J Intel Connect Veh

Research Article

Your Ride, Your Rules: Psychology and Cognition Enabled Automated Driving Systems

Zhipeng Bao, Qianwen Li✉

Cite this article: https://doi.org/10.26599/JICV.2026.9210099

ABSTRACT: Current autonomous vehicles (AVs) primarily respond to external traffic conditions and treat humans as passive occupants, limiting their ability to personalize driving behavior and handle ambiguous scenarios that could benefit from occupant input. This study proposes PACE-ADS (Psychology and Cognition Enabled Automated Driving Systems), a human-centered autonomy framework that enables AVs to interpret and respond to both external traffic conditions and internal occupant states. PACE-ADS employs three collaborating foundation model agents: the Driver Agent interprets the external environment; the Psychologist Agent infers occupant state from passive psychological signals (e.g., facial expressions) and interprets active cognitive inputs (e.g., verbal commands); and the Coordinator Agent integrates these inputs to generate high-level driving behavior decisions. Operating at the low-frequency semantic planning layer, PACE-ADS complements existing AV stacks and activates selectively in response to changes in occupant state, cognitive instructions, or vehicle immobilization. Closed-loop CARLA simulations under prescribed occupant-state trajectories show that PACE-ADS adapts driving behavior along kinematic dimensions associated with ride comfort and supports recovery from operational immobilization through self-reasoning and occupant-in-the-loop guidance.

KEYWORDS: human-centric autonomy, large language model, human-machine interaction, personalization, closed-loop simulation

1 Introduction

Many automakers have already implemented various automation features in their mass-production vehicles, such as Ford, GM, and BMW. Common features include Adaptive Cruise Control (ACC), Automatic Lane Changing (ALC), and Lane Keeping Assistance (LKA). The global market for these technologies is projected to grow from $40.25 billion in 2024 to $75.23 billion by 2030, reflecting a strong compound annual growth rate (CAGR) of 13.83% (Strategic Market Research, 2022). While lower levels of automation have seen widespread deployment, higher levels (those capable of handling more complex driving tasks with minimal human input) have progressed much more slowly. So far, only Tesla has deployed a Full Self-Driving (FSD) system in mass-produced vehicles. Other companies, such as Baidu and Waymo, have focused on limited deployments through robotaxi services in cities like Beijing, Wuhan, Phoenix, San Francisco, and Los Angeles. The slower advancement of higher-level automation can be attributed to several factors. Real-world AV crash studies continue to identify safety-critical factors and the need for robust mitigation mechanisms (Channamallu et al., 2025). Beyond such technological and regulatory challenges, one of the most significant barriers is low public trust. Human-in-the-loop research on connected autonomous shuttles similarly reports that public acceptance and trust remain challenges and that stated intentions may differ from actual usage behavior (Xu & Zheng, 2024). The underlying reasons for this hesitation are explored in the following discussion.

First, existing automated driving systems (ADS), i.e., higher-level automation, lack the ability to personalize the riding experience and offer very limited transparency in their decision-making processes. Their behavior often appears opaque, leaving most riders uncertain about why the vehicle behaves in a particular way. These systems are primarily designed to optimize safety and efficiency based on external traffic conditions, frequently overlooking the diverse and evolving preferences of individual occupants. When an ADS's driving style significantly deviates from a rider's expectations, acceptance tends to decline (Schrum et al., 2024). Some progress has been made to address this limitation through the introduction of pre-defined driving styles. For example, Tesla's FSD system allows passengers to choose among "Chill," "Average," and "Assertive" modes. In academic research, human driving data have been used to classify three primary styles: conservative, normal, and aggressive, using methods such as Gaussian Mixture Models (Su et al., 2018), K-means clustering (Zhu et al., 2018; Han et al., 2023), Gaussian Process models (Wang et al., 2022), Artificial Neural Networks (Nava et al., 2019; Schrum et al., 2024), and other unsupervised learning techniques (Gao et al., 2020; Lee & Samuel, 2024). While most studies rely on vehicle trajectory data to infer driving styles, Ling et al. (2021) adopt a different approach by assessing driving styles through emotions detected from drivers' electroencephalogram (EEG) signals. They then use the Deep Deterministic Policy Gradient (DDPG) algorithm (Lillicrap et al., 2015) to adjust control parameters based on the identified driving style.

However, the varying nature of human needs cannot be fully captured by a limited set of pre-defined driving styles. To truly enhance the user experience, ADS must be individualized to account for the diverse and dynamic preferences of occupants. Some studies have attempted to address this challenge by collecting extensive data from individual drivers and using reinforcement learning or imitation learning to train ADS models that replicate specific driving styles (Chen et al., 2017; Bao et al., 2022; Tian et al., 2022; Li & Liu, 2023; Zhao et al., 2023). These approaches rely on offline training, after which the system is expected to mimic the target driver's behavior. Yet, human psychological states are inherently fluid, shaped by context, emotion, and situational demands. As a result, rider preferences can shift

School of Environmental, Civil, Agricultural & Mechanical Engineering, University of Georgia, Athens GA 30606, USA.

✉ Corresponding author. E-mail: Cami.Li@uga.edu

Received: March 10, 2026; Revised: August 11, 2026; Accepted: September 7, 2026

significantly even within the course of a single trip. For instance, a passenger might feel rushed at the beginning of their commute if they believe they are running late but become noticeably more relaxed once they realize they will still arrive in time for their meeting. Offline-trained models often fail to adapt to such real-time changes, resulting in less satisfying experiences and eroding public trust. To address this limitation, research is needed to develop personalized ADS capable of responding to the evolving needs of occupants. A promising direction is the real-time and continuous monitoring of human emotional states. Psychological signals (such as EEG signals and heart rate) and cognitive cues (such as facial expressions and verbal commands) are particularly valuable for this purpose, as they provide immediate feedback in response to external stimuli. Although these indicators have been widely used to detect driver fatigue and distraction (Doudou et al., 2020; Kashevnik et al., 2021; Zhang et al., 2022; Misra et al., 2023), their potential to inform vehicle control decisions in automated systems remains largely untapped.

Second, ADS struggles in rare or ambiguous traffic scenarios such as construction zones, informal right-of-way negotiations, or unpredictable pedestrian behavior. These situations often lead to the vehicle becoming immobilized (Schumann et al., 2023; Baldini et al., 2024; Honda et al., 2024; Schmittle et al., 2024). For example, in December 2024, a Waymo robotaxi in California became stuck in a roundabout, circling 37 times before engineers had to intervene remotely to resolve the issue (King, 2024). A similar incident occurred in Wuhan, China, where a Baidu robotaxi failed to replan its trajectory in a congested intersection. The vehicle remained stopped for over an hour, causing significant traffic disruption. Local traffic authorities were unable to assist, and control was eventually restored by remote engineers (New Tang Dynasty Television, 2024). Although remote intervention provides a potential backup solution, it presents several challenges. It requires continuous monitoring by trained personnel who must be available to respond quickly when the system encounters difficulties. This approach demands substantial human and financial resources, and it also relies on a low-latency connection to ensure timely and effective action. Some companies have developed ADS that allows passengers to take over when the system encounters a failure (Kondyli et al., 2023). For instance, Tesla's shadow mode (Harris, 2022) enables its FSD system to learn from human driving behavior in situations where the vehicle becomes immobilized and to reduce the likelihood of similar failures in the future. However, such solutions rely on the assumption that passengers are capable of taking control. In cases where riders are elderly, have disabilities, or are otherwise unable to take over, the ADS may remain stuck indefinitely. This highlights the need for more inclusive intervention strategies that can complement existing methods. One promising direction is to allow passengers to provide guidance through natural interactions, such as verbal commands, enabling them to assist the system without requiring physical control of the vehicle. Our earlier framework, StuckSolver (Bao & Li, 2025), operationalizes this idea. Results from the Bench2Drive benchmark (Jia et al., 2024) show that it can autonomously make recovery decisions, interpret passenger instructions, and execute appropriate behavioral plans to restore vehicle mobility.

With their advanced semantic understanding and reasoning capabilities, Large Language Models (LLMs) have emerged as a promising intermediary between occupants and ADS. Recent studies have begun exploring how LLMs can be integrated into ADS frameworks (Lüddecke & Ecker, 2022; Hu et al., 2023; Mao et al., 2023; Wang et al., 2023; Cui et al., 2024a; Xu et al., 2024; Fang et al., 2025). Related work also illustrates a broader property of language-model representations: human behavioral sequences can be encoded and learned. Liu et al. (2023), for example, model real-world delivery routes as language-like sequences to capture drivers' implicit route-selection knowledge. PromptTrack (Wu et al., 2025) enhances 3D object tracking and prediction in driving scenes by incorporating cross-modal features through a prompt reasoning branch. DiLu (Wen et al., 2024) combines LLMs with a reasoning and reflection module, allowing AVs to make decisions based on common-sense knowledge and improve generalization by accumulating driving experience over time. LLMs also show potential in vehicle motion control. For example, Sha et al. (2023) use LLMs to interpret driving scenes, make decisions, and convert these decisions into actionable driving commands through a parameter matrix adaptation process. Cui et al. (2024b) deploy a fine-tuned, lightweight vision-language model (VLM) on an automated vehicle, enabling it to adjust control parameters based on passenger feedback after each trip. Personalization has since become an explicit target. PADriver (Kou et al., 2025) reasons over a scene description together with a personalized prompt and an estimated danger level, but the prompt selects among three preset modes and so inherits the granularity limitation of production driving-style settings. Drive My Way (Wang et al., 2026) removes that limitation by learning a user embedding from real driving data and conditioning the policy on it, with natural language supplying short-term guidance; the embedding, however, is fitted per user in advance. Talk2Drive (Cui et al., 2024c) demonstrates the approach on a physical vehicle, translating spoken commands into control sequences and accumulating preferences in a memory module across rides. These systems establish that language models can personalize driving, but in each case the preference acted upon is supplied deliberately, as a prompt, a stored profile, or a spoken command, and is therefore fixed for the duration of a trip. While existing studies have made significant strides in integrating LLMs into ADS, most are evaluated using open-loop testing rather than real-time, closed-loop validation. This prevents the ADS from adapting to new inputs during the trip, including rider preferences and situational changes. Without feedback-driven learning, the system cannot refine its decisions dynamically, limiting its effectiveness in real-world, continuously evolving environments.

Motivated by these limitations, we propose PACE-ADS (Psychology and Cognition Enabled Automated Driving Systems), a human-centered, closed-loop autonomy framework. Unlike existing LLM-enabled ADS that mainly focus on external scene understanding or driving decision generation, PACE-ADS jointly incorporates traffic context, passive psychological signals, and active occupant instructions into vehicle decision-making. It employs three specialized foundation model agents: a Psychologist Agent that interprets the rider's psychological state and intent, a Driver Agent that analyzes the surrounding traffic environment, and a Coordinator Agent that integrates both sources of information to select driving behaviors, adjust operation parameters, and provide explanations. Operating at the semantic and behavioral planning layer, PACE-ADS complements rather than replaces conventional real-time control modules. Its main contributions are twofold: enabling vehicle-level closed-loop adaptation to changing occupant-state inputs for personalized driving, and supporting vehicle recovery from immobilizing scenarios through context-aware reasoning and occupant-in-the-loop guidance. Relative to our earlier framework StuckSolver (Bao et al., 2025), which addressed immobilization recovery alone using a single agent, PACE-ADS extends the same agentic reasoning to occupant-state-based personalization and distributes perception, occupant interpretation, and decision-making across three specialized agents.

The remainder of this manuscript is organized as follows. Section 2 introduces the PACE-ADS methodology. Section 3 details the simulation experiments conducted to evaluate the model's performance. Section 4 concludes the study and outlines directions for future research.

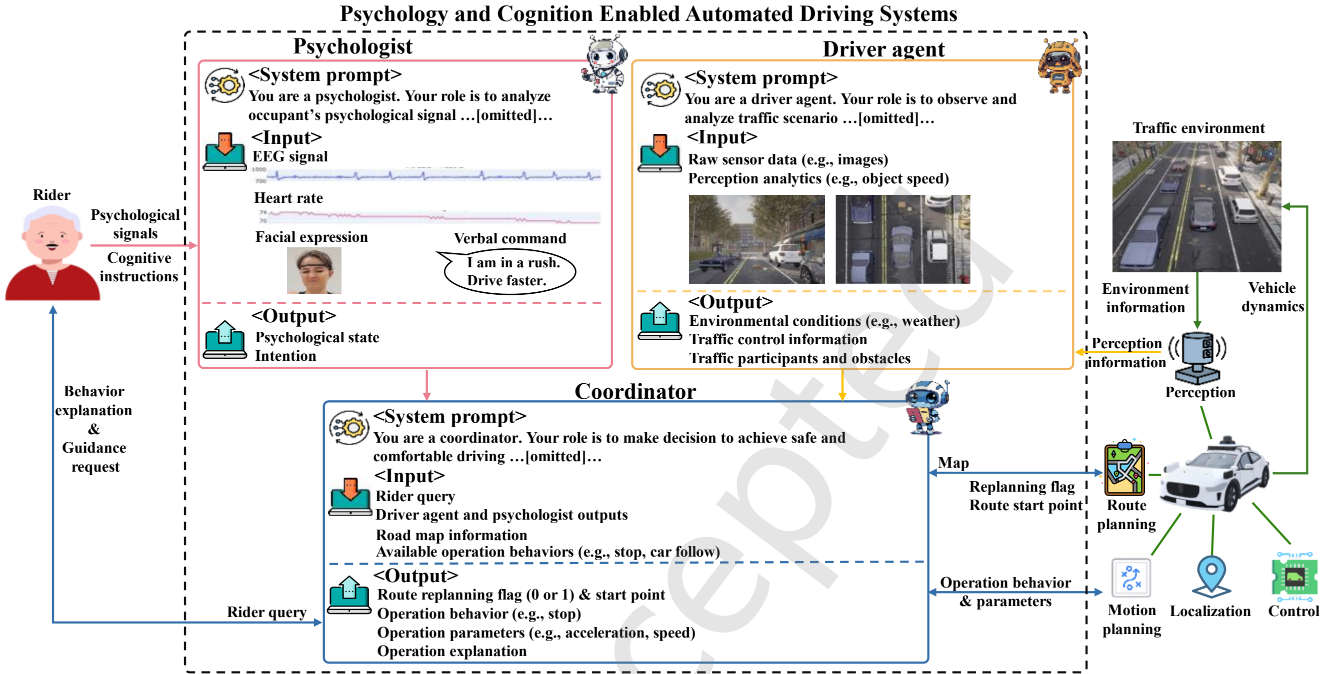


Fig. 1. An overview of Psychology and Cognition Enabled Automated Driving Systems.

2 Psychology and Cognition Enabled Automated Driving Systems

By enabling bidirectional human-machine communication through foundation models (Fig. 1), PACE-ADS personalizes the autonomous riding experience and improves vehicle responsiveness in immobilizing scenarios, with the aim of enhancing occupant comfort and trust and thereby promoting the adoption of automation technologies. PACE-ADS continuously monitors occupants' passive psychological signals (e.g., EEG, heart rate, facial expressions) and active cognitive instructions (e.g., verbal commands) to infer the occupant's state and intent. In response to such observations, the system dynamically adapts the ego vehicle's driving behavior in the direction associated with the inferred occupant state, for example by adjusting speed, acceleration, braking, or gap acceptance. It also provides behavior explanation using natural language to occupants upon request to enable operation transparency. When the AV becomes immobilized under ambiguous traffic scenarios such as unclear right of way, PACE-ADS leverages the occupant's cognitive instructions (e.g., verbal commands) to execute appropriate recovery solutions. PACE-ADS builds on LLMs given their strengths in semantic understanding, reasoning, and natural language interactions. By implementing an agentic workflow where three specialized agents collaborate to manage the complex autonomous driving, PACE-ADS achieves improved operation interpretability and system robustness. Leveraging system prompts, chain-of-thought (CoT) prompting (Wei et al., 2022), and function calling tools, each agent is guided to fulfill its designated role, execute assigned subtasks, and produce structured outputs.

A key strength of PACE-ADS is its architectural flexibility. The three agents can be configured with either the same or different models, depending on the data modality and task specialization. While this research explores a unified approach by using three LLMs to fulfill all roles, PACE-ADS is open to a hybrid configuration. For example, rule-based models and traditional lightweight machine learning models can be adopted to realize less demanding roles, such as emotion recognition, with a balance of solution quality and efficiency. In addition, PACE-ADS integrates with existing AV systems without requiring structural modifications, as it only receives input from AV modules and returns suggested operations in alignment with the AV module data format. While the AV modules handle high-frequency, low-level control tasks, PACE-ADS focuses on low-frequency, high-level behavioral decisions. It activates selectively in response to rider inputs and vehicle immobilization, addressing non-time-sensitive scenarios through closed-loop feedback that supports adaptive, personalized autonomy without disrupting real-time control.

While PACE-ADS enables adaptive autonomy, all personalization is bounded. Rider-specific adjustments are confined within pre-defined safety envelopes. These envelopes are aligned with traffic laws and social norms to ensure that behavior remains within legally and socially acceptable bounds. In the limiting case, a fully personalized AV may resemble a human-driven vehicle exhibiting user-preferred driving traits, such as assertiveness or caution, yet still operate under the same legal and infrastructural constraints as other road users. This mirrors today's traffic environments, where individuals drive in varied, but lawful, ways. Such personalization is intended to support rider comfort and trust while keeping behavior within legal, infrastructural, and pre-defined safety constraints.

2.1. Host AV Stack and PACE-ADS Interfaces

Existing AV frameworks consist of a set of modules that operate in coordination to enable automation. PACE-ADS is integrated with four of them. The description below focuses on the data that each module exchanges with PACE-ADS rather than on their internal algorithms, which are inherited from the host AV stack.

The perception module supplies three types of data to the driver agent: RGB images, object-level measurements, and ego vehicle state information. The RGB images include both front-view and bird-eye-view (BEV) perspectives (I_{front},I_{bev}). These images may be captured directly from onboard cameras or generated via sensor fusion and BEV map construction pipelines (Hao et al., 2025). The object measurement data $\boldsymbol{O}^p$ includes information for each surrounding object i: its class or identifier, relative distance, velocity, and spatial location O_i^p={id, $dist$, v, loc}. The ego vehicle state data includes the velocity and spatial location Ego={v, loc}, as detailed in Appendix Table A.1.

The route planning module computes the global route from the starting point to the destination as an ordered sequence of waypoints. We encapsulate the route planning function as an API, enabling the coordinator to invoke it when needed. The API accepts two parameters: a replanning flag δ and a new route start point P_{start}. The data transmitted from the route planning module to the coordinator includes the ego vehicle's current lane, adjacent lanes, and the set of available waypoints located ahead of the vehicle on these lanes, as outlined in the map M of Appendix Table A.1.

The motion planning module generates the vehicle's short-term trajectory along the assigned global route and manages a library of driving operations $\boldsymbol{A}_{lib}$, such as lane changing, car following, and stopping. Each supported driving operation in $\boldsymbol{A}_{lib}$ is documented with a functional description and a set of configurable parameters as well as default values (Fig. 2). The module operates using its internal decision rules when no external instruction is provided. However, when the coordinator issues an instruction (specifying the desired behavior a and associated parameters θ^*) and safety conditions allow, the module will follow the specified behavior.

```
CAR_FOLLOWING_TOOL = """
Description:  this  algorithm  is  based  on  the  Intelligent  Driver  Model(IDM).  It  can
control the vehicle to follow the leading vehicle.
Parameters:
- a: the maximum acceleration of the vehicle.
       Threshold: [0.3, 7.0] (m/s^2)
- b: the comfortable deceleration of the vehicle.
       Threshold: [0.5, 7.0] (m/s^2)
- T: the desired time headway to the vehicle ahead.
       Threshold: [0.8, 3.0] (s)
- v_0: the desired speed of the vehicle in free traffic.
       Threshold: [5, 40] (m/s)
Default values: a = 0.75, b = 1.67, T = 1.5, v_0 = 11.
"""

STOPPING_TOOL = """
Description: this algorithm can adjust your car's speed to stop.
Parameters: brake: control the brake of the vehicle. The value range is [0, 1], where 0
means no brake and 1 means full brake.
Default values: brake = 0.5
"""
```

Fig. 2. Example documentation of driving operation tools in $\boldsymbol{A}_{lib}$.

The control module executes the planned trajectory by generating low-level control commands, such as steering angle, throttle, and braking, using standard feedback control algorithms (e.g., Proportional-Integral-Derivative (PID) or Model Predictive Control (MPC)). It is inherited without modification and does not exchange data with PACE-ADS.

2.2. Driver Agent

Traditional perception modules are typically limited to low-level tasks such as object detection, classification, and tracking. These modules may misinterpret scenes and fail to capture the semantic relationships and interactions among objects (Jamali et al., 2025). The driver agent addresses this limitation by enabling high-level scene understanding (such as the effects of weather on driving behavior and the interactions among traffic participants) through semantic information extraction and commonsense reasoning $F^{driver}(\cdot)$. It receives front-view and BEV images (I_{front},I_{bev}) from the AV perception module. Guided by a system prompt S^{driver} and CoT prompting C^{driver}, the driver agent follows a step-by-step reasoning process (see Fig. 3) that includes extracting environmental conditions such as weather and road type, identifying traffic control elements like traffic signal state, traffic sign content, and work zones, and analyzing traffic participants and obstacles by determining their types, relative positions (e.g., on the left), and intentions. The driver agent then associates the resulting high-level insights with corresponding perception analytics ($\boldsymbol{O}^p$) from the perception module and enriches them with quantitative attributes such as distance, velocity, and lane ID to facilitate context-aware decision-making. The resulting structured output $\boldsymbol{Y}$ includes environmental conditions E, traffic control information T, traffic participants and obstacles information $\boldsymbol{O}$, ego vehicle state Ego, and a scene description D, as shown in the Eq. (1) below. This comprehensive scene understanding is passed to the coordinator.

$$\boldsymbol{Y} = \{E, T, \boldsymbol{O}, Ego, D\} = F^{driver}(I_{front}, I_{bev}, \boldsymbol{O}^p; S^{driver}, C^{driver}). \quad (1)$$

A detailed description of the input and output components is provided in Appendix Table A.1.

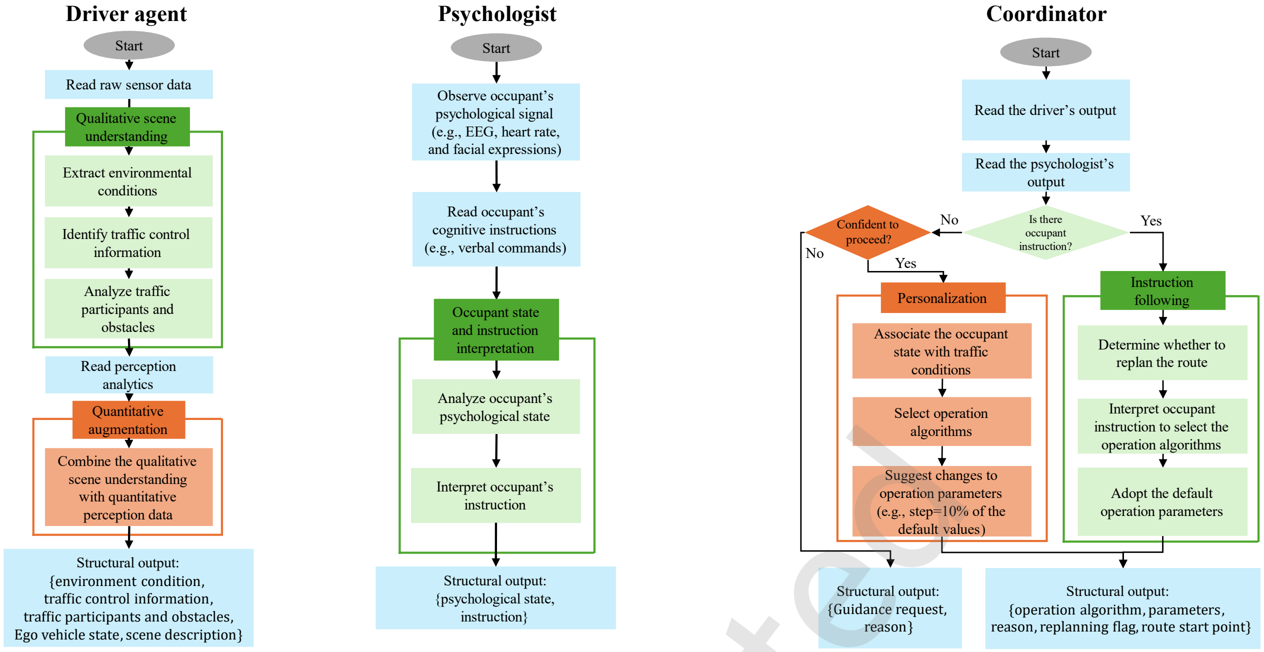


Fig. 3. A schematic illustration of the reasoning processes of the three agents.

2.3. Psychologist

The psychologist agent analyzes the occupant's passive psychological signals and active cognitive instructions by processing multimodal inputs, including EEG signals EEG, heart rate HR, facial expression images I_{face}, and verbal commands Ver. Guided by the system prompt S^{psy} and CoT prompting C^{psy}, the psychologist agent follows a step-by-step reasoning process (F^{psy}) to interpret these signals and synthesize a comprehensive understanding of the occupant's state and needs (see Fig. 3). The resulting output is denoted by R={$state$, $instr$}, where $state$ is the emotional state of the rider and $instr$ is the explicit rider commands. R is delivered to the coordinator for human-centric decision-making.

$$R=\{state,\ instr\} = F^{psy}(EEG, HR, I_{face}, Ver;\ S^{psy}, C^{psy}).\quad(2)$$

A detailed description of the input and output components is provided in Appendix Table A.1.

2.4. Coordinator

The coordinator is guided by the system prompt S^{coord} and CoT prompting C^{coord}, which defines its responsibilities and outlines the sequential procedures for decision-making. It prioritizes driving safety first, followed by personalization to accommodate the occupant's preferences. The coordinator reads the augmented traffic semantic information $\boldsymbol{Y}$ provided by the driver agent and references the psychologist's occupant report R. If no explicit occupant instruction is available, the coordinator interprets the occupant's state to select an appropriate operation behavior (a) from the vehicle motion planning module and sets the corresponding operation parameters (θ^*). When faced with uncertainty, the coordinator will actively seek guidance from the occupant, representing an enhancement over our previous work, StuckSolver (Bao & Li, 2025). When an occupant instruction is available, the coordinator enters instruction-following mode. It first determines whether route replanning is necessary. Then, using the instruction along with the augmented traffic semantic information $\boldsymbol{Y}$, the coordinator selects an appropriate operation behavior. If the instruction does not specify parameter values, the coordinator applies the default parameters associated with the selected behavior. Otherwise, the parameters are inferred from the instruction. The behavior library supplies each parameter's default value and permitted bounds; within those bounds the coordinator determines the value by reasoning over the occupant state and traffic context at inference time. No mapping from occupant states to parameter sets is defined, and no agent is fine-tuned. If route replanning is needed, the coordinator generates a replanning flag (δ, binary) and specifies a new route start point (P_{start}) based on the map M, then passes both to the vehicle route planning module. The overall decision-making workflow of the coordinator is illustrated in Fig. 3.

$$N=a,\ \theta^*,\ r,\ \delta,\ P_{start} = F^{coord}(\boldsymbol{Y}, R, M;\ S^{coord}, C^{coord}, \boldsymbol{A}_{lib}).\quad(3)$$

Here, N represents the coordinator's structured output, F^{coord} denotes the coordinator's reasoning process, $a \in \boldsymbol{A}_{lib}$ is the selected operation behavior from the behavior library $\boldsymbol{A}_{lib}$, offered by the vehicle motion planning module, and r is the explanation for the selected decision. A detailed description of the input and output components is provided in Appendix Table A.1.

The coordinator also passively detects potential vehicle immobilization by observing the outputs of the route planning and motion planning modules. Immobilization is suspected when the route planner indicates continued progress is expected, yet the motion planner shows repeated stops, failed trajectory generation, or prolonged stationary behavior without external justification (e.g., traffic signals or destination reached). When such inconsistencies persist, the coordinator flags a possible immobilization event and activates the driver agent to analyze the surrounding environment and guide further action.

3 Experiment

We design a set of scenarios to evaluate whether PACE-ADS adapts its driving behavior in the direction associated with prescribed occupant-state inputs. Additionally, we evaluate the system's ability to assist an immobilized AV by following verbal instructions from the occupant.

3.1. System Implementation Details

All three agents in PACE-ADS are built on GPT-4o-2024-08-06 (Hurst et al., 2024), which is equipped with multimodal processing capabilities, enabling the understanding and analysis of images and text. The model also supports structured output, facilitating seamless interaction and integration with existing AV modules. Each agent is encapsulated as a ROS2 (Humble version) node (Macenski et al., 2022) and connected to the CARLA (0.9.15 version) server through the carla-ros-bridge, forming a closed-loop simulation platform for system validation and testing. All simulations are conducted on a workstation running Ubuntu 22.04, equipped with an AMD Ryzen Threadripper 7960X (24 cores) CPU and two NVIDIA RTX 5000 Ada GPUs.

We build upon the Behavior Agent provided by the CARLA (Dosovitskiy et al., 2017), which integrates key components including the perception module, route planning module, and control module. Specifically, we enhance the perception module by equipping it with multiple sensors, including a front-view RGB camera, a top-down RGB camera, an Inertial Measurement Unit, a Global Navigation Satellite System, and a speedometer. These sensors enable the collection of comprehensive ego vehicle status and surrounding environmental data within the CARLA simulation environment. The route planning module in Behavior Agent generates a global route between a specified starting point and a target destination using the A* algorithm (Hart et al., 1972). The resulting global route is represented as an ordered sequence of waypoints along the road network, providing high-level navigation guidance for downstream modules. The control module unifies motion planning and low-level control. It interprets perception inputs and vehicle state information to determine appropriate driving behaviors and executes the corresponding control commands. The default parameter values of these baseline algorithms are listed in Appendix A.5.

3.2. Occupant Data

We use the publicly available Emognition dataset (Saganowski et al., 2022) to provide the passive psychological signals required by the psychologist agent. The dataset includes EEG signals, heart rate data, and facial expression images from 43 participants (see Fig. 4), along with annotated emotions such as anger, amusement, sadness, anxiousness, liking, relaxation, and fear. To align with prior studies on drivers' emotional states, we retain the categories relaxed and anxious, and remap anger to impatient. This adjustment reflects the behavioral and physiological similarity between anger and impatience in driving contexts, where frustration and a desire for quicker responses are more accurately described as impatience.

This remapping rests on two considerations. First, anger as it is measured in driving contexts is predominantly elicited by impedance to progress rather than by interpersonal hostility: the Driving Anger Scale (Deffenbacher et al., 1994) loads most heavily on situations such as slow driving, traffic obstruction, and blocked progress, which are described more precisely as impatience than as anger in its everyday sense. Second, anger and impatience occupy the same high-arousal, negative-valence region of the circumplex model of affect (Russell, 1980), and the peripheral and cortical signals available to the psychologist agent resolve position within this two-dimensional space but do not reliably separate the two states within that region. Because the coordinator consumes the occupant state solely to modulate speed, headway, and gap-acceptance parameters, quantities on which anger and impatience act identically, we adopt impatience as the label that more accurately names the state the system is designed to respond to.

To further examine the responsiveness of PACE-ADS to different levels of emotional intensity, we introduce two additional categories: very impatient and very anxious. Specifically, within the impatient and anxious categories derived from the Emognition dataset, we further distinguish intensity levels based on participants' self-reported emotion values (1 = not at all, 5 = extremely), as offered by the Emognition dataset. Samples with reported values greater than or equal to 4 are labeled as "very impatient" or "very anxious," while those with values below 4 remain labeled as "impatient" or "anxious." Importantly, our intent is not to establish a new taxonomy of passenger or driver types. Instead, our framework assumes that psychological states are fluid and continuous, and cannot be fully captured by a small set of fixed categories. The five emotional states we employ should therefore be understood not as an exhaustive classification, but as interpretive constructions that help us evaluate and communicate how PACE-ADS adapts its behavior in response to changing psychological inputs.

The psychologist agent receives the three passive modalities in the form the dataset provides, without feature engineering, filtering, or intermediate classification: EEG and heart rate are serialized to text and passed directly to the agent, and facial expression is passed as the corresponding RGB image. Verbal inputs from the occupant are first transcribed using Whisper (Radford et al., 2023), a speech-to-text model. The agent therefore performs the inference from raw signals to occupant state itself, rather than consuming the output of a separate emotion classifier.

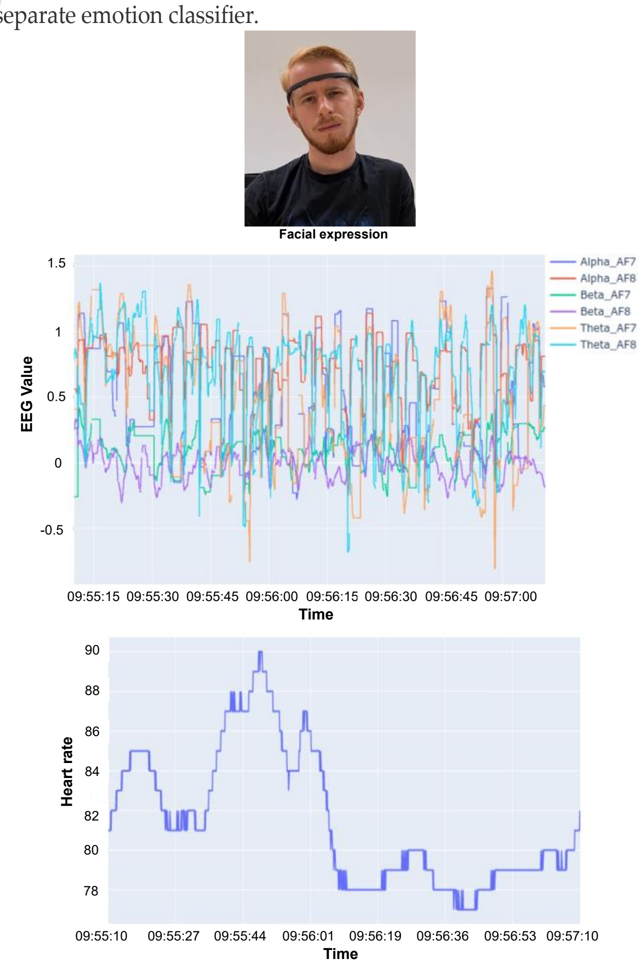


Fig. 4. Multimodal psychological signals of the same individual.

清華大学出版社 Tsinghua University Press | SciOpen

3.3. Personalization Tests

To evaluate the ability of PACE-ADS to personalize autonomous rides, we design four representative scenarios that capture diverse rider preferences: traffic signal interaction, pedestrian interaction, work zone navigation, and car-following. In each scenario, the occupant's psychological state is simulated by combining multimodal signals representing very impatient, impatient, relaxed, anxious, and very anxious states from the same individual across multiple test runs. Some runs use prescribed transitions among occupant-state inputs to represent plausible within-trip changes, while others use deliberately exaggerated prescribed transitions to rigorously assess system adaptability. This setup enables evaluation of whether PACE-ADS can infer and respond to a broad spectrum of psychological cues. Because occupant signals are injected on a prescribed schedule rather than generated by an occupant reacting to the vehicle, these runs test whether the system responds coherently to a given state trajectory; they do not establish that the resulting behavior improves how a real occupant feels. In this study, "closed-loop adaptation" therefore refers specifically to the simulated vehicle-level feedback loop: prescribed occupant-state inputs are processed by PACE-ADS to update high-level driving decisions, CARLA executes those decisions, and the resulting vehicle state is fed back to the framework. The occupant-state trajectory itself is externally prescribed and is not generated by this loop.

3.3.1 Traffic Signal Interaction

The ego vehicle starts at 40 km/h, approaching a red traffic light ahead. Once the vehicle comes to a complete stop, the signal immediately turns green, prompting the vehicle to resume driving. To evaluate the system's ability to balance occupant preferences with regulation requirements, a maximum speed of 60 km/h is imposed. At the beginning of each run, we inject psychological signals representing different emotional states to simulate how an occupant might respond to the AV's behavior. When the AV's behavior aligns with their preferences, signals corresponding to a relaxed state are injected. Otherwise, impatient and/or anxious signals are injected to reflect the rider's discomfort.

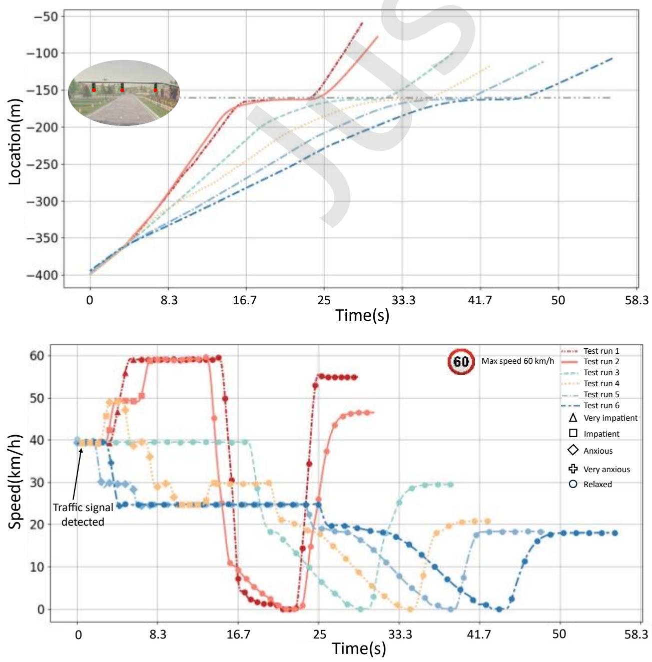


Fig. 5. Traffic signal interaction experiment results.

The experimental results, illustrated in Fig. 5, show that across different test runs, PACE-ADS infers an occupant-state category from the injected multimodal signals and adjusts the driving behavior accordingly. The system modifies cruising speed, deceleration timing, deceleration rate, and acceleration in a manner tailored to the occupant's state, with more assertive profiles under impatient inputs and more conservative profiles under anxious inputs.

Test run 3 serves as a benchmark, with relaxed signals injected throughout the simulation run as a constant relaxed-state input. In Test run 1, signals corresponding to a very impatient state are injected from the beginning as the prescribed initial condition. Once the vehicle accelerates significantly (in this case, toward the maximum permitted speed of 60 km/h), we inject relaxed signals to prescribe a transition to the relaxed state. Under the very impatient condition, the system interprets the input as indicating a stronger preference for faster travel and immediately accelerates to the maximum allowed speed of 60 km/h. During the red light interaction phase, PACE-ADS tailors its deceleration behavior based on previously observed emotional states. It infers a strong preference for maintaining speed and selects the latest possible deceleration point, applying a relatively high deceleration rate to ensure safety while aligning with the occupant's preferences. In the restart phase, the system uses memory of prior emotional states to inform acceleration. It applies higher acceleration rates of 5.4 m/s² and resumes at elevated cruising speeds of 55 km/h. Test run 2 follows the same structure as Test run 1, but starts with impatient signals instead. Once the vehicle reaches 60 km/h, relaxed signals are injected. In the impatient condition, the system determines that the vehicle is still a considerable distance from the red light and infers that the occupant's impatience stems from dissatisfaction with the current speed. In response, PACE-ADS gradually increases the cruising speed to 50 km/h. When it continues to infer an impatient state, it further raises the speed to 60 km/h. During the red light interaction phase, the system follows a similar deceleration pattern as Test run 1, but with a smaller deceleration. In the restart phase, it applies the acceleration rate of 4.2 m/s² and resumes at elevated cruising speeds of 45 km/h. In both Test runs 1 and 2, the vehicle remains within the posted speed limit while applying the state-conditioned speed and acceleration adjustments described above.

Test runs 5 and 6 follow a similar pattern to Test runs 1 and 2, using anxious and very anxious signals, respectively. For the anxious and very anxious conditions, the system adopts a comparable strategy in the opposite direction. It gradually reduces the cruising speed under these inputs. During the red light interaction phase, the system chooses an earlier and more gradual deceleration, producing a smoother braking profile. In the restart phase, it selects a low acceleration rate of 1 m/s² and a conservative cruising speed of 20 km/h, consistent with the injected state. In Test run 4, we simulate a more dynamic occupant state by injecting a sequence of emotional signals. The test begins with impatient signals. After the vehicle accelerates, anxious signals are injected to prescribe a transition to the anxious state. When the speed drops below 30 km/h, impatient signals are reintroduced to prescribe a transition back to the impatient state. Once the speed increases again, relaxed signals are injected to prescribe a transition to the relaxed state. Under the prescribed sequence with frequent state transitions, PACE-ADS updates its inferred state as the injected signals change and adapts its driving behavior accordingly. This responsiveness shows that the vehicle-level behavior follows the prescribed sequence of occupant-state inputs by adjusting speed and acceleration in the corresponding direction.

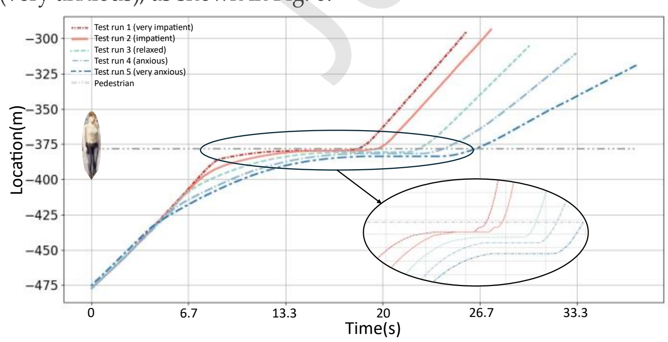
Each simulation scenario was repeated 30 times, varying the initial distance between the vehicle and the traffic light from 200 m to 300 m. The average results are summarized in Table 1, including the adjusted velocity (v_a) after emotional adaptation, the cruising velocity (v_c) after restart, the deceleration rate (b) when approaching the red light, and the acceleration rate (a) during restart. When the occupant is very impatient, the vehicle exhibits the highest speeds (v_a=59.6 km/h, v_c=55.7 km/h), while under the very anxious condition, both values are the lowest (v_a=26.3 km/h, v_c=19.3 km/h). Moreover, the very impatient state leads to the most aggressive acceleration (a=4.23 m/s²) and deceleration (b=5.51 m/s²) behaviors, whereas the very anxious state results in the smoothest and most conservative maneuvers (a=1.78 m/s², b=0.42 m/s²). A comparison with two baseline personalization methods under the same conditions is provided in Appendix A.6, Table A.6.

Table 1. Personalized evaluation results in the traffic signal interaction scenario (averaged over 30 runs).

Occupant state	Metrics			
	$v_a(km/h)$	$v_c(km/h)$	$b(m/s^2)$	$a(m/s^2)$
Very impatient	59.6 ± 0.7	55.7 ± 0.9	5.51 ± 0.28	4.23 ± 0.22
Impatient	55.3 ± 1.0	50.1 ± 1.1	4.63 ± 0.25	3.59 ± 0.20
Relaxed	40.0 ± 0.6	32.3 ± 0.8	1.02 ± 0.08	2.08 ± 0.12
Anxious	27.1 ± 0.8	19.6 ± 0.6	0.49 ± 0.05	1.86 ± 0.10
Very anxious	26.3 ± 0.7	19.3 ± 0.5	0.42 ± 0.04	1.78 ± 0.09
Note: Values are reported as mean ± SD over 30 runs.				

3.3.2 Pedestrian Interaction

The ego vehicle cruises at 40 km/h. A pedestrian stands ahead on the left side of the road, intending to cross the road. The pedestrian begins to cross after the vehicle comes to a complete stop. Once the pedestrian clears the conflict zone, the ego vehicle resumes driving. Five test runs are conducted, with each run initialized using a distinct psychological signal to emulate the occupant's emotional state. The test conditions are as follows: Test run 1 (very impatient), Test run 2 (impatient), Test run 3 (relaxed), Test run 4 (anxious), and Test run 5 (very anxious), as shown in Fig. 6.



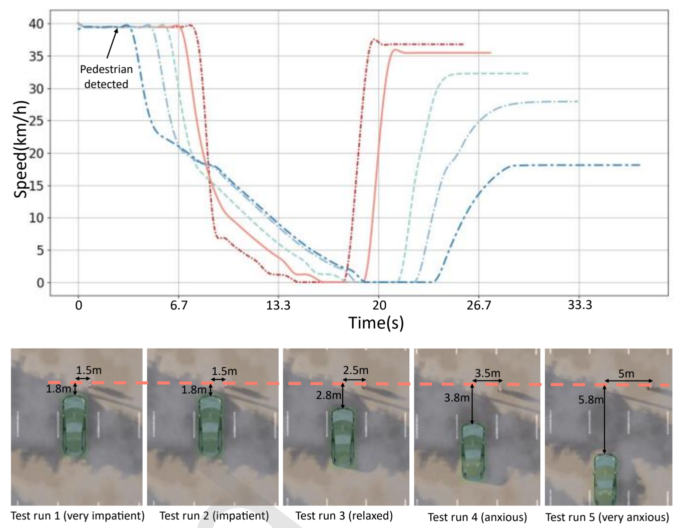


Fig. 6. Pedestrian interaction experiment results.

Upon pedestrian detection, the timing of deceleration varies based on the occupant's emotional state. The system initiates stopping earlier for more anxious occupants and later for impatient ones. The sequence of deceleration time, from earliest to latest, is very anxious > anxious > relaxed > impatient > very impatient. Deceleration rates reflect the level of aggressiveness associated with each emotional state. Impatient occupants prompt sharper deceleration, while anxious occupants lead to more gradual braking. The stopping distance and the restart timing, acceleration profile, and cruising speed after the pedestrian clears the path both vary with emotional state. In Test runs 1 and 2 (very impatient and impatient), the vehicle stops closest to the pedestrian at 1.8 meters and resumes motion when the pedestrian is just 1.5 meters away in the lateral direction. The vehicle exhibits a rapid acceleration phase, reaching 35 and 37 km/h with a 3.67 m/s² acceleration rate. In contrast, in Test runs 4 and 5 (anxious and very anxious), the vehicle maintains a larger buffer of 3.8 and 5.8 meters, indicating that PACE-ADS adopts a more conservative stopping strategy under higher anxiety. During restarting, the vehicle waits until the pedestrian moves more than 3.5 and 5 meters away. It accelerates gradually, reaching lower cruising speeds of 28 and 18 km/h with 1.1 and 0.74 m/s² acceleration rates, respectively.

Each simulation scenario was repeated 30 times, varying the initial distance between the vehicle and the pedestrian from 100 m to 200 m. The average results are summarized in Table 2, including the vehicle–pedestrian distance at the onset of deceleration (Δd), which measures how far the vehicle is from the pedestrian when it begins to slow down; the longitudinal distance to the pedestrian (d_{lon}) when the vehicle is stationary; and the lateral distance to the pedestrian (d_{lat}) when the vehicle resumes motion. To ensure pedestrian safety, the agents were instructed to maintain a minimum longitudinal and lateral distance of 1.5 m from pedestrians. PACE-ADS effectively accommodates passengers' personalized needs while consistently keeping both distances above the 1.5 m safety threshold, even under the most aggressive (very impatient) passenger state during stopping and restarting (d_{lon}=1.67 m, d_{lat}=1.62 m). Under more cautious emotional states, the system exhibits noticeably larger clearances with d_{lon}=4.14 m and d_{lat}=3.53 m for the anxious state, and d_{lon}=6.04 m and d_{lat}=5.25 m for the very anxious state. The corresponding baseline comparison is provided in Appendix A.6, Table A.7.

Table 2. Personalized evaluation results in the pedestrian interaction scenario (averaged over 30 runs, with cruising speed limited to 40 km/h).

Occupant state	Metrics		
	$\Delta d(m)$	$d_{lon}(m)$	$d_{lat}(m)$
Very impatient	24.3 ± 1.4	1.67 ± 0.05	1.62 ± 0.03
Impatient	35.4 ± 1.8	1.72 ± 0.06	1.68 ± 0.05
Relaxed	46.6 ± 2.1	2.93 ± 0.14	2.47 ± 0.12
Anxious	59.9 ± 2.7	4.14 ± 0.20	3.53 ± 0.18
Very anxious	78.8 ± 3.2	6.04 ± 0.27	5.25 ± 0.24
Note: Values are reported as mean ± SD over 30 runs.			

3.3.3 Work Zone Navigation

A work zone is placed ahead in the AV's current lane, prompting the vehicle to initiate a lane change to the left. Multiple vehicles are randomly positioned in the adjacent left lane, with varying gaps between them. At the start of the scenario, both the AV and surrounding vehicles travel at a constant speed of 40 km/h. We conduct five test runs, each with injected psychological signals representing the occupant's emotional state. The test conditions are as follows: Test run 1 (very impatient), Test run 2 (impatient), Test run 3 (relaxed), Test run 4 (anxious), and Test run 5 (very anxious).

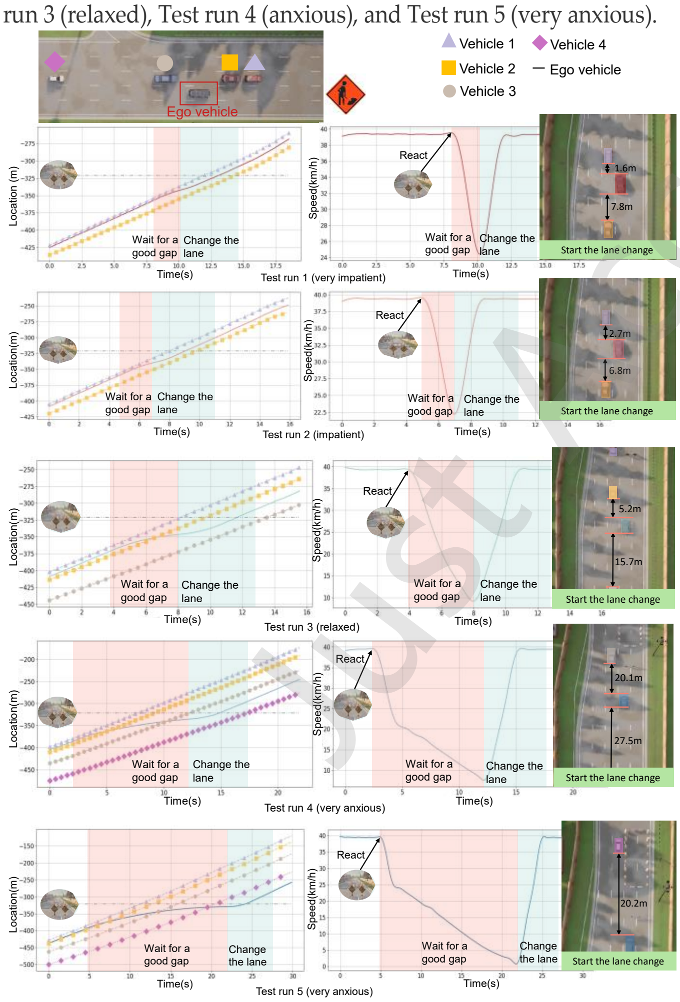


Fig. 7. Work zone navigation experiment results.

Fig. 7 highlights how PACE-ADS adapts its lane-changing behavior in response to the occupant's psychological state. When the occupant is very impatient (Test run 1), the AV initiates the lane change earliest, performing a quick and assertive maneuver into a relatively small front gap (1.6 m) while maintaining higher speeds throughout. In the impatient condition (Test run 2), the behavior remains proactive but slightly more restrained, with a lane-change speed of 22.5 km/h and a front gap of 2.7 meters. In both runs, the AV accepts the first available gap in the left lane between Vehicle 1 and Vehicle 2. In contrast, for the relaxed condition (Test run 3), the AV selects a lower lane-change speed of 10 km/h and a slightly longer front gap (5.2 m) between Vehicle 2 and Vehicle 3, resulting in a more gradual maneuver. In the anxious condition (Test run 4), the vehicle begins decelerating early and waits for a significantly larger gap of 20.1 meters before Vehicle 4. The lane-change speed is reduced to below 10 km/h. In the very anxious case (Test run 5), the AV decelerates to nearly a full stop, accompanied by earlier and smoother deceleration profiles. It delays the lane change until all four vehicles have passed and no following vehicle is present in the target lane, ensuring the largest possible gap of 20.2 m before initiating the maneuver.

Each test run was repeated 30 times using different random seeds, with vehicles in the adjacent lane generated at random spacings and with varying numbers. The average results are summarized in Table 3, including the distances to the rear and front vehicles ($dist_r$ and $dist_f$) and the time-to-collision value (TTC_r) for the following vehicle in the target lane when the ego vehicle initiates the lane change. As passenger anxiety level increases, both $dist_r$ and $dist_f$ grow significantly, while the TTC_r also increases from 1.77 s to 2.74 s, indicating a more cautious decision-making process. In particular, when the passenger is very anxious, the AV consistently delays its maneuver until no vehicles are present in the adjacent lane behind. This scenario is not included in the baseline comparison of Appendix A.6, because it requires a lane change and neither baseline method is able to perform one.

Table 3. Personalized evaluation results in the work zone navigation scenario (averaged over 30 runs, with cruising speed limited to 40 km/h).

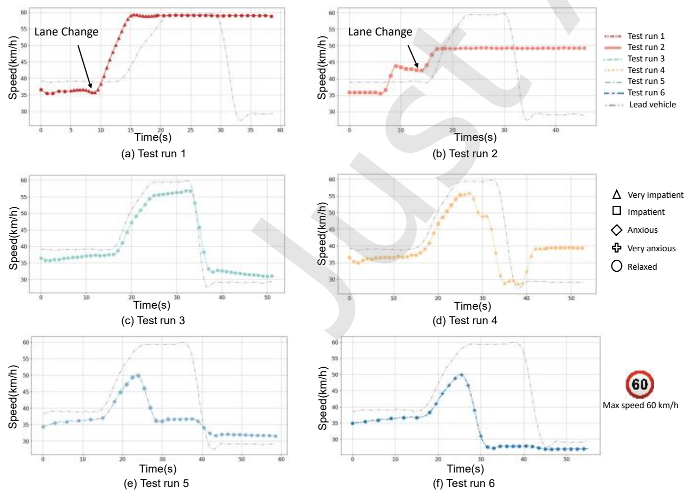
Occupant state	Metrics		
	$dist_r(m)$	$dist_f(m)$	$TTC_r(s)$
Very impatient	6.4 ± 0.5	1.52 ± 0.10	1.77 ± 0.08
Impatient	6.6 ± 0.6	2.65 ± 0.19	1.82 ± 0.09
Relaxed	18.8 ± 1.4	5.47 ± 0.38	2.03 ± 0.11
Anxious	28.8 ± 2.1	20.4 ± 1.6	2.74 ± 0.15
Very anxious	—	21.2 ± 1.7	—
Note: Values are reported as mean ± SD over 30 runs. A dash indicates that no following vehicle was present in the target lane, so the quantity is undefined.			

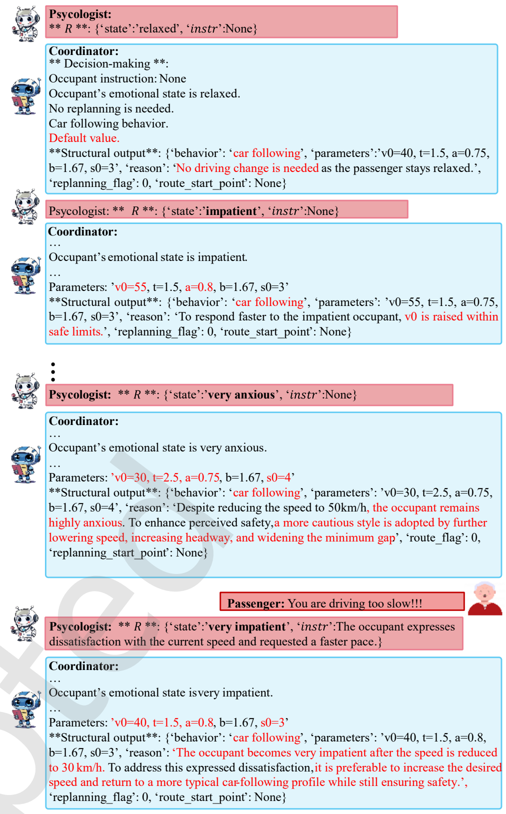
3.3.4 Car Following

The lead vehicle follows a predefined speed profile, which simulates traffic oscillation dynamics: it initially travels at 40 km/h, then accelerates to 60 km/h, and finally decelerates to 30 km/h. The initial distance between the ego vehicle and the lead vehicle is set to 10 meters.

Test run 3 (as shown in Fig. 8(c)), in which relaxed-state signals are prescribed throughout the simulation, PACE-ADS maintains its car following behavior, using the default parameters of the IDM to continuously track the lead vehicle. Test runs 1 and 2 are injected with signals corresponding to very impatient and impatient states, respectively, as the prescribed initial occupant-state inputs. Once the system increases speed noticeably, relaxed-state signals are injected to prescribe a subsequent state transition. Fig. 8(a) shows that when the injected signals are classified as very impatient in the scenario, the system switches from car following to a lane-changing maneuver, thereby producing a more assertive response associated with the very impatient category. It selects an adjacent lane with no surrounding traffic in proximity and accelerates rapidly to the maximum safe speed of 60 km/h. Once the prescribed input transitions to a relaxed state, the system continues to maintain the 60 km/h cruising speed. In Fig. 8(b), under the prescribed impatient input, PACE-ADS first attempts to increase the following speed to 45 km/h while monitoring subsequent injected signals. When it continues to infer an impatient state, the system initiates a lane change. After successfully changing lanes, the vehicle accelerates to 50 km/h. After the prescribed input transitions to a relaxed state, the system stabilizes at that cruising speed.

Test runs 5 and 6 follow the same pattern but represent anxious and very anxious emotional states. In Figs. 8(e) and 8(f), anxious and very anxious signals are prescribed, respectively, after the vehicle accelerates while following the lead vehicle. In both cases, PACE-ADS adjusts its behavior by reducing speed. In the anxious case (Fig. 8(e)), the system no longer follows the lead vehicle closely and performs a gradual deceleration—first returning to the initial following speed of 35 km/h, then further reducing to 25 km/h—demonstrating a progressive response associated with the prescribed anxious input. When the lead vehicle decelerates sharply to 30 km/h and the AV approaches at a higher speed, it further decelerates under IDM control to maintain a safe following distance. For the prescribed very anxious input (Fig. 8(f)), the system reacts more decisively, immediately reducing speed to 25 km/h. Once the prescribed input transitions to a relaxed state, the vehicle maintains the adjusted speed. Similarly, when the AV approaches the lead vehicle again, the IDM controller further reduces the speed to ensure a safe following distance.





(g) Agent collaboration process in Test run 4

Fig. 8. Car following experiment results.

In Test run 4, as shown in Fig. 8(d), the prescribed input sequence contains frequent transitions among state categories, and PACE-ADS continuously adjusts the vehicle's behavior in response to these transitions. Specifically, under an impatient input, the system increases speed, whereas under a very anxious input it reduces speed. Notably, when the AV decelerates to 30 km/h and the prescribed input transitions to very impatient, the system does not initiate an immediate lane change, as observed in Fig. 8(a). Instead, it chooses to accelerate to 40 km/h while remaining in the current lane. This decision reflects the system's assessment that although the lead vehicle is moving at a low speed, 30 km/h, the gap is sufficiently large, making it safe to accelerate while still producing a more assertive response under the very impatient input. The underlying reasoning and decision-making process of PACE-ADS in this scenario is detailed in Fig. 8(g).

Each test run was repeated 30 times with different random seeds, which introduced variability in the speed profile of the leading vehicle. This randomness affected both the timing and magnitude of its acceleration and deceleration. The vehicle's initial speed was set to 40 km/h, while its maximum and minimum speeds were limited to 60 km/h and 30 km/h, respectively. The average results are summarized in Table 4, including average speed ($\bar{v}$) over the entire car-following process, the average absolute acceleration ($|\bar{a}|$) during the acceleration phase, and the average absolute deceleration ($|\bar{b}|$) during the deceleration phase. The ego vehicle's average speed decreases progressively from very impatient (54.8 km/h) to very anxious (30.2 km/h), while the acceleration during the speeding phase becomes increasingly smoother (from 1.39 m/s^2 to 0.38 m/s^2). For the deceleration phase, there are no values for the very impatient and impatient states because the PACE-ADS, after accelerating to a higher speed, chose to change lanes and cruise instead of slowing down. In contrast, for the very anxious condition, the average deceleration (1.39 m/s^2) is greater than that in the relaxed and anxious states; this reflects a comparatively rapid speed reduction selected under the very anxious input, consistent with the response illustrated in Fig. 8(f). The corresponding baseline comparison is provided in Appendix A.6, Table A.8.

Table 4. Personalized evaluation results in the car following scenario (averaged over 30 runs).

Occupant state	Metrics		
	$\bar{v}(km/h)$	$\mid \bar{a} \mid$ (m/s^2)	$\mid \bar{b} \mid$ (m/s^2)
Very impatient	54.8 ± 0.8	1.39 ± 0.08	—
Impatient	46.1 ± 1.0	0.91 ± 0.06	—
Relaxed	42.5 ± 1.1	0.56 ± 0.04	0.98 ± 0.07
Anxious	34.4 ± 0.9	0.39 ± 0.03	0.43 ± 0.04
Very anxious	30.2 ± 0.6	0.38 ± 0.03	1.39 ± 0.09

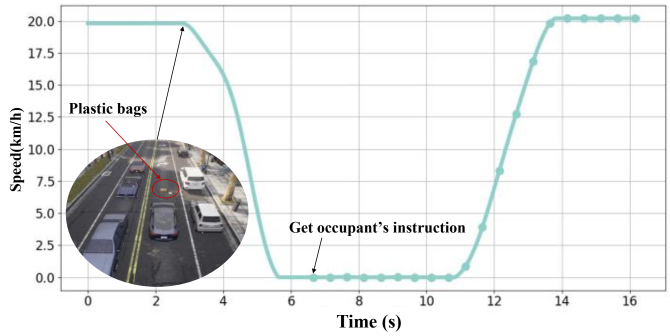
Note: Values are reported as mean ± SD over 30 runs. The leading vehicle's average speed, acceleration, and deceleration are 45.7 km/h, 0.72 m/s², and 1.17 m/s² respectively. A dash indicates that no deceleration phase occurred, because the vehicle changed lanes and cruised instead of slowing down.

3.4. Immobilization Recovery

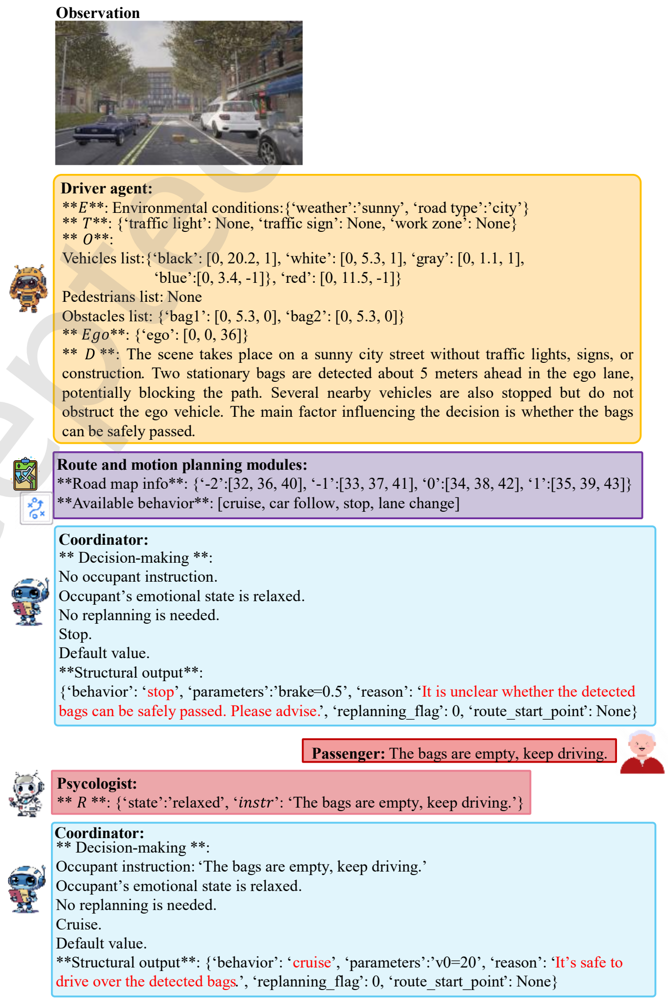
In this section, we design three additional scenarios to evaluate how PACE-ADS recovers an immobilized AV by selecting appropriate driving behaviors based on occupant instructions, thereby reducing travel delays and improving traffic efficiency. Although the Bench2Drive benchmark covers a wide range of driving tasks, it does not include sufficient edge cases that challenge recovery and decision-making capabilities. The three scenarios introduced here are not meant to be comprehensive but are representative cases that allow assessment of how PACE-ADS interprets ambiguous environments, makes decisions under uncertainty, and interacts with the occupant when necessary.

3.4.1 Static Obstacles Scenario

The ego AV cruises along a street in CARLA's city map at a speed of 20 km/h, with several vehicles parked on the right shoulder. Two paper bags are detected in the AV's lane about 5 meters ahead. Upon detection, the AV initiates a stop behavior to avoid a potential collision, as shown by the green curve without markers in Fig. 9. After stopping, the coordinator senses that the AV has halted unexpectedly, even though the destination has not been reached and no nearby traffic control devices require stopping. The coordinator activates the driver agent to analyze the surrounding traffic environment. The driver agent correctly identifies the objects and reports this information to the coordinator. However, due to insufficient information about the contents of these bags, the coordinator cannot determine whether to drive over or not. To prioritize safety, it conservatively treats the bags as a potential hazard and halts the AV. The coordinator then communicates the reason for stopping to the passenger and seeks guidance. After the passenger confirms that the bags are empty and pose no danger, PACE-ADS overrides the initial stop behavior and resumes operation, as shown by the green curve with markers in Fig. 9. This scenario illustrates PACE-ADS's ability to detect and analyze immobilized AV behavior and to proactively involve the passenger when uncertainty arises.



(a) AV's speed profile.



(b) PACE-ADS reasoning outputs.

Fig. 9. Static obstacles experiment results.

3.4.2 Roundabout Scenario

The AV system's route planning module is forced to generate a path that continuously loops a roundabout, simulating an immobilization case similar to the Waymo Robotaxi incident. This behavior is illustrated by the green curve without markers in Fig. 10.

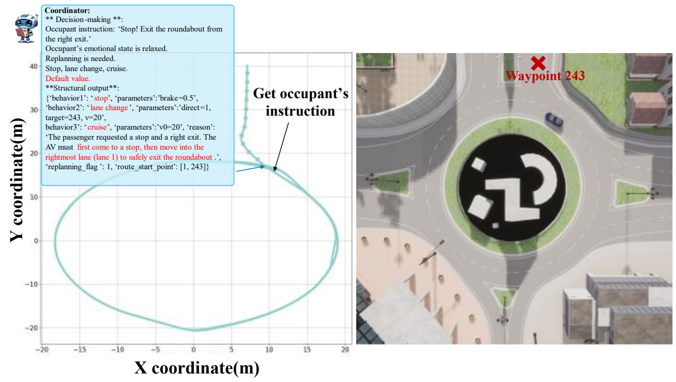


Fig. 10. Roundabout experiment results.

After the vehicle completes its fifth loop, passenger guidance is issued to PACE-ADS: "Stop! Exit the roundabout from the right exit." Upon receiving this instruction, the coordinator re-analyzes the situation. It parses the guidance into actionable components, including stop, lane change, and cruise, and performs route replanning. A new route starting point is marked by the red cross in Fig. 10. Following this decision, the AV successfully exits the roundabout, shown by the green curve with markers in Fig. 10. This scenario demonstrates the capability of PACE-ADS to interpret passenger commands. It extracts the intended meaning, decomposes the instruction into semantic actions, matches them with available driving behaviors, and generates a coherent sequence of decision-making steps to assist the AV in resuming normal operations.

3.4.3. Road Closure Scenario

We design a road closure scenario in which the AV travels from the origin (black triangle) to the destination (red cross), as shown in Fig. 11. While following the planned route (black dashed curve), the AV detects obstacles ahead and decides to stop. After stopping, the coordinator detects that the AV has stopped unexpectedly, despite not having reached its destination and with no nearby traffic signals or stop signs. In response, the coordinator activates the driver agent to analyze the surrounding traffic environment.

The driver agent detects that barricades fully block all travel lanes. Based on this input, the coordinator decides to maintain the stopping state and prompts the passenger to confirm whether a new route should be planned with an explanation. Upon receiving passenger confirmation, the coordinator queries the map data from the route planning module and selects waypoint 430 in the left lane (marked by the red triangle) as the new starting point for replanning. The newly generated route to the destination is shown as a red solid line in Fig. 11. This scenario illustrates PACE-ADS's capabilities in perceiving and interpreting the driving environment, providing explanations for vehicle behavior, and proactively engaging the passenger in decision-making.

3.4.4. Quantitative Recovery Evaluation

Each scenario was repeated 30 times, and the results are reported in Appendix Table A.4. The CARLA Behavior Agent recovers in none of the 90 runs, confirming that these scenarios lie outside what the conventional stack resolves. PACE-ADS recovers in 82 of 90 runs (91.1%), against 75 (83.3%) for StuckSolver (Bao & Li, 2025), with the gain concentrated in the two scenarios that require situational interpretation: the roundabout improves from 70.0% to 93.3% and road closure from 83.3% to 100%. We attribute this to the division of labor across agents, which narrows the context each agent must reason over and yields more reliable outputs than a single agent handling perception, interpretation, and decision together. The cost is latency: recovery takes 2.17 s on average against 1.54 s for StuckSolver, since the additional reasoning steps are sequential.

The static-obstacle scenario runs in the opposite direction, from 96.7% to 80.0%, and the reason is instructive. The six unsuccessful runs are cases in which the coordinator, having consulted the driver agent's assessment of the obstacle, declined to act on the occupant's instruction to drive over the bags. These are refusals rather than failures of recovery, and they reflect the additional verification step that PACE-ADS interposes between an occupant instruction and its execution. Whether a refusal is the correct outcome depends on what the obstacle is, which the framework cannot always determine; the same mechanism that prevents an unsafe instruction from being executed will sometimes reject a safe one.

This mechanism is nonetheless incomplete. An occupant instruction is checked against the driver agent's scene assessment, and instructions inconsistent with it are refused, but the check is a reasoning step rather than a formal guarantee, and an instruction that is unsafe for reasons the perception module does not surface may still be accepted. Occupant guidance is therefore best regarded as a source of information that the system weighs, not as an authority it obeys. Establishing a verifiable safety layer over occupant instructions remains open, and we identify it as future work.

3.5. Cross-Subject Generalizability of Psychological State Recognition

To assess whether the psychologist agent generalizes beyond the single individual used in Section 3.3, we evaluate it on all 43 Emognition participants, drawing one sample per state from each participant (215 samples) and applying the same agent, system prompt, model version, and input-processing pipeline to every participant without any participant-specific adaptation or fine-tuning. The agent attains an overall accuracy of 64.2% with a macro-averaged F1 of 64.3% (Appendix Table A.2). The structure of the residual error matters more than its magnitude: 36 of the 77 misclassifications (46.8%) fall between adjacent intensity levels within the same valence family and 31 (40.3%) involve the relaxed state, while only 10 (4.7% of all samples) invert valence between the impatient and anxious families. Recognition error therefore propagates to the vehicle mainly as an attenuation of the adaptation magnitude rather than a reversal of its direction. A modality ablation over the same samples (Appendix Table A.3) shows that no single channel is sufficient, 44.7% for heart rate alone and 57.2% for facial expression alone, and that the modalities are complementary; performance degrades gracefully under partial input, with the EEG-free configuration losing only 3.7 percentage points, which supports the architectural flexibility described in Section 2.

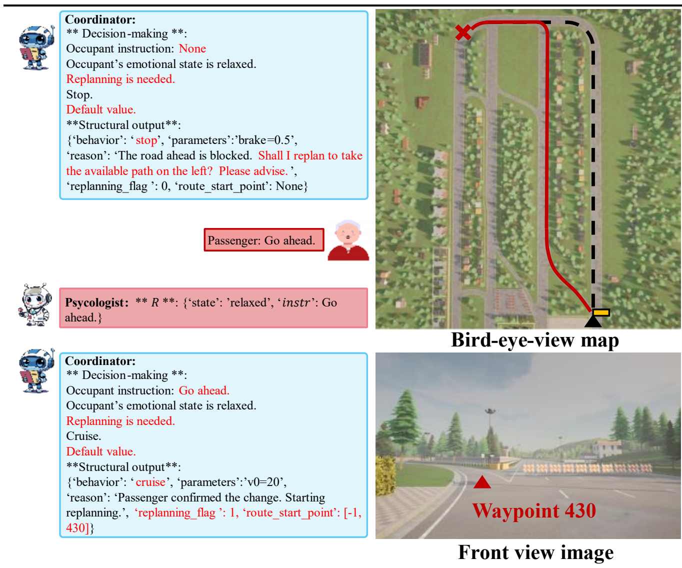


Fig. 11. Road closure experiment results.

4 Conclusions

This paper proposes PACE-ADS, a novel framework for human-centered autonomy that integrates LLMs to support adaptive and interpretable driving behavior. Simulation results demonstrate that PACE-ADS adapts its driving behavior in response to prescribed occupant-state inputs and situational contexts, adjusting speed, acceleration, braking, and gap acceptance in the corresponding direction while maintaining the imposed safety constraints. Relative to the evaluated rule-based comparator, which maps the five occupant states to three predefined CARLA driving profiles, PACE-ADS exhibits finer-grained adaptation by producing distinct driving responses for very impatient versus impatient inputs and for very anxious versus anxious inputs. This comparison is specific to the evaluated three-profile baseline and does not establish a general advantage over predefined personalization approaches. Furthermore, in scenarios where the AV becomes immobilized, PACE-ADS is capable of resolving the situation by following natural language guidance provided by the occupant. PACE-ADS redefines the occupant's role from passive observer to active participant. By inferring occupant-state categories from multimodal signals and engaging the occupant during ambiguous situations, it is designed to reduce discomfort, confusion, and risk in non-standard driving scenarios where rigid automation often falls short. Interpretability through natural language dialogue and responsiveness to individual preferences are mechanisms that prior work associates with user trust, perceived control, and acceptance; whether PACE-ADS realizes these benefits remain to be verified in the human-subject evaluation identified below. Moreover, PACE-ADS is modular and compatible with existing AV stacks, enabling seamless integration without major architectural changes. This makes it a scalable enhancement for next-generation AVs.

These results, however, are obtained entirely in simulation. CARLA supplies clean, fully observed perception and deterministic actuation, and its traffic participants follow scripted rather than genuinely interactive behavior, so the present experiments isolate the framework's decision-making from the sensing noise, detection failures, and control latency that a physical vehicle encounters. Future research will focus on deploying the PACE-ADS framework on physical AV platforms, where the framework must operate on degraded and occasionally erroneous perception rather than on ground-truth scene descriptions, and where the semantic decisions it issues must remain valid over the interval required to produce them. In the present study, psychological signals are injected on a prescribed schedule rather than sensed from an occupant reacting to the vehicle's own behavior and they derive from a dataset in which emotions were elicited by film clips in a laboratory rather than during driving, so the signal-to-state relationship used here may not transfer unchanged to on-road occupants; future evaluation will therefore acquire occupant signals on-line, alongside post-drive interviews to assess perceived personalization, comfort, and trust. A full invocation of the three agents currently takes 2.2 s on average from trigger to decision. This is workable at the low-frequency semantic layer, since real-time control remains with the existing stack, but it relies on remote API access. Lightweight LLM architecture optimized for local, on-board deployment will therefore be explored. Techniques such as model distillation, pruning, and quantization will be employed to enable low-latency inference while preserving the interpretability and responsiveness required for human-centered autonomy. To enhance long-term adaptability, future development will also incorporate a memory module into PACE-ADS, enabling the system to learn from user interactions over time. Such memory-enabled architecture will allow the LLM to retain contextual information across sessions, improving consistency in decision-making and supporting user-specific behavioral adaptation, particularly important in private vehicle applications.

Appendix

A.1. Agent input and output

Table A.1. Description of the input and output data for the driver agent, psychologist, and coordinator.

Module	Input/ Output	Data	Example	Description
Driver agent	Input	I_{front}, I_{bev}	RGB images	Captured by front view camera and BEV camera.
		$\boldsymbol{O^p}$={…O_i^p={*id*, *dist*, *v*, *loc*}…}	{…, O_i^p= {'black vehicle', 20m, 35km/h, [1, 243]}, …}	Analytics from the vehicle's perception module. The object measurement data $\boldsymbol{O^p}$ includes information for each surrounding object *i*: its class or identifier, relative distance, velocity, and

				spatial location (current lane id and waypoint id).
	Output	*E*={‘weather’: …,‘road type’:…}	{‘weather’:’sunny’, ‘road type’:’city’}	Weather and road type affect driving speed.
		T={‘traffic light’: …, ‘traffic sign’: …, ‘work zone’: …}	{‘traffic light’:[red, (10,20)], ‘traffic sign’: ‘stop’, ‘work zone’: [(30,10), affect?]}	Traffic light’s state and traffic sign’s content are observed from image. The coordinates of traffic lights and work zone are obtained from perception module. The affect? represents whether the work zone affects AV’s driving.
		O={Vehicles list: { ‘id’: [speed, distance, lane id], …}, Pedestrians list:{ ‘id’: [speed, distance, lane id], …}, Obstacles list:{ ‘id’: [speed, distance, lane id], …}}	Vehicles list: {‘black’: [35km/h, 20m, 0], …}, Pedestrians list: {‘red’: [4km/h, 10m, 1], …} Obstacles list: {‘barrier’: [0km/h, 6m, -1], …}	‘black’ represents the vehicle’s color, which is also the vehicle’s role name offered by perception module. The speed, distance, lane id are obtained from perception module. The format of data in pedestrians list is the same as vehicles list. ‘barrier’ is the role name of the obstacle offered by perception module.
		Ego={*v*, *loc*}	{45km/h, [0, 243]}	AV’s velocity, spatial location (current lane id and waypoint id).
		D	The scene is on a sunny city street with no traffic lights, signs, or construction. Two stationary bags are detected about 5.3 meters ahead in the ego lane, possibly blocking the path. Several nearby vehicles are also stationary, but they do not directly interfere with the ego vehicle.	A scene description emphasizing traffic semantics, focusing on how environmental conditions and traffic control elements affect driving, and describing traffic participants and obstacles, noting their interaction with the ego vehicle and potential impact.
Psychologist	Input	*EEG*={ $Alpha_{AF7}$:[…], $Alpha_{AF8}$:[…], $Beta_{AF7}$:[…], $Beta_{AF8}$:[…], $Theta_{AF7}$:[…], $Theta_{AF8}$:[…]}	{[0.0864, …], [0.6023, …], [0.3069, …], [0.4573, …], [0.1741, …], [0.5625, …]}	EEG signals and heart rate are extracted from the Emognition dataset. They are time-synchronized and collectively respond to the same type of interest.
		HR	[68, 68, 69, 69, 69, 70, …]	
		I_{face}	RGB image	Images are extracted from videos offered by Emognition dataset, and together with EEG and heart rate signals, reflect the same emotional state.
		Ver	‘The bags are empty, keep driving.’	Occupant’s verbal commands.
	Output	*R*={*state*, *instr*}	{‘state’: ‘relaxed’, ‘instr’:’None’}	The *state* is the passenger’s emotions, such as impatience, relaxation, or very anxiety. *instr* is occupant’s instructions.
Coordinator	Input	***Y***={*E*,*T*,***O***,*Ego*, *D*}	{E = {…}, T = {…}, O = {…}, Ego = {…}, D = {…}}	Structural output from driver agent.
		R={*state*, *instr*}	{‘state’: ‘relaxed’, ‘instr’:’None’}	Structural output from psychologist.
		M={’left lane id’: [waypoint id], ’current lane id’: [waypoint id], ’right lane id’: …}	{’-1’: [17, 18, 19], ’ 0’: [21, 22, 23], ’ 1’: [25, 26, 27]}	The AV’s current lane id is 0, with negative values representing left lanes and positive values representing right lanes.
		$\boldsymbol{A}_{lib}$={{ ‘behavior1’s introduction’: …, ‘parameters	{‘behavior1’s introduction’: …, ‘parameters description’: There are four parameters…, ‘default value’: v0=40, …}	$\boldsymbol{A}_{lib}$ includes an introduction to the behavior’ s functions, the parameters associated with the algorithm implementing the

		description': …, 'default value': …}, …}		behavior, and the corresponding default parameter values.
	Output	*N*={'behavior':…, 'parameters':…, 'reason':…, 'replanning flag':…, 'route start point': [lane id, wp id]}	{'behavior': 'car following', 'parameters': 'v0=40, t=1.5, a=0.8, b=1.67, s0=3', 'reason': '…', 'replanning_flag': 0, 'route_start_point': [0, 243]}	The structural output includes the selected behavior, parameter values, and the reasoning behind the decision. The replanning_flag is set to 0 or 1, indicating whether replanning is needed. lane id and wp id represent the lane and waypoint where the replanning starts.

A.2. Cross-subject psychological state recognition performance

Table A.2. Cross-subject psychological state recognition performance of the psychologist agent across all 43 Emognition participants (one sample per state per participant; 215 samples).

Ground-truth state	Predicted state					Metrics			Support
	VI	**I**	**R**	**A**	**VA**	**Precision (%)**	**Recall (%)**	**F1 (%)**	
Very impatient	**29**	10	2	1	1	70.7	67.4	69.0	43
Impatient	8	**27**	5	2	1	60.0	62.8	61.4	43
Relaxed	2	5	**28**	6	2	63.6	65.1	64.4	43
Anxious	1	2	6	**26**	8	57.8	60.5	59.1	43
Very anxious	1	1	3	10	**28**	70.0	65.1	67.5	43
Macro average	—	—	—	—	—	**64.4**	**64.2**	**64.3**	**215**

Note: Support denotes the number of ground-truth samples in each class; one sample per state from each of the 43 participants yields a support of 43 for every class. Overall accuracy is 138/215 = 64.2%. Bold diagonal entries are correct predictions. Of the 77 misclassifications, 36 (46.8%) occur between adjacent intensity levels within the same valence family, 31 (40.3%) involve the relaxed state, and 10 (4.7% of all 215 samples) invert valence between the impatient and anxious families. All participants are evaluated with the same agent, system prompt, model version, and input-processing pipeline; no participant-specific prompt adaptation, threshold calibration, or fine-tuning is applied.

A.3. Modality ablation for cross-subject psychological state recognition

Table A.3. Modality ablation for cross-subject psychological state recognition, evaluated on the same 215 samples.

Input configuration	EEG	Heart rate	Facial expression	Correct / Total	Accuracy (%)	Change from all modalities (pp)
EEG only	✓	—	—	113 / 215	52.6	−11.6
HR only	—	✓	—	96 / 215	44.7	−19.5
Face only	—	—	✓	123 / 215	57.2	−7.0
EEG + HR	✓	✓	—	128 / 215	59.5	−4.7
EEG + Face	✓	—	✓	133 / 215	61.9	−2.3
HR + Face	—	✓	✓	130 / 215	60.5	−3.7
All modalities	✓	✓	✓	**138 / 215**	**64.2**	—

Note: Change from all modalities indicates the difference in recognition accuracy relative to the full-modality configuration (EEG + heart rate + facial expression), measured in percentage points (pp).

A.4. Immobilization recovery performance

Table A.4. Immobilization recovery performance over 30 runs per scenario.

Scenario	Method	Successful runs (out of 30)	Success rate (%)	Recovery time (s)
Static obstacles	CARLA Behavior Agent	0	0	—
	StuckSolver	29	96.7	1.50 ± 0.31
	PACE-ADS	**24**	**80.0**	**2.10 ± 0.43**
Roundabout	CARLA Behavior Agent	0	0	—
	StuckSolver	21	70.0	1.62 ± 0.23
	PACE-ADS	**28**	**93.3**	**2.30 ± 0.44**
Road closure	CARLA Behavior Agent	0	0	—

	StuckSolver	25	83.3	1.50 ± 0.27
	PACE-ADS	**30**	**100.0**	**2.10 ± 0.21**
All scenarios	**CARLA Behavior Agent**	**0**	**0**	**—**
	StuckSolver	**75**	**83.3**	**1.54**
	PACE-ADS	**82**	**91.1**	**2.17**
Note: Static obstacles and roundabout use initial speeds sampled uniformly from 20–40 km/h; road closure uses different road maps. Recovery time is the mean ± SD over successful runs. The aggregate rows pool all 90 runs; aggregate recovery time is the unweighted mean of the three scenario means.				

A.5. Baseline driving algorithms and agent invocation regime

Car-following and cruising rely on the Intelligent Driver Model (IDM) (Kesting et al., 2010), with default parameters that include a desired velocity of 11 m/s, a safe time headway of 1.5 s, a maximum acceleration of 0.75 m/s², a comfortable deceleration of 1.67 m/s², an acceleration exponent of 4, and a minimum distance of 1.5 m. Stopping behavior is controlled by directly adjusting the ego vehicle's brake pedal input in CARLA, where the brake intensity ranges from 0 to 1, and the default brake value is set to 0.5. Lane changing decision-making is governed by a gap–TTC decision model, in which the model evaluates both the lead and rear gaps ($dist_f$, $dist_r$) as well as the corresponding time-to-collision values (TTC_f, TTC_r) in the target lane. The default values are set to $dist_f$=5 m, $dist_r$=18 m, TTC_f=2.0 s, and TTC_r=2.0 s. A lane change is executed once all these conditions are satisfied. Lane changing motion is governed by using separate PID controllers for both lateral and longitudinal directions. For longitudinal control, the proportional, integral, and derivative gains are set to K_P=1.0, K_I=0.05, and K_D=0. For lateral control, the corresponding gains are K_P=1.95, K_I=0.05, and K_D=0.2.

These modules execute at control frequency, unaffected by the agents; PACE-ADS changes which behavior is active and with what parameters, not how it is realized. The three agents do not share a common cycle. The driver and psychologist agents run continuously, each re-invoked as soon as its previous inference completes, so the framework's view of the scene and of the occupant is current to within one inference. The coordinator is triggered only by a change in the inferred occupant-state output, receipt of an instruction, or a discrepancy between the vehicle's motion and its route, such as deceleration to a low speed or a sustained stop before the destination. A full invocation from trigger to decision takes 2.2 s on average, which bounds the closed loop: changes faster than this interval cannot be tracked. It is workable for the behavioral decisions PACE-ADS issues, since real-time control remains with the algorithms above, but it is the principal constraint on responsiveness.

A.6. Comparison with baseline personalization methods

To assess how much of the adaptation reported in Section 3.3 originates in PACE-ADS rather than in the underlying stack, we repeat the traffic signal, pedestrian interaction, and car following scenarios with two baselines. The first is the unmodified CARLA Behavior Agent running its default profile, which receives no occupant signal and therefore provides a reference for driving behavior in the absence of any personalization. The second, referred to here as rule-based personalization, uses the three driving profiles the Behavior Agent already provides, cautious, normal, and aggressive, and maps the five occupant states onto them through a fixed lookup: the impatient states select the aggressive profile, the relaxed state the normal profile, and the anxious states the cautious profile. This baseline adapts to occupant state without any language model and provides a concrete three-profile comparator for the present evaluation; it is not intended to represent the full range of predefined personalization approaches.

All three methods are evaluated under identical conditions, with the same randomization used in Section 3.3: the initial distance to the traffic light is varied between 200 m and 300 m, the initial distance to the pedestrian between 100 m and 200 m, and in the car following scenario the lead vehicle follows the speed profile described in Section 3.3.4 with an additional random variation of ±5 km/h applied at each phase of the profile. Each condition is repeated 30 times. The work zone scenario is not included, because it requires a lane change and neither baseline is able to perform one; the comparison there would not be meaningful.

Two patterns are visible in Tables A.6 to A.8. The CARLA Behavior Agent produces essentially the same behavior under all five occupant states, as expected, which establishes that the full range of behavior reported in Section 3.3 is attributable to the framework rather than to the underlying algorithms. Rule-based personalization recovers part of that range but cannot distinguish within a valence family, since only three profiles are available: its values under very impatient and impatient are indistinguishable, as are those under anxious and very anxious, whereas PACE-ADS separates them. Accordingly, relative to this evaluated three-profile rule-based baseline, PACE-ADS provides finer-grained adaptation across the five prescribed occupant states. This result should not be interpreted as establishing a general advantage over predefined personalization approaches. The cost is inference time, which rises from a few milliseconds for both baselines to roughly 2.3 s for PACE-ADS.

Table A.6. Comparison of traffic signal interaction results across personalization methods (30 runs per condition).

Occupant state	**Method**	**v_a (km/h)**	**v_c (km/h)**	**b (m/s²)**	**a (m/s²)**	***Inference time* (s)**
Very impatient	Behavior Agent	40.1±0.6	32.2±0.8	1.01±0.08	2.06±0.11	0.004±0.001
	Rule-based	53.7±1.2	48.5±1.4	4.37±0.29	3.41±0.22	0.007±0.002
	PACE-ADS	**59.6±0.7**	**55.7±0.9**	**5.51±0.28**	**4.23±0.22**	**2.31±0.29**
Impatient	Behavior Agent	40.0±0.6	32.1±0.8	1.03±0.08	2.05±0.12	0.004±0.001
	Rule-based	53.8±1.1	48.7±1.3	4.39±0.29	3.41±0.23	0.007±0.002

	PACE-ADS	**55.3±1.0**	**50.1±1.1**	**4.63±0.25**	**3.59±0.20**	**2.27±0.26**
Relaxed	Behavior Agent	40.1±0.5	32.2±0.8	1.02±0.06	2.07±0.14	0.004±0.002
	Rule-based	40.0±0.7	32.1±0.9	1.01±0.09	2.04±0.13	0.007±0.002
	PACE-ADS	**40.0±0.6**	**32.3±0.8**	**1.02±0.08**	**2.08±0.12**	**2.19±0.24**
Anxious	Behavior Agent	40.1±0.6	32.2±0.8	1.02±0.08	2.06±0.11	0.004±0.001
	Rule-based	29.2±0.9	21.4±0.7	0.57±0.06	1.94±0.12	0.007±0.002
	PACE-ADS	**27.1±0.8**	**19.6±0.6**	**0.49±0.05**	**1.86±0.10**	**2.35±0.31**
Very anxious	Behavior Agent	40.0±0.6	32.1±0.8	1.03±0.08	2.05±0.12	0.004±0.001
	Rule-based	29.2±0.6	21.2±0.7	0.56±0.06	1.95±0.11	0.007±0.003
	PACE-ADS	**26.3±0.7**	**19.3±0.5**	**0.42±0.04**	**1.78±0.09**	**2.41±0.33**

Note: Values are reported as mean ± SD over 30 runs. "Behavior Agent" is the unmodified CARLA Behavior Agent on its default profile and "Rule-based" is the fixed mapping of occupant states onto its three driving profiles. Inference time is measured from input receipt to behavioral-decision output. Neither baseline responds to occupant state in the very impatient versus impatient pair or in the anxious versus very anxious pair, because the rule-based method has only three profiles available.

Table A.7. Comparison of pedestrian interaction results across personalization methods (30 runs per condition).

Occupant state	**Method**	***Δd*** **(m)**	***d_{lon}*** **(m)**	***d_{lat}*** **(m)**	***Inference time*** **(s)**
Very impatient	Behavior Agent	46.5±2.0	2.91±0.14	2.46±0.12	0.004±0.002
	Rule-based	37.0±1.9	1.66±0.05	1.62±0.04	0.007±0.003
	PACE-ADS	**24.3±1.4**	**1.67±0.05**	**1.62±0.03**	**2.38±0.30**
Impatient	Behavior Agent	46.6±2.1	2.92±0.15	2.47±0.13	0.004±0.001
	Rule-based	37.2±1.7	1.66±0.02	1.63±0.05	0.007±0.002
	PACE-ADS	**35.4±1.8**	**1.72±0.06**	**1.68±0.05**	**2.31±0.27**
Relaxed	Behavior Agent	46.5±2.0	2.92±0.13	2.46±0.15	0.004±0.002
	Rule-based	46.4±2.2	2.90±0.15	2.45±0.13	0.007±0.002
	PACE-ADS	**46.6±2.1**	**2.93±0.14**	**2.47±0.12**	**2.22±0.25**
Anxious	Behavior Agent	46.7±2.1	2.94±0.15	2.48±0.13	0.004±0.001
	Rule-based	57.6±2.8	3.89±0.22	3.31±0.19	0.007±0.002
	PACE-ADS	**59.9±2.7**	**4.14±0.20**	**3.53±0.18**	**2.42±0.32**
Very anxious	Behavior Agent	46.6±2.0	2.93±0.14	2.47±0.12	0.004±0.001
	Rule-based	57.6±2.5	3.88±0.21	3.34±0.21	0.007±0.001
	PACE-ADS	**78.8±3.2**	**6.04±0.27**	**5.25±0.24**	**2.49±0.35**

Note: Values are reported as mean ± SD over 30 runs. Δd is the vehicle–pedestrian distance at the onset of deceleration, d_lon the stopping clearance, and d_lat the pedestrian's lateral clearance when the vehicle resumes motion.

Table A.8. Comparison of car following results across personalization methods (30 runs per condition).

Occupant state	**Method**	**$\bar{v}$ (km/h)**	**$\|\bar{a}\|$ (m/s²)**	**$\|\bar{b}\|$ (m/s²)**	***Inference time*** **(s)**
Very impatient	Behavior Agent	42.1±1.1	0.60±0.05	0.95±0.08	0.004±0.003
	Rule-based	42.7±1.2	0.76±0.05	1.04±0.05	0.007±0.002
	PACE-ADS	**54.8±0.8**	**1.39±0.08**	—	**2.34±0.32**
Impatient	Behavior Agent	41.2±1.5	0.58±0.02	0.91±0.07	0.004±0.003
	Rule-based	42.9±1.1	0.75±0.05	1.08±0.08	0.007±0.001
	PACE-ADS	**46.1±1.0**	**0.91±0.06**	—	**2.29±0.27**
Relaxed	Behavior Agent	41.1±1.1	0.62±0.08	0.97±0.05	0.003±0.002
	Rule-based	42.1±1.4	0.60±0.03	0.95±0.07	0.007±0.001
	PACE-ADS	**42.5±1.1**	**0.56±0.04**	**0.98±0.07**	**2.21±0.25**
Anxious	Behavior Agent	42.2±1.2	0.59±0.07	0.96±0.07	0.004±0.001
	Rule-based	36.2±0.9	0.44±0.04	0.72±0.06	0.006±0.004

	PACE-ADS	**34.4±0.9**	**0.39±0.03**	**0.43±0.04**	**2.39±0.31**
Very anxious	Behavior Agent	41.8±1.1	0.59±0.04	0.96±0.04	0.003±0.001
	Rule-based	36.3±0.8	0.41±0.06	0.70±0.05	0.007±0.002
	PACE-ADS	**30.2±0.6**	**0.38±0.03**	**1.39±0.09**	**2.46±0.34**
Note: Values are reported as mean ± SD over 30 runs. The CARLA Behavior Agent and the rule-based method remain in the current lane, so their speeds are constrained by the lead vehicle and both acceleration and deceleration are reported. PACE-ADS changes lanes under the impatient and very impatient conditions, so no lead-vehicle-induced deceleration phase occurs in those runs, indicated by a dash.					

Author contributions

Zhipeng Bao: Writing – original draft, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization, Software, Data curation. **Qianwen Li**: Writing – review & editing, Supervision, Conceptualization, Methodology, Funding acquisition.

Replication and data sharing

All data, code, simulation configurations, and supplementary materials required to reproduce the results of this study are currently being organized. In accordance with the journal's Data and Code Policy, a complete replication package, together with an explanatory replication file detailing the procedures for reproducing the main findings, will be made publicly available upon acceptance of the article. After acceptance, the replication materials will be deposited in ETS-Data (https://ets-data.sciopen.com) or another openly accessible repository, ensuring that other researchers can readily access, verify, and extend our work.

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

Strategic Market Research. (2022). Advanced Driver Assistance Systems (ADAS) Market: By System Type, Component Type, Vehicle Type, Electric Vehicle, Region – Global Growth Forecast 2021-2030. https://www.strategicmarketresearch.com/market-report/adas-market. Accessed February 5, 2026.

Channamallu, S. S., Almaskati, D., Kermanshachi, S., & Pamidimukkala, A. (2025). Autonomous vehicle safety: An advanced bagging classifier technique for crash injury prediction. *Multimodal Transportation*, *4*(1), 100189.

Xu, Z., & Zheng, N. (2024). Integrating connected autonomous shuttle buses as an alternative for public transport – A simulation-based study. *Multimodal Transportation*, *3*(2), 100133.

Schrum, M. L., Sumner, E., Gombolay, M. C., & Best, A. (2024). Maveric: A data-driven approach to personalized autonomous driving. IEEE Transactions on Robotics, 40, 1952-1965.

Su, C., Deng, W., He, R., Wu, J., & Jiang, Y. (2018). Personalized adaptive cruise control considering drivers' characteristics (No. 2018-01-0591). SAE Technical Paper.

Zhu, B., Yan, S., Zhao, J., & Deng, W. (2018). Personalized lane-change assistance system with driver behavior identification. IEEE Transactions on Vehicular Technology, 67(11), 10293-10306.

Han, Z., Zheng, X., Ren, Y., Li, X., & Wang, Q. (2023). Personalized adaptive cruise control with deep reinforcement learning. In Ergonomics in Design (AHFE Open Access, Vol. 77). AHFE International. https://doi.org/10.54941/ahfe1003421

Wang, Y., Wang, Z., Han, K., Tiwari, P., & Work, D. B. (2022). Gaussian process-based personalized adaptive cruise control. IEEE Transactions on Intelligent Transportation Systems, 23(11), 21178-21189.

Nava, D., Panzani, G., Zampieri, P., & Savaresi, S. M. (2019, October). A personalized Adaptive Cruise Control driving style characterization based on a learning approach. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (pp. 2901-2906). IEEE.

Gao, B., Cai, K., Qu, T., Hu, Y., & Chen, H. (2020). Personalized adaptive cruise control based on online driving style recognition technology and model predictive control. IEEE transactions on vehicular technology, 69(11), 12482-12496.

Lee, H., & Samuel, S. (2024). Classification of User Preference for Self-Driving Mode and Behaviors of Autonomous Vehicle. IEEE Transactions on Intelligent Vehicles.

Ling, J., Li, J., Tei, K., & Honiden, S. (2021, December). Towards personalized autonomous driving: An emotion preference style adaptation framework. In 2021 IEEE International Conference on Agents (ICA) (pp. 47-52). IEEE.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., ... & Wierstra, D. (2015). Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971.

Chen, X., Zhai, Y., Lu, C., Gong, J., & Wang, G. (2017, June). A learning model for personalized adaptive cruise control. In 2017 IEEE intelligent vehicles symposium (IV) (pp. 379-384). IEEE.

Bao, N., Capito, L., Yang, D., Carballo, A., Miyajima, C., & Takeda, K. (2022). Data-driven risk-sensitive control for personalized lane change maneuvers. IEEE Access, 10, 36397-36415.

Tian, H., Wei, C., Jiang, C., Li, Z., & Hu, J. (2022).

Personalized lane change planning and control by imitation learning from drivers. IEEE Transactions on Industrial Electronics, 70(4), 3995-4006.

Li, D., & Liu, A. (2023). Personalized lane change decision algorithm using deep reinforcement learning approach. Applied Intelligence, 53(11), 13192-13205.

Zhao, Z., Liao, X., Abdelraouf, A., Han, K., Gupta, R., Barth, M. J., & Wu, G. (2023, October). Real-time learning of driving gap preference for personalized adaptive cruise control. In 2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 4675-4682). IEEE.

Doudou, M., Bouabdallah, A., & Berge-Cherfaoui, V. (2020). Driver drowsiness measurement technologies: Current research, market solutions, and challenges. International Journal of Intelligent Transportation Systems Research, 18, 297-319.

Kashevnik, A., Shchedrin, R., Kaiser, C., & Stocker, A. (2021). Driver distraction detection methods: A literature review and framework. IEEE Access, 9, 60063-60076.

Zhang, Z., Ning, H., & Zhou, F. (2022). A systematic survey of driving fatigue monitoring. IEEE transactions on intelligent transportation systems, 23(11), 19999-20020.

Misra, A., Samuel, S., Cao, S., & Shariatmadari, K. (2023). Detection of driver cognitive distraction using machine learning methods. IEEE Access, 11, 18000-18012.

Schumann, O., Buchholz, M., & Dietmayer, K. (2023, September). Efficient Path Planning in Large Unknown Environments with Switchable System Models for Automated Vehicles. In 2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC) (pp. 2466-2472). IEEE.

Baldini, F., Tariq, F. M., Bae, S., & Isele, D. (2024). Don't Get Stuck: A Deadlock Recovery Approach. arXiv preprint arXiv:2408.10167.

Honda, K., Yonetani, R., Nishimura, M., & Kozuno, T. (2024, May). When to replan? an adaptive replanning strategy for autonomous navigation using deep reinforcement learning. In 2024 IEEE International Conference on Robotics and Automation (ICRA) (pp. 6650-6656). IEEE.

Schmittle, M., Baijal, R., Hou, B., Srinivasa, S., & Boots, B. (2024, May). Multi-Sample Long Range Path Planning under Sensing Uncertainty for Off-Road Autonomous Driving. In 2024 IEEE International Conference on Robotics and Automation (ICRA) (pp. 4598-4604). IEEE.

King, R.E. (2024). Empty Waymo robotaxi loops roundabout 37 times as if it was stuck in logic loop. Jalopnik. https://jalopnik.com/empty-waymo-robotaxi-loops-roundabout-37-times-as-if-it-1851721933. Accessed February 5, 2026.

New Tang Dynasty Television. (2024, July 12). "Apollo Go" robotaxi is unable to move and causes traffic congestion; Wuhan traffic police are helpless [in Chinese]. https://www.ntdtv.com/gb/2024/07/12/a103896787.html. Accessed February 5, 2026.

Kondyli, V., Bhatt, M., Levin, D., & Suchan, J. (2023). How do drivers mitigate the effects of naturalistic visual complexity? On attentional strategies and their implications under a change blindness protocol. Cognitive research: principles and implications, 8(1), 54.

Harris, M. (2022, August 4). Tesla's Autopilot depends on a deluge of data. IEEE Spectrum. https://spectrum.ieee.org/tesla-autopilot-data-deluge. Accessed February 5, 2026.

Bao, Z., & Li, Q. (2025). Large language model-assisted autonomous vehicle recovery from immobilization. arXiv preprint arXiv:2510.26023.

Jia, X., Yang, Z., Li, Q., Zhang, Z., & Yan, J. (2024). Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. Advances in Neural Information Processing Systems, 37, 819-844.

Lüddecke, T., & Ecker, A. (2022). Image segmentation using text and image prompts. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 7086-7096).

Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., ... & Corrado, G. (2023). Gaia-1: A generative world model for autonomous driving. arXiv preprint arXiv:2309.17080.

Mao, J., Qian, Y., Ye, J., Zhao, H., & Wang, Y. (2023). Gpt-driver: Learning to drive with gpt. arXiv preprint arXiv:2310.01415.

Wang, Y., Jiao, R., Zhan, S. S., Lang, C., Huang, C., Wang, Z., ... & Zhu, Q. (2023). Empowering autonomous driving with large language models: A safety perspective. arXiv preprint arXiv:2312.00812.

Cui, C., Ma, Y., Cao, X., Ye, W., & Wang, Z. (2024a). Drive as you speak: Enabling human-like interaction with large language models in autonomous vehicles. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 902-909).

Xu, Z., Zhang, Y., Xie, E., Zhao, Z., Guo, Y., Wong, K. Y. K., ... & Zhao, H. (2024). Drivegpt4: Interpretable end-to-end autonomous driving via large language model. IEEE Robotics and Automation Letters.

Fang, S., Liu, J., Ding, M., Cui, Y., Lv, C., Hang, P., & Sun, J. (2025). Towards interactive and learnable cooperative driving automation: a large language model-driven decision-making framework. IEEE Transactions on Vehicular Technology.

Liu, Y., Wu, F., Liu, Z., Wang, K., Wang, F., & Qu, X. (2023). Can language models be used for real-world urban-delivery route optimization? *The Innovation*, *4*(6), 100520.

Wu, D., Han, W., Liu, Y., Wang, T., Xu, C. Z., Zhang, X., & Shen, J. (2025, April). Language prompt for autonomous driving. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 39, No. 8, pp. 8359-8367).

Wen, L., Fu, D., Li, X., Cai, X., Ma, T., Cai, P., Dou, M., Shi, B., He, L., & Qiao, Y. (2024). DiLu: A knowledge-driven approach to autonomous driving with large language models. In The Twelfth International Conference on

Learning Representations (ICLR).

Sha, H., Mu, Y., Jiang, Y., Chen, L., Xu, C., Luo, P., Li, S. E., Tomizuka, M., Zhan, W., & Ding, M. (2023). LanguageMPC: Large language models as decision makers for autonomous driving. arXiv preprint arXiv:2310.03026.

Cui, C., Yang, Z., Zhou, Y., Peng, J., Park, S. Y., Zhang, C., ... & Wang, Z. (2024b). On-Board Vision-Language Models for Personalized Autonomous Vehicle Motion Control: System Design and Real-World Validation. arXiv preprint arXiv:2411.11913.

Kou, G., Jia, F., Mao, W., Liu, Y., Zhao, Y., Zhang, Z., Yoshie,

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., ... & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824-24837.

Hao, X., Diao, Y., Wei, M., Yang, Y., Hao, P., Yin, R., ... & Liu, Y. (2025). MapFusion: A novel BEV feature fusion network for multi-modal map construction. Information Fusion, 119, 103018.

Jamali, M., Davidsson, P., Khoshkangini, R., Ljungqvist, M. G., & Mihailescu, R. C. (2025). Context in object detection: a systematic literature review. Artificial Intelligence Review, 58(6), 1-89.

Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., ... & Kivlichan, I. (2024). Gpt-4o system card. arXiv preprint arXiv:2410.21276.

Macenski, S., Foote, T., Gerkey, B., Lalancette, C., & Woodall, W. (2022). Robot operating system 2: Design, architecture, and uses in the wild. Science Robotics, 7(66), eabm6074.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., & Koltun, V. (2017, October). CARLA: An open urban driving simulator. In Conference on robot learning (pp. 1-16). PMLR.

Hart, P. E., Nilsson, N. J., & Raphael, B. (1972). Correction to" a formal basis for the heuristic determination of minimum cost paths". ACM SIGART Bulletin, (37), 28-29.

Kesting, A., Treiber, M., & Helbing, D. (2010). Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 368(1928), 4585-4605.

Saganowski, S., Komoszyńska, J., Behnke, M., Perz, B., Kunc, D., Klich, B., ... & Kazienko, P. (2022). Emognition dataset: Emotion recognition with self-reports, facial expressions, and physiology using wearables. Scientific Data, 9(1), 158.

Deffenbacher, J. L., Oetting, E. R., & Lynch, R. S. (1994). Development of a driving anger scale. Psychological

O., Wang, T., Li, Y., & Zhang, X. (2025). PADriver: Towards personalized autonomous driving. arXiv preprint arXiv:2505.05240.

Wang, Z., Jiang, H., Dong, S., Wang, Y., Qiu, H., & Li, J. (2026). Drive my way: Preference alignment of vision-language-action model for personalized driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 25204–25214).

Cui, C., Yang, Z., Zhou, Y., Ma, Y., Lu, J., Li, L., Chen, Y., Panchal, J., & Wang, Z. (2024c). Personalized autonomous driving with large language models: Field experiments. arXiv preprint arXiv:2312.09397.

Reports, 74(1), 83-91.

Russell, J. A. (1980). A circumplex model of affect. Journal of Personality and Social Psychology, 39(6), 1161-1178.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023, July). Robust speech recognition via large-scale weak supervision. In International conference on machine learning (pp. 28492-28518). PMLR.

Author biography

Zhipeng Bao received the B.S. degree in automotive engineering from Wuhan University of Technology, and the M.S. degree in vehicle engineering from Jilin University, China. He is currently pursuing a Ph.D. degree in mechanical engineering at the University of Georgia, USA. His research interests lie in embodied AI and the application of vision language models in autonomous driving. He has worked on vehicle trajectory prediction, drone trajectory planning, 3D computer vision and SLAM systems.

Qianwen (Cami) Li is an Assistant Professor in the School of Environmental, Civil, Agricultural & Mechanical Engineering (ECAM) at the University of Georgia. She received the Ph.D. degree in civil engineering (transportation) from the University of South Florida. Her research focuses on enhancing transportation safety and efficiency using emerging technologies such as vehicle automation, communication, electrification, and smart infrastructure.