsilon\alpha]$ , and the AV only performs overtaking operations within this range, as shown in Equations (2) and (3).

$$t' = -\epsilon\alpha + \frac{2\epsilon\alpha}{o_p} t, t \in [0, o_p] \tag{2}$$

$$S(-\epsilon\alpha) < \xi \tag{3}$$

where  $\zeta$  is a hyperparameter, which is a number close to 0. If  $\zeta$  is too large, it could result in the vehicle not making smooth turns at the beginning and end of the overtaking maneuver, and it could also amplify the yaw angle error caused by differential operations. On the other hand, if  $\zeta$  is too small, it might lead to a delay in transitioning to the opposite lane after receiving the overtaking signal, potentially causing a missed opportunity for optimal overtaking. When  $t$  is equal to  $o_p$ , the vehicle has occupied the opposite lane and complete the lane change.  $\varepsilon$  is a hyperparameter that is utilized to control the range of lateral displacement in the transition lane. After putting Equation (2) into  $S(x)$  and calculating its second-order derivative, the output value theta is the rotation angle  $s$  of the steering wheel at this moment, as shown in Equation (4).

$$\theta(t') = \frac{t'}{k\sigma_s^3} e^{-\frac{t'^2}{2\sigma_s^2}} \quad (4)$$

where  $k$  is a fixed value that can affect the maximum lateral displacement of the vehicle during overtaking and is determined through repeated experiments. Once the vehicle completes the maneuver of overtaking the target vehicle in the opposite lane, the system reverses the past direction control sequence to return to the main lane. When the output probability is low, our system gradually reduces the vehicle's speed. Due to the symmetry of the overtaking trajectory, the AV can restore the heading angle by performing the inverse operation of the recently executed direction control sequence. By performing the reverse direction control operation again, the vehicle can return to the main lane and await the next overtaking opportunity.

## 2.2. Overtake Feasibility Predictor

The OFP utilized the overtaking phase parameter  $o_p$  to determine past and future vehicle behaviors and explored the feasibility of using this parameter as a replacement for RNNs to consider contextual information. This approach allowed us to reduce the model size and improve operational efficiency. Furthermore, our model focused on outputting the probability of being able to continue overtaking in the current state, rather than generating planned waypoints and control signals. Our end-to-end model was trained using a dataset collected from Carla. The overall framework consisted of an image encoding module, a vehicle information encoding module, a phase selector, and a *GDCP*. Our optimization objectives were as follows:

$$\arg \min_{\theta} E_{(x,p) \sim D} [L(p, f_{\theta}(x))] \quad (5)$$

where  $D = (x, p)$  is a dataset containing the information  $x$  available to the vehicle and the probability  $p$  that it can safely overtake in the current state, and  $f_{\theta}(x) \in [0, 1]$ ,  $p \in [0, 1]$ . Under the premise of ensuring that the generation probability is as far below the prior probability as possible, the difference between the generation probability and the ground truth is minimized. Each piece of information is a bound group  $x_t = (i_t, s_t), t \in [0, 2o_p]$  and  $s_t = (\theta_t, v_t, o_{p_t})$ .

As depicted in Figure 1, this study used yolop [24] as the DAD (driving area detector) to focus on the features of the exercisable area. The image  $i$  that is preprocessed by the DAD undergoes feature extraction through *Conv*<sub>1</sub>, resulting in a feature vector  $j^{img}$ . The overtaking stage information  $o_p$ , along with the vehicle's heading  $s$  and speed, are input into *MLP*<sub>1</sub>, which generates a constraint feature,  $j^{state}$ . The final encoded feature  $F$  is formed by the fusion of these vectors, shared among the next stages:

$$F_t = MLP_2(Concat[j_t^{img}, j_t^{state}]) \quad (6)$$

The structure of the neural network is shown in Table 1. The safety of overtaking maneuvers may be substantial when the autonomous vehicle (AV) is still in the original lane and there are preceding vehicles. However, if there is a vehicle not far ahead af-

ter entering the opposite lane, the likelihood of a successful overtaking becomes almost zero. To mitigate the influence of similar image features on decision making, this study introduced the Phase Selector (PS) in the feature analysis stage after feature fusion. The selection of network branches is determined by incorporating the current overtaking phase parameter, guided by  $o_p$ , to generate the probability of safe overtaking in the current phase, denoted as  $p_t$ :

$$p_t = \text{Sigmoid}[PS(o_p, F_t)] \quad (7)$$

$$Ctr(t+1) = \begin{cases} \theta(t+1), & \frac{1}{n} \sum_{i=t-n}^t p_t \geq P \\ -\theta(t), & \frac{1}{n} \sum_{i=t-n}^t p_t < P \end{cases} \quad (8)$$

where  $Ctr(t+1)$  represents the control signal to be executed by the vehicle in the next moment, and  $P$  and  $n$  are hyperparameters designed to address issues related to instability in the output probabilities of the probability generation module. When there is noise interference, the decision probability obtained from a certain piece of data may be lower than this threshold. If a vehicle that is overtaking in a safe environment conservatively returns to the main lane based on these data, the network will be too sensitive and it will be difficult for the vehicle to complete an overtaking maneuver. If  $n$  is too small, the error probabilities caused by recent noise in the received data could significantly impact the accuracy of current decisions, resulting in oscillations between overtaking and non-overtaking determinations. Conversely, if “ $n$ ” is too large, the probabilities generated from past time instances could heavily influence current decisions, potentially causing the vehicle to miss the optimal overtaking or evasion timing. When the probability average of the last  $n$  predictor outputs is greater than  $P$ , the system can continue to use the overtaking signal to overtake. On the contrary, the vehicle should execute the reverse control signal of the previous step and finally return to the main lane.

**Table 1.** Detailed network structure of our OFFP model. The  $\times$  sign in the right column indicates the number of modules in the network.

Layer Type	# of Filters	Activation Function	#
Driving area detector			
yolop			
Conv1			
ResNet-34			
MLP1			
FC	128	ReLu	$\times 2$
MLP2			
FC	512	ReLu	$\times 2$
FC	256	ReLu	$\times 1$
Dense Block1			
FC	256	ReLu	$\times 2$
FC	1	ReLu	$\times 1$
Dense Block2			
FC	256	ReLu	$\times 1$
FC	128	ReLu	$\times 1$
FC	1	ReLu	$\times 1$
Dense Block3			
FC	256	ReLu	$\times 2$
FC	1	ReLu	$\times 1$

### 2.3. Loss Design

The objective of this model was to differentiate the output probabilities as much as possible while they were relatively concentrated, maximizing the effectiveness of repairing models with significant probability differences. Initially, binary cross-entropy loss was used as the loss function of the model. The total loss of the model decreased during the convergence process, and the prediction probability was generally close to the label probability. However, it was almost always higher than the label probability. This result made overtaking decisions dangerous. Hence, this study aimed to design a penalty term that can amplify the loss value when the estimated probability is too large and decrease it slightly when the estimated probability is conservative. Therefore, the superposition of the exponential function and the traditional binary cross-entropy loss function was more in line with our wishes. In order to keep  $loss \in (0, +\infty)$ , the loss function was written as in Equation (9), where  $p_t$  represents the probabilities generated by the model and  $\hat{p}_t$  represents the ground-truth probabilities. If the generated probabilities are lower than the actual probabilities, the system tends to be conservative and may miss overtaking opportunities. However, if the generated probabilities are higher than the actual probabilities and overtaking is performed, it poses risks to the passengers. Therefore, the penalty term in the right half of Equation (9) was used to amplify the impact of the error probability on the model when  $\hat{p}_t > p_t$  and, conversely, could serve as a relaxation term to reduce the model updates when  $\hat{p}_t < p_t$  in order to reduce conservative output probabilities.

$$L(p, \hat{p}) = \sum_{t=0}^N p_t \log \hat{p}_t + (1 - p_t) \log(1 - \hat{p}_t) + e^{(\lambda(\hat{p}_t - p_t))} - 1 \quad (9)$$

### 2.4. Algorithm Process

The entire process can be divided into two main phases, training and simulation, as illustrated in Algorithm 1. Initially, CarlaLanePass was utilized to train the OFP. This network was trained iteratively in batches based on the training epochs. Data points with approximate values of  $p = 0$  and  $p = 1$  were prioritized for training, gradually incorporating data until all instances were covered. After completing the training phase, the information obtained by the ego vehicle was fed into the OFP network to generate the current probability  $p$ . Subsequently, the average of the most recent 'n' probabilities was computed to decide whether the upcoming action should involve overtaking or returning to the original lane.

---

#### Algorithm 1: OG-Net process.

---

**Data:** CarlaLanePass.  
**Result:** Output control signal  $Ctr(t + 1)$ , probability of safe passing  $p$ ;  
**Input:** Sensory input  $i$ , speed of the ego vehicle  $v$ , steering of the ego vehicle  $\theta$ , overtaking phase parameter  $o$ , critical probability  $P$ , significant degree  $n$ .  
data :=  $(i, v, \theta, o, \hat{p}) \in CarlaLanePass$ ;  
Initialize  $P, n$ ;  
Training OFP;  
**for**  $i:=1$  to epochs **do**  
    Import data excluding  $\hat{p} \in (\frac{i}{2 \times epochs}, 1 - \frac{i}{2 \times epochs})$ ;  $j^{img} \leftarrow Conv_1(DAD(i))$ ;  
     $j^{state} \leftarrow MLP_1(v, \theta, o)$ ;  
     $F \leftarrow MLP_2(Concat(j^{img}, j^{state}))$ ;  
    **switch**  $o$  **do**  
    |  $p_t \leftarrow Sigmoid[PS(o, F)]$   
    | Backprop;  
Simulation;  
**for**  $t:=1$  to  $\infty$  **do**  
     $\theta(t) \leftarrow GDGP(v, o)$ ;  
     $p_t \leftarrow OFP(i, v, \theta, o)$ ;  
    **if**  $\frac{1}{n} \sum_{i=t-n}^t p_i < P$  **then**  
    |  $Ctr(t + 1) \leftarrow (-\theta(t - 1), decelerate)$ ;  
    **else**  
    |  $Ctr(t + 1) \leftarrow (\theta(t), accelerate)$ ;

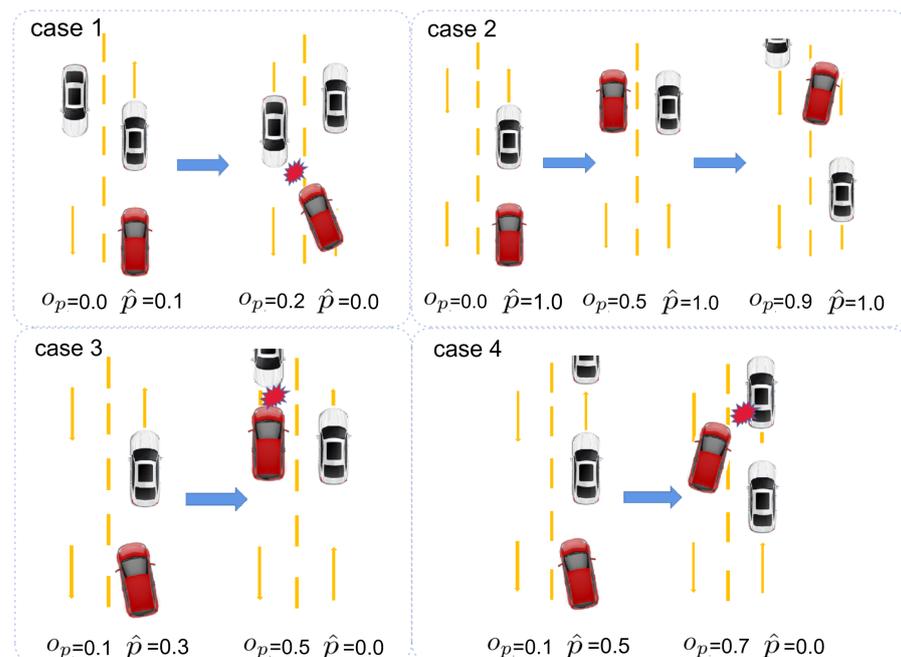
---

### 2.5. CarlaLanePass Description

To investigate the feasibility of implementing two-lane overtaking scenarios using an end-to-end model, this paper proposes a dataset CarlaLanePass specifically for overtaking on narrow two-lane roads. CarlaLanePass covers both correct and incorrect overtaking scenarios along with their corresponding safety probabilities, rather than providing waypoint or control signal coordinates. It gathers information captured from 170 distinct scenarios involving autonomous driving vehicles performing lane-changing overtaking maneuvers. We used a GDCP to perform overtaking operations in any scenario. From the beginning of the overtaking task until an accident occurred or the overtaking was successful, we collected data every 50 ms. When a vehicle in the opposite lane was very close to the ego vehicle, it would quickly collide with the oncoming vehicle when overtaking. In this case, at least about 20 pictures were collected, and the overtaking probability label of each picture was close to 0. If a collision occurred, we recorded the overtaking probability as 0 and recursively calculated the overtaking probability of the previous data entry using Equation (10). However, during the entire overtaking process, if no collision occurred, we recorded the overtaking probability of each data entry as 1, as illustrated in Figure 3. When the entire overtaking process went smoothly, a total of 240 pictures were collected. The dataset includes a total of 12,605 jpg images with dimensions of  $1280 \times 720$  pixels. Additionally, the dataset provides easily obtainable vehicle information such as speed, overtaking phase, and steering wheel angles.

$$p_t = \begin{cases} p_{t+1} + \frac{1}{T}, & \text{Overtaking failed} \\ 1, & \text{Successful overtaking} \end{cases} \quad (10)$$

where  $t \in (1, 2, \dots, T)$ . Calculating the overtaking probability in a recursive way is similar to feeding back “experience” from the future to the current moment. Sending a signal to the current moment: If one continues to perform overtaking operations at this time, what is the probability of a collision? To capture a variety of urban double-lane overtaking scenarios, we utilized the Carla simulation platform to record the first-person view images, vehicle speeds, cornering information, overtaking progress, and other relevant data for each frame in different overtaking scenarios with varying vehicle distributions and weather conditions.



**Figure 3.** Dataset description. The dataset includes complete sequences of overtaking successes or failures in most two-lane overtaking scenarios.

### 3. Simulation Experiment and Result Analysis

This paper proposes a batch training method inspired by human learning characteristics. Starting from the training data with output values close to 0 and 1, the dataset was gradually expanded, and the model was trained step by step.

#### 3.1. Training with Adaptive Data Importation

During the training process, we initially constructed a simple neural network that combined image features and vehicle information features for regression tasks. However, we encountered a challenge as the model failed to converge after 5–10 training iterations, and the output values remained concentrated around a fixed value regardless of the input situation. After conducting multiple experimental adjustments, we discovered that this issue arose due to the high similarity among the training samples. Many datasets exhibited uniformly distributed output probabilities within the range of (0, 1). Without effective pre-training, it became challenging for the model to distinguish features among the data. To address this issue, we drew on the method of adaptive image clustering and adopted an Adaptive Data Induction Training (ADIT) method inspired by human learning characteristics [25]. We started with training data that had output values close to 0 and 1, gradually expanding the dataset to train the model in a step-by-step manner, and the model gradually converged through this process. In this experiment, our model performed 100 training iterations, gradually reducing the learning rate from 0.002 to 0.00001. The training was conducted using the SGD optimizer.

#### 3.2. Predictor Test

We conducted model testing on *CarlaLanePass* to evaluate the model's output probability. The dataset primarily consists of three scenarios: safe overtaking throughout the process, collision occurring during the tentative overtaking phase, and no overtaking initially. The experimental results on the test set are shown in Figure 4. When the AV was driving in the main lane and there were no oncoming vehicles initially, the system assigned a relatively high probability for overtaking, encouraging the AV to cautiously enter the opposite lane. If an oncoming vehicle was detected while the AV was in the process of changing lanes and had not completed the overtaking maneuver, the system considered the possibility of a safe overtaking maneuver to be very low. Therefore, as the vehicle progressively advanced, the probability of successfully completing an overtaking maneuver gradually decreased until it reached zero. The probabilities output by our model should be slightly lower than the safety probabilities in the dataset to ensure the safety of autonomous driving.

#### 3.3. Comparison with Existing Methods

To explore the advantages and disadvantages of this approach in comparison to others, we conducted an evaluation according to the following criteria:

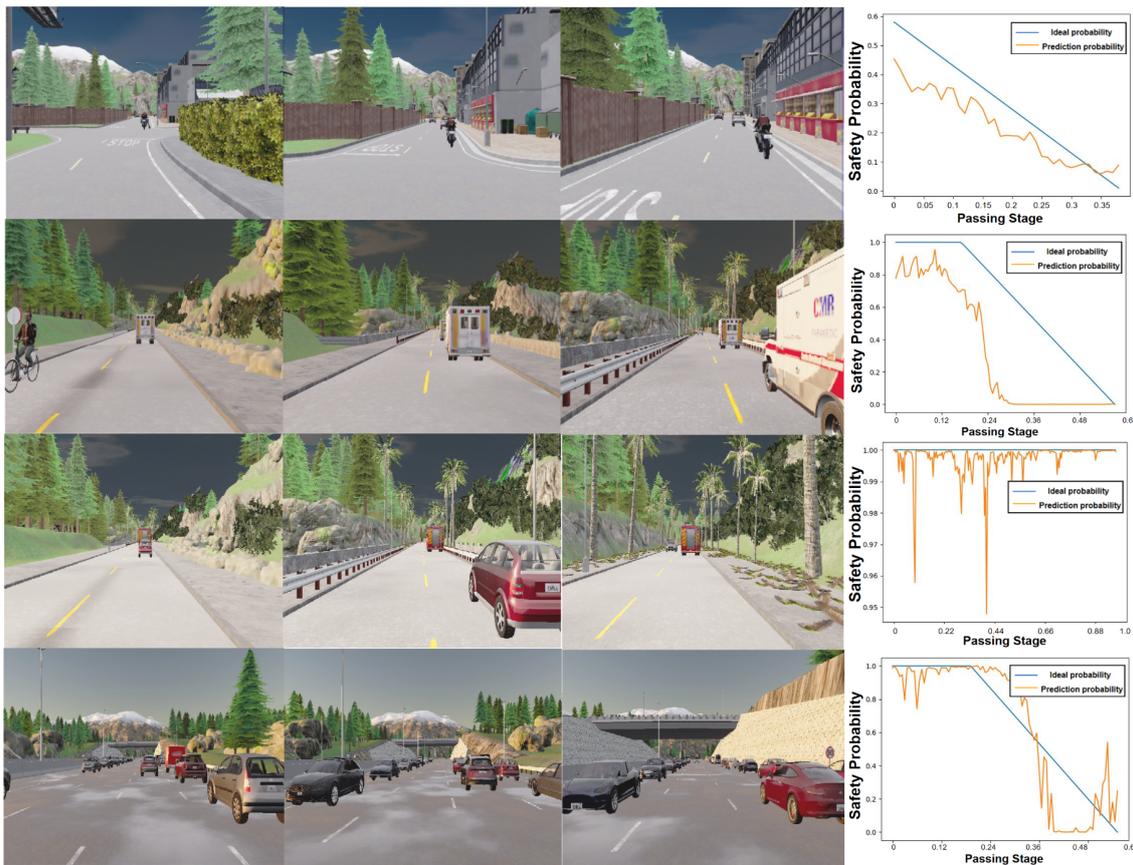
- (i) The average L2 distance between the ideal and actual positions within the first five seconds of a successful overtaking maneuver.
- (ii) Time spent in the adjacent lane  $M_t$ ;
- (iii) Number of overtaking maneuvers initiated  $M_o$ ;
- (iv) Success percentage of overtaking maneuvers  $M_s$ .
- (v) Collision probability  $P_c$ .
- (vi) Average completion time per kilometer  $T_c$ .
- (vii) Completion of selected road segments *Route Completion*.
- (viii) Trajectory smoothness and control signal smoothness.

Table 2 provides a quantitative comparative analysis of the overtaking task. We replicated several lane-changing overtaking methods with lane-changing capabilities on the Carla platform. The experiments were conducted on a dual-lane highway segment measuring 1 km in length. We collected experimental data from 20 distinct scenarios, each

involving lane-changing overtaking tasks. Various factors were considered, including traffic density, vehicle speeds, and vehicle types.

**Trajectory fitting analysis.** The experiments showed that although our method did not discretize the reference point of the path like the traditional method and then carry out control planning towards the target path point, its actual effect was not poor for the overtaking route with a single trajectory. The differences between the ideal trajectory points and the actual trajectory points were calculated within the first five seconds, commencing from the initiation of the overtaking steering maneuver. In our approach, the steering control sequence was solely influenced by the velocity factor, resulting in consistent trajectories for the same velocity sequence. As a result, the disparities between the ideal and actual trajectories were significantly reduced compared to conventional methods. Table 2 reveals that our method demonstrated better performance in terms of trajectory fitting.

**Overtaking ability comparison.** Existing lane-changing research methods generally involve motion planning based on the assumption of ideal perceptual information. Therefore, we compared the performance of our approach under two scenarios: one where the information about other vehicles was known, and another where the information about other vehicles was unknown. When the information about other vehicles was known, relative velocities and positions between vehicles were calculated, and *GDCPs* were employed to ensure path safety robustly. Simultaneously, in cases where the speeds of other vehicles were known, if the leading vehicle did not cooperate by reducing its speed, our method could abandon the overtaking attempt and ensure a return to the original lane within a certain timeframe. This contributed significantly to reducing the number of overtaking maneuvers initiated ( $M_o$ ). Our approach followed the principle of emulating human driver behavior, overtaking only when success is certain, resulting in a shorter time taken for overtaking ( $M_t$ ) in comparison to other methods. In scenarios where perception accuracy is high, the advantages of the vehicular networking series of methods in terms of overtaking success are undeniable. This is because these methods not only acquire the positional information of surrounding vehicles but also gain insight into the intentions of other vehicles, which makes executing overtaking maneuvers more convenient. However, the requirement for all vehicles in the driving environment to be equipped with vehicular networking systems reveals certain drawbacks. Although the success rate of overtaking significantly drops when the perception information is less than ideal, the drop is not as pronounced compared to other perception-aware methods. This provides evidence that an end-to-end approach for achieving lane changes is indeed feasible.



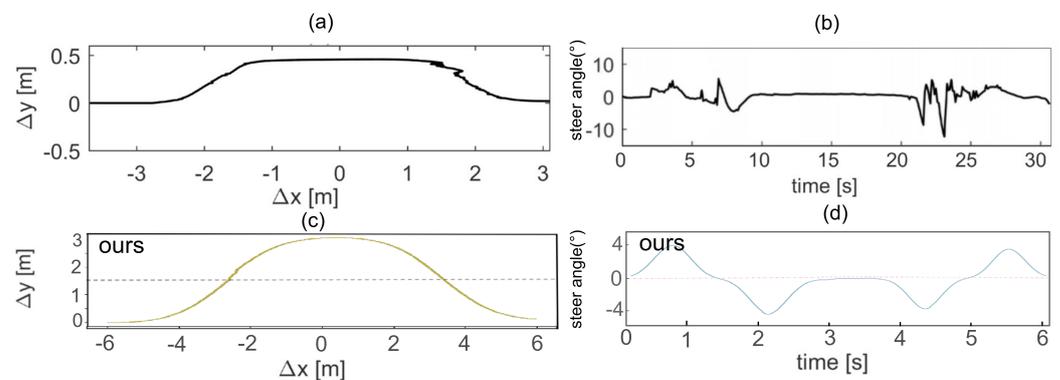
**Figure 4.** The output of the OFF. The left side of each row is an average of three pictures captured in a certain overtaking scenario. The right side of each row shows the fitting performance of the predictor under different scenarios. In most cases, the predicted probability was close to and generally lower than the label probability.

**Table 2.** Track matching metric. The table shows the difference between the vehicle trajectory and the ideal trajectory within three seconds from the overtaking action when the vehicle speed was 20 m/s. It also includes a comparison of lane occupation time, number of attempts, and overtaking success rate when performing overtaking tasks, as well as the collision rate, road segment completion rate, and time per kilometer when driving on a straight road.

Method	L2 (m)			$M_t$	$M_o$	$M_s$	$P_c$	$T_c$	Route Completion
	1 s	3 s	5 s						
EMPlanner [2]	0.291	1.853	4.134	8.2	1.09	72.6	0.73	194.5	99.4
TCP [21]	—	—	—	—	—	—	0.84	215.6	97.1
Cooperation [26]	0.285	1.977	4.066	11.2	6.8	78.1	<b>0.58</b>	159.4	99.2
Global Info [26]	0.279	2.011	4.153	11.4	6.5	<b>84.3</b>	1.47	182.8	98.3
KF Control [27]	0.314	<b>1.799</b>	3.976	10.7	33.5	63.4	2.88	198.5	89.0
GDCP	<b>0.276</b>	1.919	<b>3.906</b>	<b>5.2</b>	<b>3.2</b>	81.7	0.71	<b>147.4</b>	98.7
GDCP-OFP	0.289	1.936	4.011	5.4	7.9	62.3	2.42	164.2	94.6

**Smoothness analysis.** Traditional path planning involves first planning trajectory points and then executing trajectory tracking control. This approach offers the advantage of better robustness during the planning process, making it suitable for most autonomous driving scenarios. However, the scenarios addressed in this experiment involved relatively straight-

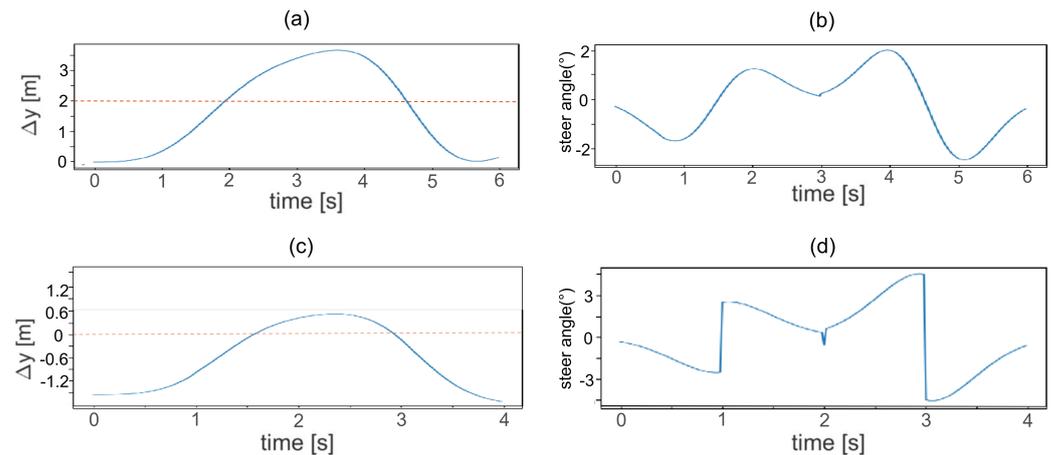
forward route options, encompassing only three situations: following a vehicle, overtaking, and returning to the original lane when danger is detected. The results of the experiment indicated that employing a method that directly plans control signals with varying maximum lateral displacements can effectively complete these tasks. This approach not only reduces the computational complexity but also ensures the smoothness of both the trajectory and control signals, ultimately enhancing passenger comfort. Huang et al. [4] built an overtaking environment with a mini car and recorded the overtaking track and the steering wheel angle signal. As one of the rare approaches that utilize common function curves for overtaking trajectory planning, our method was compared to the abovementioned method, and the trajectory outcomes are illustrated in Figure 5. Assuming that the speeds of the leading vehicle and the ego vehicle were constant, and using the geometric curve to plan the overtaking path, the route planned by the model in this paper was smoother, and the fluctuation of the control signal was not severe. This was because Huang et al.'s solution was to plan the trajectory first and then use the PID algorithm to restore the planned trajectory as much as possible. Our solution was to directly calculate the executed control signal based on the derivative of each key point of the target curve, so that both the control signal and the trajectory looked smoother.



**Figure 5.** (a,b) Smoothness of overtaking trajectory and control signal variation, where (c,d) indicate the fitting effect of the method in this paper.

### 3.4. Ego Vehicle Travels at Varying Speeds

In an actual overtaking operation, it is impossible for the vehicle to keep running at a constant speed to complete the overtaking maneuver. Overtaking is often accompanied by the behavior of accelerating first, then overtaking, and then slowly returning to the main lane after deceleration. As shown in Figure 6, we tested the trajectories and control signal changes of two cases under variable speed conditions. The experiments showed that although the vehicle control signal was not stable when returning to the main lane in case of danger, the vehicle trajectory was relatively smooth and did not affect passenger comfort. Additionally, in the test with a total driving distance of 20 km, the collision rate was 3.7% when the vehicle failed to return to the original lane during overtaking. Moreover, our model ensured relatively safe lateral displacement and complete overtaking even under variable speed conditions.



**Figure 6.** (a) The trajectory of the vehicle accelerating and overtaking before decelerating and returning to the original lane, (c) the trajectory of the vehicle returning to the original lane when it tried to overtake and encountered danger. (b,d) show the changes in the steering wheel angle in the two cases.

### 3.5. Ablative Study and Visualization

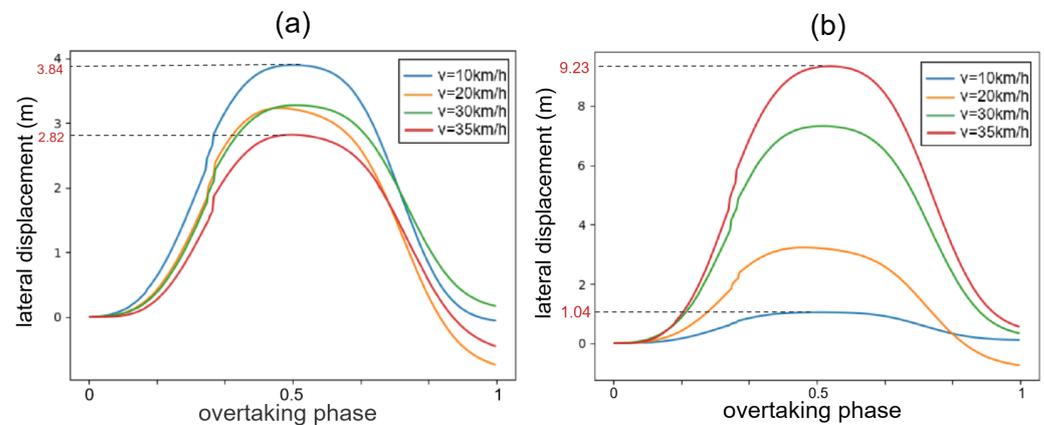
**Predictor component analysis.** In this paper, the ablation experiments of several innovative points were analyzed to verify the importance of each part, and the results are shown in Table 3. Firstly, in different overtaking stages, different image features may correspond to similar safety probabilities. Since we discarded the temporal network to learn contextual information, and the number of parameters required for optimal fitting varied across different overtaking stages, the Phase Selector (PS) played a crucial role. Due to the continuous distribution of probability label values in each group of data, it was challenging to quickly distinguish the features among the data. Numerous experiments have shown that directly importing all data for training without a pre-trained model can easily lead to oscillations and failure to converge at local optima. By importing the dataset in batches based on probability labels and prioritizing highly discriminative data for training, we achieved satisfactory fitting results.

**Table 3.** This paper considers a correct prediction to be obtained when the difference between the output probability and the probability label of the dataset is within 0.15.

ADIT	PS	Loss Function	Optimal Loss	Optimal Accuracy
✓	×	Our Loss	0.425	76.6
✓	✓	BCELoss	0.381	79.6
×	✓	Our Loss	0.491	68.2
×	×	BCELoss	0.488	53.1
✓	✓	Our Loss	<b>0.361</b>	<b>86.7</b>

**Velocity adaptation coefficient.** If one calculates the second derivative of the planned overtaking trajectory to derive the steering control signal sequence and then executes steering control at each moment, with a fixed velocity, the vehicle's trajectory will be quite stable. However, in reality, different situations will result in varying velocities. If the vehicle's speed is higher than the ideal speed, the traveled distance at each position will be greater, causing an increase in lateral displacement that could extend beyond the lane. Conversely, if the lateral displacement decreases, the vehicle could collide with the leading vehicle. Therefore, the introduction of the parameter  $\alpha$  can help control the variation in trajectory length while ensuring that even after a simple second-derivative calculation, the vehicle's maximum lateral displacement is maintained. According to Figure 7, it can

be seen that within the range of  $v \in [10, 35]$ , the maximum lateral displacement was in the range of  $[2.82, 3.84]$  when  $\alpha$  was introduced, while the maximum lateral displacement without introducing  $\alpha$  was in the range of  $[1.04, 9.23]$ .



**Figure 7.** Variation in lateral displacement during overtaking phase at different speeds. (a) shows the displacement change introduced by  $\alpha$ . (b) shows the displacement change without  $\alpha$ .

#### 4. Conclusions

This paper proposed a novel framework for lane overtaking using an end-to-end model. It explored the use of fixed geometric curves in overtaking scenarios and introduced a dataset that emphasizes the importance of including incorrect driving operations as training data and provides new ideas for datasets in the field of autonomous driving. The experimental results demonstrated the effectiveness of this model, even in challenging overtaking situations. Our approach has advantages in terms of trajectory smoothness and passenger comfort, as well as time per kilometer. Even without the assistance of high-precision maps, the performance will not lag much behind other methods that use map information as a priori knowledge in overtaking scenarios. However, the application scenarios of this model are still very limited. Future work will focus on extending overtaking application scenarios, such as multi-lane overtaking, and we will improve the accuracy of feasibility probability prediction and the stability of trajectories to enhance the practical application of autonomous driving.

**Author Contributions:** Conceptualization, K.L.; methodology, D.L.; software, D.L.; validation, K.L. and D.L.; formal analysis, K.L.; investigation, D.L.; resources, K.L.; data curation, D.L.; writing—original draft preparation, D.L.; writing—review and editing, K.L. and D.L.; visualization, D.L.; supervision, K.L. and D.L.; project administration, K.L.; funding acquisition, D.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Natural Science Foundation of Hebei Province (F2022201009).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data that support the findings of this study are available upon request. The data are not publicly available due to privacy.

**Conflicts of Interest:** The authors declare no conflict of interest.

#### References

1. Da, F. Comprehensive Reactive Safety: No Need for A Trajectory If You Have A Strategy. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022; IEEE: Toulouse, France, 2022; pp. 2903–2910.
2. Fan, H.; Zhu, F.; Liu, C.; Zhang, L.; Zhuang, L.; Li, D.; Zhu, W.; Hu, J.; Li, H.; Kong, Q. Baidu apollo em motion planner. *arXiv* **2018**, arXiv:1807.08048.

3. Kurzer, K. Path Planning in Unstructured Environments: A Real-Time Hybrid A\* Implementation for Fast and Deterministic Path Generation for the kth Research Concept Vehicle. Master's Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2016.
4. Huang, X.; Zhang, W.; Li, P. A path planning method for vehicle overtaking maneuver using sigmoid functions. *IFAC-PapersOnLine* **2019**, *52*, 422–427. [[CrossRef](#)]
5. Ding, W.; Zhang, L.; Chen, J.; Shen, S. Safe trajectory generation for complex urban environments using spatio-temporal semantic corridor. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2997–3004. [[CrossRef](#)]
6. Brito, B.; Agarwal, A.; Alonso-Mora, J. Learning interaction-aware guidance policies for motion planning in dense traffic scenarios. *arXiv* **2021**, arXiv:2107.04538
7. Chen, B.; Xu, M.; Liu, Z.; Li, L.; Zhao, D. Delay-aware multi-agent reinforcement learning for cooperative and competitive environments. *arXiv* **2020**, arXiv:2005.05441
8. Ghimire, M.; Choudhury, M.R.; Lagudu, G.S.S.H. Lane Change Decision-Making through Deep Reinforcement Learning. *arXiv* **2021**, arXiv:2112.14705
9. Yuan, F.; Shou, L.; Pei, J.; Lin, W.; Gong, M.; Fu, Y.; Jiang, D. Reinforced multi-teacher selection for knowledge distillation. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021; Volume 35, pp. 14284–14291.
10. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M. Playing atari with deep reinforcement learning. *arXiv* **2013**, arXiv:1312.5602
11. Rasch, A.; Dozza, M. Modeling drivers' strategy when overtaking cyclists in the presence of oncoming traffic. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 2180–2189. [[CrossRef](#)]
12. Liu, J.; Luo, Y.; Zhong, Z.; Li, K.; Huang, H.; Xiong, H. A probabilistic architecture of long-term vehicle trajectory prediction for autonomous driving. *Engineering* **2022**, *19*, 228–239. [[CrossRef](#)]
13. Eysenbach, B.; Salakhutdinov, R.R.; Levine, S. Robust predictable control. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 27813–27825.
14. Mo, C.; Li, Y.; Zheng, L. Simulation and analysis on overtaking safety assistance system based on vehicle-to-vehicle communication. *Automot. Innov.* **2018**, *1*, 158–166. [[CrossRef](#)]
15. Hegde, B.; Bourouche, M. Design of AI-based lane changing modules in connected and autonomous vehicles: A survey. In Proceedings of the Twelfth International Workshop on Agents in Traffic and Transportation, ATT 2022, Vienna, Austria, 25 July 2022.
16. Zeng, W.; Luo, W.; Suo, S.; Sadat, A.; Yang, B.; Casas, S.; Urtasun, R. End-to-end interpretable neural motion planner. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8660–8669.
17. Rhinehart, N.; McAllister, R.; Levine, S. Deep imitative models for flexible inference, planning, and control. *arXiv* **2018**, arXiv:1810.06544
18. Han, L.; Wu, L.; Liang, F.; Cao, H.; Luo, D.; Zhang, Z.; Hua, Z. A novel end-to-end model for steering behavior prediction of autonomous ego-vehicles using spatial and temporal attention mechanism. *Neurocomputing* **2022**, *490*, 295–311. [[CrossRef](#)]
19. Chen, L.; Tang, T.; Cai, Z.; Li, Y.; Wu, P.; Li, H.; Shi, J.; Yan, J.; Qiao, Y. Level 2 autonomous driving on a single device: Diving into the devils of openpilot. *arXiv* **2022**, arXiv:2206.08176.
20. Kalaria, D.; Lin, Q.; Dolan, J.M. Towards Safety Assured End-to-End Vision-Based Control for Autonomous Racing. *arXiv* **2023**, arXiv:2303.02267.
21. Wu, P.; Jia, X.; Chen, L.; Yan, J.; Li, H.; Qiao, Y. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. *arXiv* **2022**, arXiv:2206.08129.
22. Perumal, P.S.; Wang, Y.; Sujasree, M.; Mukthineni, V.; Shingekar, S.R. Intelligent advice system for human drivers to prevent overtaking accidents in roads. *Expert Syst. Appl.* **2022**, *199*, 117178. [[CrossRef](#)]
23. Mandai, G.; Bhattacharya, D.; De, P. Real time vision based overtaking assistance system for drivers at night on two-lane single carriageway. *Comput. Syst.* **2021**, *25*, 403–416.
24. Wu, D.; Liao, M.W.; Zhang, W.T.; Wang, X.G.; Bai, X.; Cheng, W.Q.; Liu, W.Y. Yolop: You only look once for panoptic driving perception. *Mach. Intell. Res.* **2022**, *19*, 550–562 [[CrossRef](#)]
25. Chang, J.; Wang, L.; Meng, G.; Xiang, S.; Pan, C. Deep adaptive image clustering. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5879–5887.
26. Tariq, F.M.; Suriyarachchi, N.; Mavridis, C.; Baras, J.S. Cooperative Bidirectional Mixed-Traffic Overtaking. In Proceedings of the 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022; IEEE: Toulouse, France, 2022; pp. 2494–2501.
27. Tariq, F.M.; Suriyarachchi, N.; Mavridis, C.; Baras, J.S. Vehicle overtaking in a bidirectional mixed-traffic setting. In Proceedings of the 2022 American Control Conference (ACC), Atlanta, GA, USA, 8–10 June 2022; pp. 3132–3139.

**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.