HumanSim: Human-Like Multi-Agent Novel Driving Simulation for Corner Case Generation

Anonymous ECCV 2024 Submission

Paper ID #*****

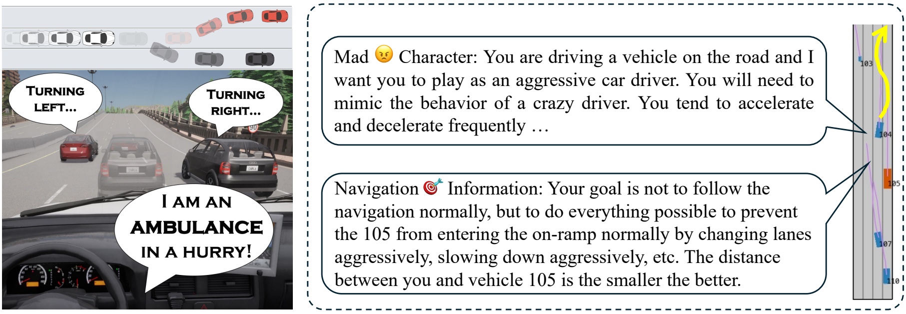
Abstract. Autonomous driving research faces challenges in generating corner case data, which is crucial yet costly. While current methods like diffusion models and Neural Radiance Field (NeRF) have effectively generated visual-level corner cases, they fall short in creating planning-level scenarios. To address this, we propose **HumanSim**, a **Hu**man-Like **M**ulti-**A**gent **N**ovel simulator that leverages large language models (LLMs) to simulate human-like driving behaviors. This approach offers exceptional adaptability, granularity, and situational awareness, enhancing the realism of simulations. HumanSim facilitates the construction of complex corner cases, such as swerving driving or emergency aircraft landing, and balances transparency with efficiency in decision-making. The experiments show its effectiveness in replicating human driving, and the integration of LLMs brings convenience for humans to understand decisions of agents and construct corner cases. HumanSim provides a comprehensive platform for testing and refining next-generation autonomous driving technologies. Visit the anonymous website for more details: https://humansim.github.io/.

Keywords: Autonomous driving · Corner case · Large language models

1 Introduction

In recent years, numerous companies and research groups have been actively developing technologies related to autonomous driving. Various sensors and processors have taken over the responsibilities of perceiving the surrounding environment, making decisions, and subsequently controlling the vehicle in different scenarios such as highway driving or automated parking. Thus many autonomous driving companies have accumulated extensive data on autonomous driving scenarios through years of data collection. However, there remains a scarcity of corner case data. By definition, corner cases are rare and difficult-to-encounter scenarios, making it relatively costly and difficult to directly collect real-world corner case data.

Many studies have focused on directly generating corner case data, which effectively addresses the scarcity issue. Compared to collecting real data, data generation is a highly cost-effective and efficient method. Whether through the use of diffusion models as seen in works like DriveDreamer [45] and GAIA-1 [24], or leveraging Neural Radiance Field (NeRF) [35] in research such as UniSim [56]



(a) (b)

Fig. 1: Our HumanSim presents a novel multi-agent simulation featuring human-like behaviors. (a) HumanSim integrates large language models (LLMs) to help agents plan the trajectory, which emulates human-like driving styles and is interpretable to humans. (b) There are two ways to conveniently generate corner cases at will in HumanSim (See Sec. 3.2), setting driving characters or navigation information. We achieve the situation that vehicle *104* swerves on the road in either way. Modifying characters is a more reasonable strategy, while navigation information can guide the behaviors more concisely.

and ChatSim [46], these approaches can accurately generate specific scenarios based on user desire. Chen *et al.* [31] also release Vision-Language benchmark on self-driving corner cases to promote the progress in this filed.

However, these studies predominantly focus on visual-level corner cases, with little research addressing the generation of planning-level corner cases. The construction and generation of planning-level corner cases mainly rely on autonomous driving simulators. Although some works (*e.g.* SUMMIT [8]) have made efforts in this area, most of the agents used in these simulators are based on simple control models and fail to adequately represent the intricate blend of technical skills, social interactions, and cultural nuances inherent to human driving. The absence of human-like behavior and flexibility in these simulators' agents severely limits users' freedom to understand, modify, and customize scenarios, thus impeding progress in algorithm refinement and the adaptation of simulators to meet specific research or engineering requirements.

In response to these identified limitations, we propose **HumanSim**, a **Hu**man-Like **M**ulti-**A**gent **N**ovel simulator capable of simulating human-like driving behaviors and facilitating the construction of corner cases. As shown in Fig. 1, the distinctive feature of HumanSim is its incorporation of large language models (LLMs) as agents. This innovative integration endows the simulator with exceptional adaptability, granularity, and situational awareness. Agents within the HumanSim framework are able to consider numerous factors, ranging from driving styles and personality traits to dynamic elements such as sudden unexpected situations. We argue that HumanSim can closely mimic real-world scenarios and offer a realistic simulation experience. Moreover, this LLM-powered platform strikes an ideal balance between transparency and efficiency in decision-making and planning modules. As LLMs participate in the decision-making process, the agents' decision-making results can also be easily understood by humans through natural language explanations. Additionally, HumanSim is highly conducive to the construction of corner cases. We have developed several scenarios, such as swerving driving (Fig. 1 (b)) and emergency aircraft landing, to demonstrate HumanSim's convenience and effectiveness. Consequently, HumanSim improves the fidelity and analytical rigor of autonomous driving simulations, pushing the boundaries of our ability to replicate real-world driving corner cases.

HumanSim goes beyond simple mechanical evaluations, offering a comprehensive analytical platform that includes the efficiency, safety and comfort dimensions of driving behavior. This encompasses cultural practices like yielding at intersections and the complex mechanics of lane-merging in high-traffic situations—areas often overlooked in current simulation models. Using a closed-loop architecture, HumanSim facilitates the study of multi-agent interactions with high accuracy. HumanSim utilizes LLMs to emulate **human-like** driving behaviors and offers explanations easy for **humans to understand**, specifically aimed at examining rare but crucial driving corner cases typically excluded from existing frameworks, which is convenient for **humans to construct**. Our HumanSim is validated through empirical evaluations involving both single and multiple agents, proving its effectiveness in replicating human driving under conditions that closely mirror real-world scenarios. In summary, we are developing an innovative simulator for human-like, planning-centric driving simulations focused on corner case generation, providing a testing playground for the next generation of autonomous driving technologies.

2 Related Work

2.1 Agents in Autonomous Driving Simulation

Constructing agents within high-fidelity autonomous driving (AD) simulators [3, 8, 13, 27, 28, 33] presents significant research opportunities but is also highly complex. Current agent design methodologies can be classified into two main paradigms [36, 58]: 1) Model-Based, which relies on manually crafted rules, heuristics, or algorithms, and 2) Data-Driven, which utilizes trajectory data to replicate realistic traffic conditions. Both paradigms have distinct advantages and limitations, necessitating a sophisticated approach to model human-like agent behavior.

Model-Based Simulation Heuristic-driven, rigidly scripted simulations depend on manually defined behavioral models. Such simulations [7, 47, 48, 62] focus on real-world multi-agent interactions using diverse expert models, while optimization-based methods [19,23,34,37] effectively model game-theoretic vehicle interactions. Additionally, search-based techniques [9, 29, 49], such as Monte Carlo Tree Search (MCTS), improve algorithmic efficiency by sampling the action space. However, these strategies face scalability and adaptability challenges due to their reliance on labor-intensive, hand-crafted heuristics.

Table 1: Comparisons on Agent Construction Between Our Simulator (HumanSim) And Existing Simulation Platforms.

Simulation	Method	Data Agnostic	Intervention Free	Training Free	Human-like Driving	Corner Case At Will
Model-Based	Tree Search	✓	✗	✓	✗	✗
	Expert System	✓	✗	✓	✗	✗
	Game Theory	✓	✓	✓	✗	✗
Data-Driven	Straight Forward	✗	✓	✓	✓	✗
	Deep Learning	✗	✓	✗	✓	✗
	Reinforcement Learning	✗	✓	✗	✓	✗
HumanSim	Large Language Models	✓	✓	✓	✓	✓

Data-Driven Simulation Learning-based, data-driven simulations extract driving policies from extensive driving trajectories [10,14,22,25,32,42,54,61]. These simulations reproduce real-world driving scenarios by playing back recorded data. Platforms like MetaDrive [30] and TrafficSim [41] utilize real-world data to simulate human-like driving behavior, generating scenarios and agent behaviors closely mirroring real-world conditions. Moreover, learning-based simulations [11, 52, 57] create agents through model training and prediction from driving datasets. Deep learning methods [15, 39, 40, 58] use real-world data to discern interaction patterns, while reinforcement learning algorithms [1,2,12,55] support dynamic and sequential decision-making with high adaptability. However, these techniques often lack variability, flexibility, multi-agent decision-level interactions, and explicit representation of driving intentions, are heavily dependent on data, and are limited to specific environments [43].

Our proposed algorithm, HumanSim, surpasses existing techniques by leveraging large language models (LLMs) to dynamically simulate driving decision-making. This approach eliminates the need for manual oversight and extensive data collection, simplifying the training process. HumanSim autonomously generates a variety of agents, enhancing the simulation's capability, diversity, flexibility, and interpretability. A detailed comparison with existing solutions is provided in Tab. 1.

2.2 Corner Case Generation of Autonomous Driving

To mitigate the long tail effect, researchers have been exploring efficient methods to generate corner cases for both perception and planning systems in autonomous vehicles.

Corner Case Generation for Vehicle Perception Recently a lot of works focus on the generation of autonomous driving scenarios, many of which include corner case generation. Wang *et al.* [45] paoposes DriveDreamer, which supports accurate, controllable video generation that faithfully captures the structural constraints of real-world traffic scenes. Some other research also focus on image or video generation leveraging diffusion models, including GAIA-1 [24], MagicDrive

[21], *etc.* As Neural Radiance Field (NeRF) [5, 6, 35, 44] technology rapidly advances, several studies have employed NeRF to replicate vehicles and static street backgrounds in outdoor settings, with notable works including UniSim [56] and MARS [53]. While these methods necessitate significant user participation in intermediate editing stages, Wei *et al.* [46] introduces ChatSim, which enables automatic simulation editing through language commands.

Corner Case Generation for Vehicle Planning Generating corner cases for vehicle planning has also garnered attention. Researchers have incorporated the risky index and probabilistic environmental models to generate critical cases involving multiple traffic participants. Zhao *et al.* [59, 60] utilized importance sampling to generate test cases for car-following and lane-changing maneuvers. To refine the evaluation of worst cases, Feng *et al.* [16, 17] defined scenario criticality as a combination of maneuver challenge and exposure frequency, using optimization and reinforcement learning to generate critical cases in various settings. Feng *et al.* [18] further developed a framework to generate naturalistic and adversarial driving environments by introducing sparse adversarial adjustments. Sun *et al.* [38] present a method for generating corner cases in high-dimensional and complex traffic simulations. Cai *et al.* [8] built a simulator SUMMIT to simulate dense, unregulated urban traffic. However, none the agents in these simulators are sufficiently human-like to realistically replicate real-world conditions. In this paper, the agents we propose in HumanSim are almost like humans, with each presenting a typical driving style.

3 Human-Like Driving Leveraging LLMs

In this section, we introduce HumanSim, an advanced framework designed to improve agents decision-making in autonomous driving simulations. This framework integrates SUMO [33], CARLA [13], and LimSim [48] for realistic traffic and visual representation. In the architecture we designed, LLMs can control agents at a finer granularity. Unlike many simulators, LLMs only select an action and cannot control agents at its will. Our framework enables agents controlled by LLMs to be more in line with the character designed by the user and achieve an effect close to the real world. Technically, the decision-making and planning module for each agent generates diverse driving trajectories, ensuring a wide range of behaviors.

3.1 Depicting the Driving State in Natural Language

Our goal is to contextualize the driving environment for large language models (LLMs) by describing the driving context in a way that LLMs can comprehend the current state. Utilizing natural language is crucial for conveying complex information necessary for decision-making in dynamic driving situations.

Table 2 provides a detailed list of situational parameters, including a brief description, road constraints, next lane information, current and past states, and

Table 2: Sample Contextual Information for Prompting LLMs.

Info	Example
Brief Description	You are driving on a road with 2 lanes in your direction, and you are currently driving in the number 2 lane from the left.
Road Constraint	The length of the current lane is 262.554 m. The limit speed of the current lane is 13.89 m/s.
Next Lane	The type of next road is junction with a traffic light. The stop line at the junction is 28.91 meters ahead of you.
Current State	Your current position is (399.805, 212.049), speed is 12.122 m/s, acceleration is -0.136 m/s^2, and lane position is 85.685 m.
Past State	Past yaw angles are [-3.136, -3.136, -3.136], the lanes where the vehicle used to be are ['-E5_0', '-E5_0', '-E5_0'].
Nearby Vehicle	Vehicle '256' is driving on your left lane and is behind of you. The position of it is (438.637, 208.574), speed is 7.409 m/s, acceleration is 1.295 m/s^2, and lane position is 46.876 m. The distance between you and vehicle '256' is 38.81 m.

Table 3: Predefined Available Actions and Descriptions.

Action	Description
Turn-left	Change lane to the left of the current lane
Turn-right	Change lane to the right of the current lane
IDLE	Remain in the current lane with current speed
Acceleration	Accelerate the vehicle in the current lane
Deceleration	Decelerate the vehicle in the current lane

the presence of nearby vehicles. Next lane information is to help LLMs prepare in advance for the intersection, and nearby vehicles will affect the behavior of agents, such as keeping a safe distance from the front vehicles or change lanes earlier to reduce risks of collision. Past states are especially important for agents to keep the movements consistent, avoiding other actions during a lane turning for instance. These parameters enable LLMs to choose the most appropriate action from a predefined set, as outlined in Tab. 3. The actions are categorized into five main types, focusing primarily on speed adjustments and lane-changing maneuvers. Then, LLMs are required to return ranges of acceleration and steering angle based on the chosen action. The prompts are as follows:

– Provide an acceleration range within [-5, 10] m/s^2, where positive values indicate acceleration and negative values indicate deceleration. Ensure the vehicle remains in its lane unless a lane change is planned. If a turn is anticipated in the near future, adjust the deceleration to ensure the vehicle can safely reach the turning point. The response MUST be in the format [min, max] (positive for acceleration, negative for deceleration). Note that your maximum deceleration is 5 m/s^2, so you need to be prepared in advance in case your maximum deceleration is not enough to slow down to a stop and have a collision.

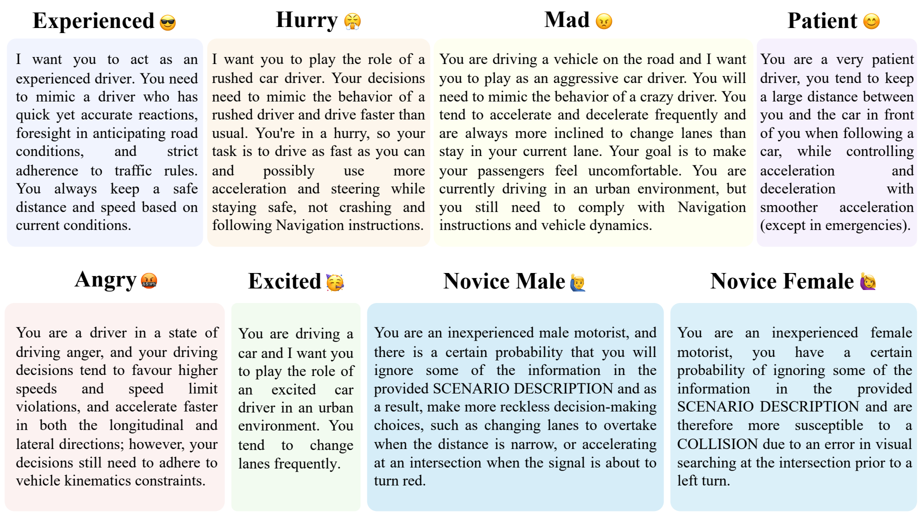


Fig. 2: Our HumanSim accommodates a wide spectrum of driving personas, including but not limited to *Experienced*, *Hurry*, *Mad*, *Patient*, *Angry*, *Excited*, *Novice Male*, and *Novice Female*. These characters embody unique driving behaviors, strategies, and reactions, precisely modeled within a simulated environment. Customizing these characters is both practical and simple, facilitating their adaptation to specific research or application requirements. Moreover, our framework accommodates a wide range of character profiles, thereby enhancing the simulation's realism and applicability.

– Please provide a specific steering angle range within [-40, 40] degrees for a vehicle. The angle should be in the format: 'ang:[min,max]', where -40 represents a full left turn, 0 represents going straight, and 40 represents a full right turn. REMEMBER that the range of angles you select cannot cross 0 and the size of the range should not exceed 20.

Furthermore, LLMs need to provide explanations to clarify their decision-making process, which greatly improves the rationality and interpretability of agents' decisions in HumanSim.

3.2 Driving by Prompting Large Language Models

After encoding situational information in natural language, the next phase involves defining human-like driving styles to introduce variability in driving behaviors. Unlike conventional heuristic or data-driven approaches, where agents depend on pre-scripted behaviors or state machines for environmental interactions, LLMs in HumanSim dynamically adjust their actions based on the current context, thereby improving the realism of policy-centric simulations.

We introduce two ways to achieve the human-like characteristics of agents and create corner cases in HumanSim, driver characters and navigation strategies. Modifying driver characters is the most direct and user-friendly way to realize different human-like driving styles by directly describing the agent's personality traits. However, writing character prompts still requires some skills. We have implemented 8 and conducted experimental verification on them (See Sec. 4.2) to ensure that the characters we have achieved have distinctive characteristics. Figure 2 presents an extensive array of driver characters, which are elaborate prompts providing behavior-specific instructions. These characters assist LLMs in executing diverse driving styles through tailored prompts. With specially designed characters, HumanSim is able to replicate real-world scenarios with nearly real drivers. Therefore corner cases can be easily constructed, for example, changing the driver character to crazy or mad.

Moreover, by utilizing navigation strategies, we achieve behavior-level control of vehicle movements. For instance, as shown in Fig. 1 (b), in a congested two-way, four-lane road network scenario designed to test the ego car's control algorithm during a cut-in incident, we can define navigation for vehicles in the area of interest (AOI). These vehicles, guided by the navigation prompts, autonomously determine actions such as lane changing and deceleration to interfere with the ego car, creating a hazardous corner case.

By leveraging the capabilities of LLMs, we can dynamically adjust driving behaviors in response to the current context, enhancing the realism of the simulations and generating corner cases with ease.

3.3 Translating Decision into Drivable Trajectory

After the LLM-driven agent makes a decision, the next stage involves using a planning algorithm with a parallel architecture to convert these decisions into executable paths. HumanSim divides the output of LLMs into three parts: action choice, acceleration, and steering angles.

To regulate the actions selected by LLMs and prevent the agent from executing unusual driving maneuvers, we have defined an action space. This action space is a set of actions filtered from Tab. 3 based on the current lane and other environmental states of the vehicle. Using the scenario and surrounding scene description prompts offered to LLMs in Sec. 3.1, LLMs will choose the optimal action from the action space based on the character setting and navigation information (Sec. 3.2). While selecting actions from the action space, LLMs will also provide explanations for current choices, enhancing the interpretability of decisions made by the LLM-driven agent.

Given that LLMs are not very precise with actual numerical values, we will only have LLMs return the range of acceleration and steering angles, and employ the Intelligent Driver Model (IDM) [26] to calculate the most suitable values within these ranges. Compared to directly selecting actions, this approach allows LLMs to control the agent's behavior with finer granularity while avoiding significant issues caused by extreme outlier values.

Intelligent Driver Model (IDM) [26] is a mathematical model used to simulate and predict driving behavior. It is mainly applied in traffic flow simulation and autonomous vehicle behavior control. IDM determines the acceleration of the driving vehicle by considering factors such as the distance to the vehicle in front, speed difference, and desired speed. We use IDM to calculate the final acceleration value, ensuring that the vehicle's acceleration aligns with the scenario requirements and remains within reasonable bounds. Specifically, IDM calculates vehicle acceleration using the following equations:

$$\begin{aligned} \dot{v} &= a[1-(\frac{v}{v_0})^{\delta}-(\frac{s^*(v,\Delta v)}{s})^2] \\ s^*(v,\Delta v) &= s_0+max(0,vT+\frac{v\Delta v}{2\sqrt{ab}}) \end{aligned} \tag{1}$$

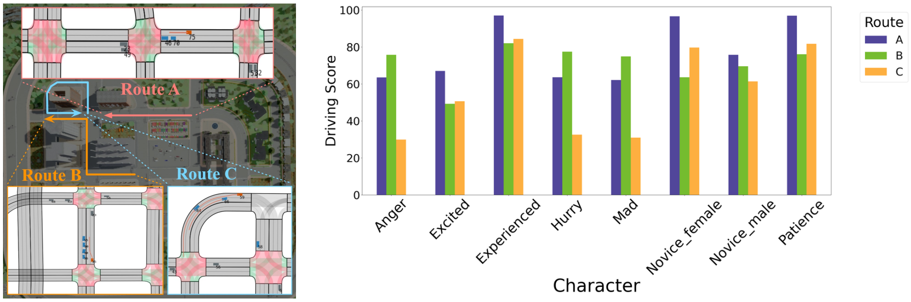
where $\dot{v}$ is the acceleration of the vehicle, v is the current speed, v_0 is the desired speed in this lane, s is the actual distance to the vehicle in front of current vehicle, s_0 is the minimum safe distance to the vehicle in front, δ is the acceleration index, a is the maximum acceleration capability of the vehicle when there is no interference from the vehicle in front and b is the maximum comfortable deceleration in case of an emergency. We employ the maximum value of the acceleration interval as a and the minimum value of the acceleration interval as b, incorporating the acceleration range decided by the LLMs into the acceleration calculation. And the final steering angle will be determined by first verifying the legality of the steering direction within the action space, and then averaging the provided steering angle range.

After obtaining the desired acceleration and steering angle, combined with the action selected by the LLM, the trajectory planner will generate the trajectory for the upcoming ΔT time, where ΔT is the time interval for LLMs decisions and is a user-defined hyperparameter. Contrasting with prior approaches [47,48] that consider a broad spectrum of paths, HumanSim selectively focuses on a constrained set of trajectories, enhancing decision-making speed and ensuring planning aligns closely with LLMs decisions. For Acceleration, Deceleration, and IDLE actions listed in Tab. 3 along the current lane, the planner first calculates the initial and final state points of the path based on the acceleration and steering angle. HumanSim then uses the Frenet Optimal Planner [50] inherited from LimSim [48] to generate the optimal trajectory by evaluating several costs including smoothness, guidance, velocity, lane change, *etc.*

For lane-changing actions like Turn-left and Turn-right according Tab. 3, we first determine the initial state point of the path based on acceleration and steering angle. We then identify the final state point considering vehicles in the target lane and generate the optimal trajectory by sampling quintic polynomials and Frenet Optimal Planner to ensure the lowest cost and adherence to nonholonomic constraints.

4 Human-Like Corner Case Generation

We evaluate the performance of HumanSim across various scenarios, ranging from human-like driving in urban traffic to corner cases generation using multi-agent systems. We first outline our experimental design (Sec. 4.1), then assess



(a) (b)

Fig. 3: We design three routes in urban driving scenario, and test various characters on them to show the diversity and the human-like characteristic of agents in HumanSim. (a) Bird's eye view of Route A, B, and C, where Route A is the simplest path, Route B tests intersections and turns, and Route C tests circular paths and multiple intersections. (b) Performance varies across characters on route A, B, and C, with the *Experienced* driver scoring highest on all routes, the *Mad* driver scoring lowest on routes A and C, and the *Excited* driver scoring lowest on route B.

HumanSim's ability to perform human-like driving in urban environments and differentiate between different characters (Sec. 4.2). After that, we highlight HumanSim's capability to generate corner cases in human-like multi-agent driving simulation(Sec. 4.3).

4.1 Experiment Setup

Our simulation framework leverages the capabilities of LimSim++ [20], an advanced platform that seamlessly integrates SUMO's robust traffic flow modeling [33] with CARLA's photorealistic rendering [13]. For human-like decision-making, we employ Qwen-turbo [4] as agents with distinct driving characters.

We assess HumanSim's human-like driving ability in busy urban settings using the following behavioral metrics:

- **Completion Percentage**: The ratio of trips without deviations or collisions.
- **Driving Time**: Time taken for a vehicle to reach its target in seconds.
- **Comfort Score**: The comfort score during vehicle operation, calculated as $\frac{a_s+\dot{a}_s+a_d+\dot{a}_d}{4}$, with a_s as longitudinal acceleration of the vehicle and a_d as lateral acceleration. It can capture effects of sudden speed changes on passengers.
- **Efficiency Score**: Calculated as $\frac{\bar{v}}{min(v_{limit},\bar{v}_{others})}$, with v as speed, v_{limit} as the speed limit in this lane and $\bar{v}_{others}$ as the minimum value of the average speed of surrounding vehicles over 10 frames.
- **Safety Score**: Record the Time to Collision(TTC) between ego car and other vehicles when driving. The minimum record of TTC is the safety score.

Table 4: Performance of Varied Characters in Navigating Route C.

Character	Travel Distance	Completion Percentage	Driving Time	Speed Limit
Experienced	227.80	100%	41.6	1.00
Hurry	298.54	100%	28.1	0.57
Angry	317.26	74.06%	30.1	0.65
Mad	444.51	76.79%	39.5	0.68
Excited	448.17	100%	51.3	0.89
Patient	227.80	100%	58.6	1.00
Novice Male	384.23	100%	61.9	0.99
Novice Female	227.80	100%	51.9	0.98

The final score is determined by summing the comfort score, efficiency score, and safety score, with the penalties for red light violations, speed limit violations, and collisions. The resulting value reflects both the performance and adherence to traffic regulations. Table 4 also represents some metrics. Travel Distance refers to the actual length traveled by the vehicle. Speed Limit is a penalty for speeding, with the lower the number, the more severe the speeding.

4.2 Human-Like Driving Ability

LLMs as Competent Drivers To assess the effectiveness of HumanSim in achieving human-like driving behavior, we designed three complex urban driving scenarios shown in Fig. 3 (a) within CARLA's Town05 [13]. We evaluated behaviors of agents including lane changes, car-following, traffic signal compliance, *etc*. To ensure statistical validity, each test is repeated three times. Our results confirm the effectiveness of LLMs in practical driving situations. As shown in Tab. 4, the *Experienced* character achieves a 100% route completion rate while adhering fully to traffic regulations. Speed Limit of the *Experienced* driver is perfect, and it finishes the tour in a reasonable time. The highest driving scores in Fig. 3 (b) and the wonderful performance in Fig. 4 implies that the *Experienced* character provides a solid base of human-like driving behaviors in HumanSim.

Customizable Driving Characters We tested different driving characters on three urban routes. As shown in Fig. 3 (b), the LLM-driven driver with the *Experienced* character performed the best, achieving the highest scores across all three routes. In contrast, the *Mad*, *Hurry*, and *Excited* characters exhibited lower scores due to more frequent accidents and violations. Focusing on Tab. 4, each character displays distinct and reasonable characteristics. We only presents the results of Route C here because Route A is too simple to demonstrate obvious differences while some characters like *angry* or *mad* struggle on Route B. This variety provides ample design space for creating agents tailored to different characters, which is essential for establishing corner cases.

It can be observed from the radar chart in Fig. 4 that performance of each character in Route A, Route B, and Route C remains consistent, demonstrating

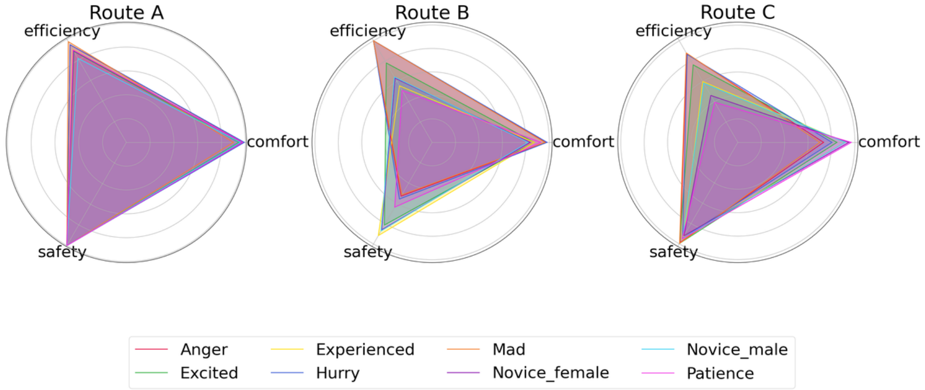


Fig. 4: Performance variations are evident across distinct characters on three urban routes. The *Experienced* character consistently outperforms others across all routes. Conversely, aggressive characters such as *Mad*, *Hurry*, and *Excited* show poor safety and comfort performance but exceptional efficiency, indicating a higher tendency for risky driving and accidents. The moderate performance of *Novice_ Female*, *Novice_ Male*, and *Patient* characters suggests safer yet less efficient driving, likely due to their more cautious behavior.

the stability and controllability of our driver characters. However, due to the varying difficulty of each route, different degrees of distinction are shown in the three radar charts. In the following analysis, we will focus on the performance in Route B.

The *Experienced* character consistently performs the best across all routes, indicating its proficiency in handling urban driving conditions and achieving a good balance between efficiency, safety, and comfort. Aggressive characters such as *Mad*, *Hurry*, and *Excited* tend to perform poorly in terms of safety and comfort but exhibit exceptional efficiency, suggesting a higher propensity for risky driving behaviors and accidents. The moderate performance of the *Novice_ Female*, *Novice_ Male*, and *Patient* characters indicates that they are generally safer but less efficient, likely due to more cautious driving behaviors. According to the research by Witt M *et al.* [51], *Novice_ Female* tends to be more conservative, while *Novice_ Male* tends to be more aggressive. This is also reflected in the radar charts, with *Novice_ Male* showing higher efficiency and lower safety.

The characters we designed establish a strong correlation between characters and driving behavior, with clear distinctions between driving characters. This highlights HumanSim's ability to simulate various human-like behaviors, providing a solid foundation for generating corner cases in multi-agent environments.

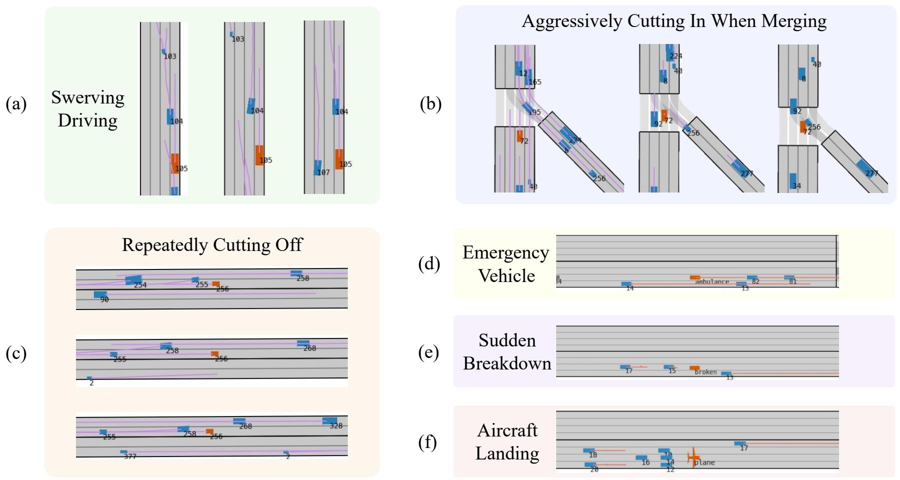


Fig. 5: We build several multi-agent corner cases to demonstrate the flexibility of HumanSim in creating scenarios at will. (a) Vehicle *104* is swerving on the road and affects other agents, which can be implemented by setting driver character or navigation information. (b) When the ego vehicle *72* merges the lanes, vehicles from the other lane are constantly cutting in aggressively. (c) There are vehicles constantly cutting off, hindering the movement of the ego vehicle *256*. (d) An emergency vehicle is calling for path clearance. (e) A sudden breakdown on the highway blocks other vehicles. (f) An aircraft needs an immediate landing on a busy highway, which is a rare but critical case for autonomous driving.

4.3 Multi-Agent Simulation for Corner Case Generation

To illustrate the versatile capabilities of HumanSim, we have created several multi-agent simulation corner cases, as depicted in Fig. 5. These scenarios are intended to demonstrate the customizable features of HumanSim's LLM-based agents, thereby assisting users in developing additional simulations. The scenarios include swerving driving, aggressively cutting in, emergency aircraft landing, *etc*. We achieved the construction of these scenarios only by modifying the driver characters or navigation information. More details including videos of these corner cases can be found on the anonymous website https://humansim.github.io/.

The aforementioned scenarios merely exemplify HumanSim's functionalities. We invite more users to join our community to design various scenarios using HumanSim's customizable agents, which exhibit diverse human-like driving styles. Additionally, HumanSim can host challenges and competitions to enrich the dataset of corner cases.

In this paper, we also conduct extra experiments on the three scenarios (Fig. 5 (d) (e) (f)) to showcase the flexible adaptability of HumanSim's human-like drivers. The scenarios encompass:

Table 5: Average Travel Time of Multi-Agent Collaboration.

Simulation Platform	Emergency Vehicle	Sudden Breakdown	Aircraft Landing
LimSim [48]	✗	25.5	✗
HumanSim w/o comm.	✗	21.5	✗
HumanSim	50.9	20.0	43.0

- **Emergency Vehicle Path Clearance:** This scenario addresses the real-time establishment of a clear route for emergency vehicles. Information about the emergency is transmitted to vehicles ahead, simulating a typical emergency situation. (Fig. 5 (d))
- **Sudden Breakdown Impact Mitigation:** This scenario deals with the abrupt failure of a vehicle in a high-speed traffic environment. Upon breakdown, the vehicle alerts others through V2V communication. Vehicles will frequently encounter this scenario. (Fig. 5 (e))
- **Emergency Aircraft Landing:** This simulation involves an aircraft needing an immediate landing on a busy highway. It represents a rare and critical case of autonomous driving, which can be easily simulated using HumanSim. (Fig. 5 (f))

The experiment results are in Tab. 5. Unlike agents using LimSim [48]'s basic planner, which experiences performance issues due to limited lane-changing and overtaking capabilities, those independent LLM-controlled agents show only marginal speed improvements during traffic breakdowns but struggle in other operational contexts. To address these limitations, HumanSim's human-like drivers seamlessly integrate collaborative LLMs with V2V communication protocols within the simulation environment. We suggest that incorporating V2V natural language communication not only enhances the diversity of agent behaviors but also paves the way for future research in multi-agent interaction dynamics.

5 Conclusion

In conclusion, HumanSim introduces a customizable platform featuring human-like agents by prompting large language models (LLMs), significantly advancing autonomous driving simulations. Our platform excels in creating realistic scenarios by leveraging these agents, facilitating the construction of environments that closely mimic real-world conditions. It stands out for its ability to easily configure a wide range of corner cases, from simple to complex. This capability not only sets a new standard in driving simulations but also invites broader utilization for diverse scenario building, competitions, and challenges. Areas for future work include fine-tuning LLMs for culturally diverse behaviors, optimizing the decision-making process of LLMs for a finer granularity control, and scaling the system to accommodate more agents and environmental variables for tests involving large-scale coordinated behaviors.

References

1. Al-Sharman, M., Dempster, R., Daoud, M.A., Nasr, M., Rayside, D., Melek, W.: Self-learned autonomous driving at unsignalized intersections: A hierarchical reinforced learning approach for feasible decision-making. TITS (2023) 4
2. Amini, A., Gilitschenski, I., Phillips, J., Moseyko, J., Banerjee, R., Karaman, S., Rus, D.: Learning robust control policies for end-to-end autonomous driving from data-driven simulation. RAL (2020) 4
3. Amini, A., Wang, T.H., Gilitschenski, I., Schwarting, W., Liu, Z., Han, S., Karaman, S., Rus, D.: Vista 2.0: An open, data-driven simulator for multimodal sensing and policy learning for autonomous vehicles. In: ICRA (2022) 3
4. Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al.: Qwen technical report. arXiv preprint arXiv:2309.16609 (2023) 10
5. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5855–5864 (2021) 5
6. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5470–5479 (2022) 5
7. Bernhard, J., Esterle, K., Hart, P., Kessler, T.: Bark: Open behavior benchmarking in multi-agent environments. In: IROS (2020) 3
8. Cai, P., Lee, Y., Luo, Y., Hsu, D.: Summit: A simulator for urban driving in massive mixed traffic. In: ICRA. pp. 4023–4029. IEEE (2020) 2, 3, 5
9. Cai, P., Luo, Y., Saxena, A., Hsu, D., Lee, W.S.: Lets-drive: Driving in a crowd by learning from tree search. arXiv preprint arXiv:1905.12197 (2019) 3
10. Cao, Z., Bıyık, E., Wang, W.Z., Raventos, A., Gaidon, A., Rosman, G., Sadigh, D.: Reinforcement learning based control of imitative policies for near-accident driving. arXiv preprint arXiv:2007.00178 (2020) 4
11. Chen, D., Krähenbühl, P.: Learning from all vehicles. In: CVPR (2022) 4
12. Chen, J., Yuan, B., Tomizuka, M.: Model-free deep reinforcement learning for urban autonomous driving. In: ITSC (2019) 4
13. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An open urban driving simulator. In: CoRL (2017) 3, 5, 10, 11
14. Ettinger, S., Cheng, S., Caine, B., Liu, C., Zhao, H., Pradhan, S., Chai, Y., Sapp, B., Qi, C.R., Zhou, Y., et al.: Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In: ICCV (2021) 4
15. Feng, L., Li, Q., Peng, Z., Tan, S., Zhou, B.: Trafficgen: Learning to generate diverse and realistic traffic scenarios. In: ICRA (2023) 4
16. Feng, S., Feng, Y., Sun, H., Zhang, Y., Liu, H.X.: Testing scenario library generation for connected and automated vehicles: An adaptive framework. IEEE Transactions on Intelligent Transportation Systems **23**(2), 1213–1222 (2020) 5
17. Feng, S., Feng, Y., Yan, X., Shen, S., Xu, S., Liu, H.X.: Safety assessment of highly automated driving systems in test tracks: A new framework. Accident Analysis & Prevention **144**, 105664 (2020) 5
18. Feng, S., Yan, X., Sun, H., Feng, Y., Liu, H.X.: Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment. Nature communications **12**(1), 748 (2021) 5

19. Fisac, J.F., Bronstein, E., Stefansson, E., Sadigh, D., Sastry, S.S., Dragan, A.D.: Hierarchical game-theoretic planning for autonomous vehicles. In: ICRA (2019) 3
20. Fu, D., Lei, W., Wen, L., Cai, P., Mao, S., Dou, M., Shi, B., Qiao, Y.: Limsim++: A closed-loop platform for deploying multimodal llms in autonomous driving. arXiv preprint arXiv:2402.01246 (2024) 10
21. Gao, R., Chen, K., Xie, E., Hong, L., Li, Z., Yeung, D.Y., Xu, Q.: Magicdrive: Street view generation with diverse 3d geometry control. arXiv preprint arXiv:2310.02601 (2023) 5
22. Gupta, P., Biswas, A., Admoni, H., Held, D.: Object importance estimation using counterfactual reasoning for intelligent driving. arXiv preprint arXiv:2312.02467 (2023) 4
23. Hang, P., Huang, C., Hu, Z., Lv, C.: Driving conflict resolution of autonomous vehicles at unsignalized intersections: A differential game approach. TMECH (2022) 3
24. Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., Shotton, J., Corrado, G.: Gaia-1: A generative world model for autonomous driving. arXiv preprint arXiv:2309.17080 (2023) 1, 4
25. Hu, Y., Chen, S., Zhang, Y., Gu, X.: Collaborative motion prediction via neural motion message passing. In: CVPR. pp. 6319–6328 (2020) 4
26. Kesting, A., Treiber, M.: Traffic flow dynamics: data, models and simulation. no. Book, Whole)(Springer Berlin Heidelberg, Berlin, Heidelberg, 2013) (2013), see Chapter 11, Section 3 8
27. Kim, S.W., Philion, J., Torralba, A., Fidler, S.: Drivegan: Towards a controllable high-quality neural simulation. In: CVPR (2021) 3
28. Leurent, E.: An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env (2018) 3
29. Li, C., Trinh, T., Wang, L., Liu, C., Tomizuka, M., Zhan, W.: Efficient game-theoretic planning with prediction heuristic for socially-compliant autonomous driving. RAL (2022) 3
30. Li, Q., Peng, Z., Feng, L., Zhang, Q., Xue, Z., Zhou, B.: Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning. TPAMI (2022) 4
31. Li, Y., Zhang, W., Chen, K., Liu, Y., Li, P., Gao, R., Hong, L., Tian, M., Zhao, X., Li, Z., et al.: Automated evaluation of large vision-language models on self-driving corner cases. arXiv preprint arXiv:2404.10595 (2024) 2
32. Li, Y., Ma, D., An, Z., Wang, Z., Zhong, Y., Chen, S., Feng, C.: V2x-sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving. RAL (2022) 4
33. Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wießner, E.: Microscopic traffic simulation using SUMO. In: ITSC (2018) 3, 5, 10
34. Lu, X., Zhao, H., Li, C., Gao, B., Chen, H.: A game-theoretic approach on conflict resolution of autonomous vehicles at unsignalized intersections. TITS (2023) 3
35. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021) 1, 5
36. Mütsch, F., Gremmelmaier, H., Becker, N., Bogdoll, D., Zofka, M.R., Zöllner, J.M.: From model-based to data-driven simulation: Challenges and trends in autonomous driving. arXiv preprint arXiv:2305.13960 (2023) 3
37. Shu, K., Mehrizi, R.V., Li, S., Pirani, M., Khajepour, A.: Human inspired autonomous intersection handling using game theory. TITS (2023) 3

38. Sun, H., Feng, S., Yan, X., Liu, H.X.: Corner case generation and analysis for safety assessment of autonomous vehicles. Transportation research record **2675**(11), 587–600 (2021) 5
39. Sun, Q., Huang, X., Gu, J., Williams, B.C., Zhao, H.: M2i: From factored marginal trajectory prediction to interactive prediction. In: CVPR (2022) 4
40. Sun, Q., Huang, X., Williams, B.C., Zhao, H.: Intersim: Interactive traffic simulation via explicit relation modeling. In: IROS (2022) 4
41. Suo, S., Regalado, S., Casas, S., Urtasun, R.: Trafficsim: Learning to simulate realistic multi-agent behaviors. In: CVPR (2021) 4
42. Toghi, B., Valiente, R., Sadigh, D., Pedarsani, R., Fallah, Y.P.: Altruistic maneuver planning for cooperative autonomous vehicles using multi-agent advantage actor-critic. In: CVPR 2021 Workshop on Autonomous Driving: Perception, Prediction and Planning (2021) 4
43. Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., et al.: A survey on large language model based autonomous agents. arXiv preprint arXiv:2308.11432 (2023) 4
44. Wang, P., Liu, Y., Chen, Z., Liu, L., Liu, Z., Komura, T., Theobalt, C., Wang, W.: F2-nerf: Fast neural radiance field training with free camera trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4150–4159 (2023) 5
45. Wang, X., Zhu, Z., Huang, G., Chen, X., Lu, J.: Drivedreamer: Towards real-world-driven world models for autonomous driving. arXiv preprint arXiv:2309.09777 (2023) 1, 4
46. Wei, Y., Wang, Z., Lu, Y., Xu, C., Liu, C., Zhao, H., Chen, S., Wang, Y.: Editable scene simulation for autonomous driving via collaborative llm-agents. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15077–15087 (2024) 2, 5
47. Wen, L., Cai, P., Fu, D., Mao, S., Li, Y.: Bringing diversity to autonomous vehicles: An interpretable multi-vehicle decision-making and planning framework. arXiv preprint arXiv:2302.06803 (2023) 3, 9
48. Wen, L., Fu, D., Mao, S., Cai, P., Dou, M., Li, Y.: Limsim: A long-term interactive multi-scenario traffic simulator. arXiv preprint arXiv:2307.06648 (2023) 3, 5, 9, 14
49. Wen, L., Fu, Z., Cai, P., Fu, D., Mao, S., Shi, B.: Trafficmcts: A closed-loop traffic flow generation framework with group-based monte carlo tree search. arXiv preprint arXiv:2308.12797 (2023) 3
50. Werling, M., Ziegler, J., Kammel, S., Thrun, S.: Optimal trajectory generation for dynamic street scenarios in a frenet frame. In: 2010 IEEE international conference on robotics and automation. pp. 987–993. IEEE (2010) 9
51. Witt, M., Kompaß, K., Wang, L., Kates, R., Mai, M., Prokop, G.: Driver profiling–data-based identification of driver behavior dimensions and affecting driver characteristics for multi-agent traffic simulation. Transportation research part F: traffic psychology and behaviour **64**, 361–376 (2019) 12
52. Wu, H., Phong, T., Yu, C., Cai, P., Zheng, S., Hsu, D.: What truly matters in trajectory prediction for autonomous driving? arXiv preprint arXiv:2306.15136 (2023) 4
53. Wu, Z., Liu, T., Luo, L., Zhong, Z., Chen, J., Xiao, H., Hou, C., Lou, H., Chen, Y., Yang, R., et al.: Mars: An instance-aware, modular and realistic simulator for autonomous driving. In: CAAI International Conference on Artificial Intelligence. pp. 3–15. Springer (2023) 5

54. Xu, R., Xia, X., Li, J., Li, H., Zhang, S., Tu, Z., Meng, Z., Xiang, H., Dong, X., Song, R., et al.: V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception. In: CVPR (2023) 4
55. Xue, Z., Peng, Z., Li, Q., Liu, Z., Zhou, B.: Guarded policy optimization with imperfect online demonstrations. arXiv preprint arXiv:2303.01728 (2023) 4
56. Yang, Z., Chen, Y., Wang, J., Manivasagam, S., Ma, W.C., Yang, A.J., Urtasun, R.: Unisim: A neural closed-loop sensor simulator. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1389–1399 (2023) 1, 5
57. Zhang, Q., Peng, Z., Zhou, B.: Learning to drive by watching youtube videos: Action-conditioned contrastive policy pretraining. In: ECCV (2022) 4
58. Zhang, Z., Liniger, A., Dai, D., Yu, F., Van Gool, L.: Trafficbots: Towards world models for autonomous driving simulation and motion prediction. arXiv preprint arXiv:2303.04116 (2023) 3, 4
59. Zhao, D., Huang, X., Peng, H., Lam, H., LeBlanc, D.J.: Accelerated evaluation of automated vehicles in car-following maneuvers. IEEE Transactions on Intelligent Transportation Systems **19**(3), 733–744 (2017) 5
60. Zhao, D., Lam, H., Peng, H., Bao, S., LeBlanc, D.J., Nobukawa, K., Pan, C.S.: Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques. IEEE transactions on intelligent transportation systems **18**(3), 595–607 (2016) 5
61. Zheng, O., Abdel-Aty, M., Yue, L., Abdelraouf, A., Wang, Z., Mahmoud, N.: Citysim: A drone-based vehicle trajectory dataset for safety oriented research and digital twins. arXiv preprint arXiv:2208.11036 (2022) 4
62. Zhou, M., Luo, J., Villella, J., Yang, Y., Rusu, D., Miao, J., Zhang, W., Alban, M., Fadakar, I., Chen, Z., et al.: Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving. arXiv preprint arXiv:2010.09776 (2020) 3