Behaviour Modeling of Virtual Autonomous Driving Agent Using Voice Command in Risky Scenarios

Velichko Minev
Department Of Computer Systems and Technologies
Technical University of Sofia, Branch Plovdiv,
Plovdiv, Bulgaria
Email: vilim2001@abv.bg

Dilyana Budakova
Department Of Computer Systems and Technologies
Technical University of Sofia, Branch Plovdiv
Plovdiv, Bulgaria
Email: dilyana_budakova@tu-plovdiv.bg

Abstract— **The present paper describes behavior modeling of a multimodal virtual agent–automobile, used in an urban environment under conditions of reduced visibility. In situations in which the virtual agent cannot decide how to act, it receives voice commands from a human assistant. The agent reacts to the voice instructions, reasoning over them, interpreting, and eventually carrying them out. It operates in a hybrid control mode, which includes autonomy with the possibility for a human voice intervention. The architecture of the agent is presented in the paper. Experimental evaluation of the effectiveness of recognition and interpretation of voice commands by a system, based on a Large Language Model (LLM) in risky scenarios with reduced visibility, has been made and described. The results show that the agent's behavior in a risky environment improves as a result of receiving and executing voice commands from a remote operator-assistant. Further research possibilities into behavior modeling of autonomous virtual agents, related to integration of Virtual Reality and Multimodal Large Language Models, are also discussed.**

Keywords-behaviour; modelling; agents; voice commands; tele-assistance; risky scenarios; visibility.

I. INTRODUCTION

Autonomous, driverless electric automobiles, such as Waymo [1], Zoox robot taxis [2], AutoX's self-driving grocery delivery vehicles [3], and Tesla's Full Self-Driving (Supervised) [4], combine inspiring innovative technologies, aiming to provide a safe, environmentally friendly and accessible urban mobility environment for people.

However, there are risky scenarios, such as traffic jams, dense fog, smoke, fire, destruction or rescue operations, in which the automated system cannot independently decide on action. Various approaches, in which a human assistant can remotely give commands to the system, are applied in these cases. The system can be switched to a mode of manual or remote control on the side of a tele-operator, or it can be operated semi-autonomously. The human tele-operator can remotely perform the driving tasks in whole or in part.

This paper describes a developed model of a multimodal virtual autonomous agent–automobile, traveling along a given route in a dynamic urban environment under conditions of reduced visibility due to smoke and fog. The agent–automobile operates in a hybrid control mode, which includes both autonomy and the possibility for a remote intervention by a tele-assistant by means of voice commands. The agent's architecture integrates visual perception, recognition of dynamic objects (people and other cars), interpretation, reasoning, and execution of voice commands via Whisper and Large Language Model (LLM).

An approach has been proposed and experimentally investigated in which voice commands are used to assist the virtual agent-automobile in its driving task when, due to reduced visibility, it cannot decide what action to take. The experiments show that voice commands help reduce uncertainty, and when implemented, the agent's behavior in a risky environment improves. This has been confirmed experimentally in dangerous scenarios by measuring the virtual agent's reaction time and the number of successful passages through various obstacles.

The model combines several modern research directions: multimodal autonomous agents; human-machine interaction through voice advice; 3D modeled virtual environment as an experimental platform; voice control in case of low visibility (fog or smoke); voice control, implemented using Whisper and Large Language Model.

The paper is structured as follows: Section 2 reviews existing solutions for human-assisted autonomous driving. Section 3 details the architecture of the proposed multimodal autonomous agent. Section 4 describes the experimental setup, including the technologies used, the modeling of the virtual environment, and the specific test scenarios. Section 5 presents and analyzes the experimental results, while Section 6 provides conclusions, generalizations, and outlines future trends for autonomous agent development.

II. SUPPORTING THE AUTONOMOUS AGENTS DRIVERS BY HUMAN TELE-ASSISTANTS IN SITUATIONS, IN WHICH THE AGENTS CANNOT MAKE A DECISION INDEPENDENTLY

Conducting repeated tests of autonomous driving systems in risky scenarios is dangerous, expensive, and virtually impossible. Realistic test platforms, based on virtual reality and 3D modeling, are used for these purposes. Such a platform is presented in [5]. Other examples include: Carcraft software developed by Google's Waymo automated driving team; AirSim system for autopilot vehicle testing at Microsoft [6]; Apollo virtual driving platform, created by Baidu Apollo [7].

According to manufacturers and scientists, at this stage, the Autonomous Driving Systems (ADS) may encounter situations in which they cannot make an independent

decision. In these cases, a human being is needed to solve the problem remotely [8].

One solution is Full Self-Driving (Supervised) Systems [4], [9]. They require a human driver to actively monitor the traffic situation during the journey and his/her minimal intervention. The driver can simultaneously see the real situation on the road and watch the road situation as it is perceived by the autonomous system on a screen. Thus, he/she is in a position to react appropriately when registering a discrepancy.

When critical situations occur, a solution package can be provided for autonomous automobile fleets [10]. This package includes support for tele-operations [4], [10], [11], by means of which the fleets are monitored remotely, and real-time information is provided, allowing the operators to offer help when needed.

There is a wide variety of approaches to the ways of solving problems encountered by an autonomous automobile. They are classified according to their complexity in [8], and a taxonomy for Remote Human Input Systems (RHIS) is presented, as well as a Dynamic Driving Task (DDT), where remote assistance and remote driving may be required. Here are examples of such approaches: detecting an object or event, sending information, guiding along a path, and others.

According to [12], [13], innovative models for remote control of autonomous vehicles are needed. A Survey on Tele-operation Concepts for Automated Vehicles is presented in [14]. In [12], the authors explore the construction of a command language as a first step for designing a Tele-assistance user interface. A tablet and command buttons are used for each of the commands for this purpose. When a button is pressed, a command is transmitted to the autonomous vehicle and the vehicle executes it.

According to [15], the integration of LLM into autonomous driving systems will improve the natural language user interface. The autonomous driving systems will explain everything they see and do during a journey. This is supposed to create comfort, trust, and a feeling of security.

LLMs will also help with processing big data (on the scale of the internet data), as well as with managing complex, multi-step scenarios, requiring higher-level reasoning. LLMs will be integrated into planning systems, which will improve understanding of the context of the monitored situation and the corresponding decision-making for a given context [15]. Comparison of LLM-based Autonomous Driving Systems, as well as Comparison of Multimodal Large Language Model-based (MLLM-based) Autonomous Driving Systems, is made in [15].

With the introduction of a video encoder, MLLM systems can directly process visual information from driving scenarios and implement multimodal reasoning. Recent trends in science show that Generative Artificial Intelligence for autonomous driving is approaching the field of embodied AI, such as robotics. This helps to develop vision-language-action (VLA) models.

Since the risky situations are numerous, the need for tele-operation and especially tele-assistance by a human operator is increasingly seen as extremely important. The low latency of 5G networks and the high reliability of wireless communication channels allow to construct “vehicle control towers” where several human assistants can control many vehicles. Challenges related to achieving safety and cybersecurity of such a system are discussed in [16].

III. ARCHITECTURE OF A MULTIMODAL AUTONOMOUS AGENT-VEHICLE

The considered trends in developing remote-control models for autonomous vehicles and the existing guidelines for creating a tele-assistance user interface justify the conduction of a study to assess the effectiveness of using LLM-based recognition systems, reasoning, and executing voice commands, given, e.g., by means of Whisper + LLM in various risky scenarios. What is important is this system's resistance to noise, latency, and the number of misinterpretations of the given commands at critical moments.

This paper examines the degree of effectiveness of using voice commands for controlling automated driving systems in risky scenarios.

The architecture of the multimodal autonomous agent-driver is presented in Figure 1.

The main blocks in this architecture are as follows:

- Multimodal input, supporting: Unity Camera for recognizing the environment, the street, other pedestrians, and vehicles; Unity Radar and Unity Raycast system for determining the distance to dynamic and static objects; Unity Raycast system to model the composition of fog and smoke; Whisper model for automatic recognition of the voice commands, given by a remote tele-assistant.
- A Cognitive module for reasoning and for deciding whether to drive autonomously or to execute voice commands, coming from a remote human assistant. The module includes Multimodal Large Language Modules (MLLM), Whisper model, algorithms for Deep Learning (DL), and for Reinforcement Learning (RL), as well as for Reinforcement Learning with Human Feedback, and Rules for choosing the type of driving control (autonomous or following voice commands from a tele-assistant).
- A module for implementing the movement of the vehicle, controlled by the autonomous agent-driver. It includes the capabilities provided by the multiplatform environment Unity, such as a navigation system, radar models, raycast, and a keyboard.
- Whisper is a state-of-the-art model for Automatic Speech Recognition (ASR) and speech translation [17]. Built as an encoder-decoder model, Whisper processes audio input and extracts a text command, passed to FastAPI. An asynchronous speech activity detection mechanism is used for this

purpose, which records and analyzes the audio signal.

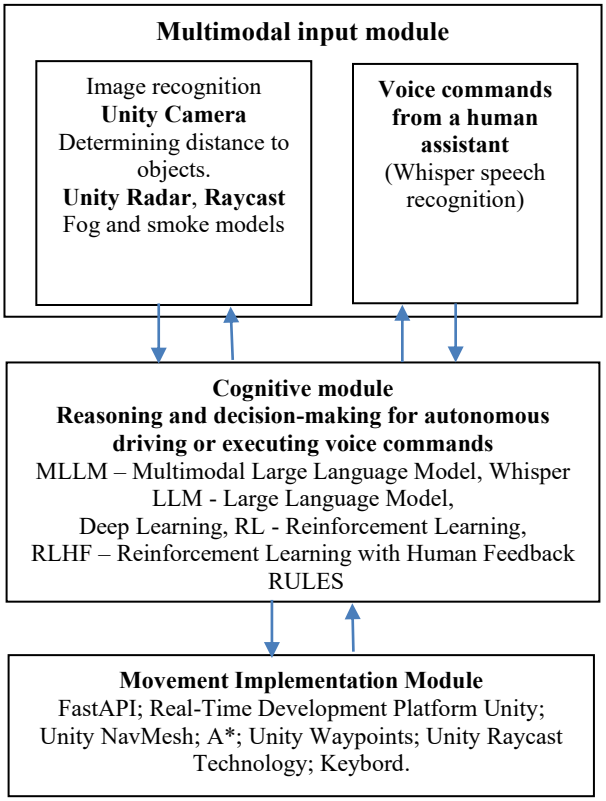

Figure 1. Architecture of a multimodal autonomous agent-driver.

FastAPI [18] is a web framework for building RESTful API interfaces in Python. It provides efficient and asynchronous communication and is therefore suitable for systems processing data in real time. In this project, FastAPI provides the connection between Whisper and the Real-Time Development Platform Unity. It ensures fast and reliable transmission of text commands received through speech recognition. After processing the command, FastAPI returns a confirmation of its successful execution or an error message (e.g., "command accepted" or "command unrecognized"). This ensures clear and interactive communication within the system.

The multiplatform environment for development and 3D simulations, Unity [19], [20], was used to model the realistic virtual scene and the behavior model of the autonomous vehicle. For the agent to drive the automobile intelligently and autonomously in a complex 3D environment, the concept of a navigation mesh (NavMesh) was used. NavMesh allows agents to move smoothly and continuously within defined walkable areas without the need to follow predefined paths. The built-in A* (A-star) algorithm is used to find the shortest or most efficient path through the network of traversable polygons.

Impassable areas are automatically avoided by the virtual agent-automobile by using the NavMesh Agent component. In the context of the system, the commands received from FastAPI (e.g., "go to the intersection") are translated into target coordinates in the virtual environment. These coordinates are then fed to the NavMesh Agent. It autonomously calculates and follows the required path.

The proposed architecture integrates a suite of state-of-the-art technologies to ensure robust collaboration between the human and the agent. Whisper was selected for its resilient encoder-decoder architecture, which provides high-fidelity speech transcription even under the stressful conditions of tele-assistance. The resulting text is processed by Large Language Models (LLM) and Multimodal LLMs (MLLM), which serve as the system's 'cognitive core' by fusing linguistic commands with the visual environmental context to perform high-level reasoning. Reinforcement Learning from Human Feedback (RLHF) is incorporated as a fine-tuning mechanism to align the agent's decision-making with human expectations and safety standards. Finally, the entire framework is deployed within the Unity 3D platform, which provides a high-fidelity, physics-based simulation environment. This allows for the safe testing of edge-case scenarios, such as reduced visibility and obstacle avoidance, prior to real-world implementation.

The information flow begins with the real-time processing of environmental sensor data to assess visibility levels. If visibility falls below the 6-meter threshold, the system triggers a transition to tele-assisted mode, where the Whisper-based module captures audio input. The LLM-based cognitive core then parses this input to extract actionable instructions, which are finally converted into specific vehicle control commands (e.g., steering or braking) within the Unity simulation environment. The system uses a 6-meter visibility threshold as its quantitative uncertainty criterion.

IV. Modeling the Scene, the Agents, and the Scenarios for the Experiments

To conduct the experiments, a realistic and lively urban environment was modeled. An asset [21] was imported, which contained pre-arranged blocks of buildings and a road network. This allows the generation of an extensive and detailed map, on which the virtual autonomous automobile can travel.

Figure 2 shows a top view of the modeled urban environment and shows the route for conducting experimental tests by means of colored lines and numbers. The movement of the autonomous automobile starts at point 1, passes sequentially through points 2 and 3, and ends at point 4. The route from point 1 to point 2 is marked with a red line. The route from point 2 to point 3 is shown with a green line, and the line from point 3 to point 4 is blue.

The experimental route is designed to simulate critical situations in an urban environment. It includes sequential left and right turns, as well as a critical obstacle - a stationary large-scale truck positioned within the driving lane.

The objective is to demonstrate that voice commands from the tele-assistant can prevent a collision with an object that the vehicle's sensors fail to detect in time due to reduced visibility. In this manner, voice commands, such as 'Overtake from the left' serve as a high-level semantic control.

The arrival time of the autonomous automobile at each point on the route was measured by a script, implementing a timer.

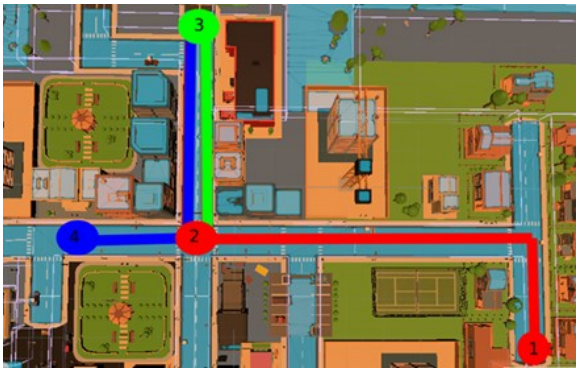

Figure 2. Model of an urban environment. Route for conducting experiments marked with colored lines and numbering.

To model traffic and achieve a realistic atmosphere, 3D models of the virtual autonomous automobile, of pedestrians, and other vehicles were implemented. To achieve the fog effect, the Unity “Fog” function was used, and its parameters were adjusted.

Raycast technology, supported by Unity, was used to simulate reduced visibility in foggy conditions. The visual range of the virtual autonomous agent-automobile is reduced by shortening the length of the rays used to detect obstacles. This leads to the delayed detection of obstacle objects and, accordingly, to a delayed reaction of the agent-automobile to avoid them. This impacts the reliability and safety of autonomous driving under conditions of low visibility.

In this study, the term 'scenarios' refers to a combination of external factors and the methods used to control the vehicle. The external factors to which the autonomous vehicle must adapt include the degree of fog and specific road conditions. The control methods include autonomous mode and voice command tele-assisted mode.

Consequently, the experimental design is structured as a 2*3 matrix: Control Modes (Autonomous vs. Voice-Assisted) and External Factors, which include full visibility, varying degrees of limited visibility, and the emergence of a complex situation (the obstacle). This structure enables a rigorous evaluation of how human intervention mitigates risks when the agent's autonomous capabilities are challenged by both environmental and situational factors.

Consequently, the experiments comprise the following scenarios:

• Monitoring autonomous movement of an autonomous automobile in a dynamic urban environment, without and with the use of voice command tele-assistance.

According to the following conditions and problems:

- Clear and sunny weather.
- Limited visibility due to dense fog.
- An event of a complex situation, which the autonomous agent cannot resolve independently.

The following characteristics were considered:

- The average time from giving a voice command to its execution.
- The number of successfully recognized and executed commands in the event of voice control.
- The total time for the different scenarios for completing the whole route from point 1 to the end point 4.
- Whether situations arise in autonomous driving in which voice commands are needed in order to continue driving (for example, a stuck autonomous vehicle).

The tele-assistant continuously monitors the autonomous vehicle and provides support through voice commands. A confidence threshold has been established, which is determined by visibility levels in foggy conditions. In clear weather, the vehicle prioritizes autonomous operation, maintaining a high level of confidence. However, when visibility drops to six (6) meters or less (quantitative uncertainty criterion), the agent’s confidence level decreases, and it begins to execute all voice commands provided. In critical situations, the autonomous vehicle is unable to proceed without the assistance of the tele-assistant. At this stage of the system's implementation, the agent places full trust in the tele-assistant. Future work could introduce a safety module to verify command-context consistency.

V. Experimental Results and Discussion

The results of the experiments reflect the arithmetic mean of the time it takes for the automobile to get from the starting point to the end point of the route and are presented in seconds.

Table 1 presents the results of the experiments conducted for autonomous vehicle control under conditions of full visibility of the order of 20 meters and when modeling reduced visibility of the order of eight (8), six (6), and four (4) meters. The reported time for the autonomous automobile to travel the entire route is given in the second column of Table 1. The last third column indicates whether and under what visibility conditions the need to assist autonomous driving with voice commands arises.

The results show that when visibility is reduced to six (6) meters or fewer, for example, four (4) meters, there is a need for driving assistance with voice commands. Under these risky conditions, the time to complete the entire route increases.

Table 2 presents the results of the experiments, conducted with an autonomous agent-automobile, receiving voice commands under conditions of both full visibility and reduced visibility, caused by fog. The first column of Table 2 gives the meteorological conditions modeled for each group of experiments.

The result in the form of the response time of the autonomous vehicle when receiving a voice command is given in the second column in seconds. The third column shows the time for completion of the route with the help of voice commands. The last column indicates the number of unrecognized voice commands.

TABLE I. RESULTS FROM THE BEHAVIOR OF A VIRTUAL AGENT-VEHICLE DURING A CHANGE IN VISIBILITY. TIME TO TRAVEL THE GIVEN ROUTE. NEED FOR USE OF VOICE COMMANDS

Meteorological conditions. Degree of Visibility simulated with raycast	Time to travel on the given route	Need for voice commands to continue the agent's movement along the route
Sunny Rays Visibility – 20 m.	80,36 sec.	no
Foggy Rays Visibility – 8 m.	80,47 sec.	no
Foggy Rays Visibility – 6 m.	107,72 sec.	yes
Foggy Rays Visibility – 6 m.	109,23 sec.	yes
Foggy Rays Visibility – 4 m.	117,07 sec.	yes
Foggy Rays Visibility т – 4 m.	123,63 sec.	yes

TABLE II RESULTS OF THE BEHAVIOR OF A VIRTUAL AGENT-VEHICLE DURING DRIVING UNDER DIFFERENT METEOROLOGICAL CONDITIONS. REACTION TIME FOR EXECUTING VOICE COMMANDS. NUMBER OF UNRECOGNIZED VOICE COMMANDS

Meteorological conditions	Voice command response time	Route travel time	Number of unrecognized commands
Sunny	2,66 sec.	105,95 sec.	1 num.
Sunny	2,68 sec.	96,76 sec.	0 num.
Sunny	2,64 sec.	98,77 sec.	0 num.
Foggy	2,69 sec.	109,76 sec.	0 num.
Foggy	3,00 sec.	123,85 sec.	2 num.
Foggy	3,21 sec.	117,85 sec.	0 num.

The results show a high extent of recognition of voice commands. Only two commands were not recognized in foggy conditions and one in sunny weather.

These commands include 'Turn left' and 'Overtake the truck from the right.' The failure to recognize voice commands can be attributed to changes in the tele-assistant's intonation. Stress in critical situations can impact the tele-operator's psychological state, potentially leading to alterations in the volume, speed, and intonation of the command, as well as its specific wording. Additionally, technical issues, such as audio signal interruptions may occur.

If the voice command is not recognized, the human assistant must repeat it. The autonomous vehicle's response time to voice commands is of the order of three (3) seconds.

Therefore, when using voice commands and communication between an autonomous vehicle and a human operator, this delay must be considered to effectively assist driving and avoid accidents.

The effectiveness of facilitating driving by voice commands depends on their timely delivery. The experiments show that the agent-driver may collide with an obstacle or stop if the voice command (such as turn left, go around, turn right) is not given in time.

Under conditions of simulated dense fog, a change in the behavior of the agent-automobile was observed. It executed the voice commands of the human assistant, and driving effectiveness was improved.

In risky conditions, such as reduced visibility, when there was no communication with a human assistant, it was observed that the autonomous automobile did not detect obstacles or detected them too late. This led to collisions or to situations in which the agent could not decide to take an action and stopped moving.

There were some cases, in which to continue along the route proved impossible without additional intervention by means of voice commands, such as "go around to the right" or "avoid left". As a result of the experiments, a conclusion was drawn that in the event of reduced visibility and when complex situations arise, the agent-driver needs assistance in the form of voice commands, short and clear, and given in a timely manner.

The speed of voice command transmission, i.e., the lack of a delay, is extremely important. The voice command recognition software is critical, as it must ensure correct recognition. And, finally, the speed of the autonomous vehicle's centralized electronic supercomputer architecture, which can execute the assigned commands in a timely manner, is also of great significance.

VI. CONCLUSIONS AND FUTURE WORK

This paper investigates the effectiveness of using voice commands in modeling the behavior of a multimodal autonomous virtual agent-automobile, driving in a risky environment of reduced visibility caused by fog or smoke.

The architecture of the agent is proposed, and an improvement in its behavior when executing voice commands in critical scenarios with reduced visibility is shown.

To conduct the research, a 3D scene and a model of a virtual autonomous agent-automobile are implemented. Unity Real-Time Development Platform and the Whisper model for automatic speech recognition are used. A fog model and varying degrees of visibility are included. The obtained results, concerning the behavior of the autonomous automobile, the speed of the system and the effectiveness of its operation both when using voice commands in tele-assistance and without their use, are discussed.

The results show that in risky scenarios, the use of tele-assistance with voice commands for Autonomous Driver Systems is both necessary and very effective.

Voice commands are used as high-level semantic control and reduce uncertainty, rather than replacing low-level vehicle control.

An advantage is the use of a large language model, such as the Unity Whisper model, to implement communication with voice commands. LLMs allow for the use of natural language expressions in communication. The commands can

correctly be interpreted by the autonomous vehicle without requiring humans to learn a special set of commands.

Future iterations of the system will address potential LLM risks, such as hallucinations, by incorporating a cross-verification safety module to validate commands against real-time sensor data. Additionally, a robust fail-safe protocol is planned to ensure a controlled vehicle stop in the event of connectivity or model failure.

Another question to be cleared is at what point the agent-vehicle should seek help from a human tele-assistant or from other passengers and vehicles around.

We believe that it is worth exploring the need and form of communication to be maintained between vehicles in a risky environment in one and the same area, in close proximity to each other. It is also of importance to create a balance, allowing for discussing a particular situation by neighboring vehicles on the road and for avoiding distraction and inattention.

We also place emphasis on researching the need to ensure protection and cybersecurity of Tele-operation of Connected and Automated Vehicles.

The paper also considers the latest inspiring technologies introduced in the implementation of ADS and the opportunities for further research, created by Virtual Reality, Generative Artificial Intelligence, LLM, and MLLM.

ACKNOWLEDGMENT

The authors gratefully acknowledge the financial support provided by the Project No.: BG-RRP-2.004-0005 “Improving research capacity and quality for international recognition and sustainability of Technical University of Sofia”; National Recovery and Sustainability Plan, BG-RRP-2.004 - Creating a network of research universities in Bulgaria.

REFERENCES

[1] “Waymo - Self-Driving Cars - Autonomous Vehicles - Ride-Hail,” Waymo. Accessed: Dec. 30, 2025. [Online]. Available: https://waymo.com/index/ [retrieved: February, 2026]

[2] K. Korosec, “Zoox becomes fourth company to land driverless testing permit in California,” TechCrunch. Accessed: Dec. 30, 2025. [Online]. Available: https://techcrunch.com/2020/09/18/zoox-becomes-fourth-company-to-land-driverless-testing-permit-in-california/ [retrieved: February 2026]

[3] “Journey – AutoX.” Accessed: Dec. 30, 2025. [Online]. Available: https://www.autox.ai/en/journey.html [retrieved: February 2026]

[4] “Full Self-Driving (Supervised) | Tesla.” Accessed: Dec. 30, 2025. [Online]. Available: https://www.tesla.com/fsd [retrieved: February, 2026]

[5] S. Yao, J. Zhang, Z. Hu, Y. Wang, and X. Zhou, “Autonomous-driving vehicle test technology based on virtual reality,” The Journal of Engineering, vol. 2018, no. 16, pp. 1768–1771, 2018, doi: 10.1049/joe.2018.8303.

[6] S. Shah, D. Dey, C. Lovett, and A. Kapoor, “AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles,” in Field and Service Robotics, M. Hutter and R. Siegwart, Eds., Cham: Springer International Publishing, 2018, pp. 621–635. doi: 10.1007/978-3-319-67361-5_40.

[7] Q. Liu, “Baidu Turnip Fast Running Self-driving Car Industry Development Status and the Discussion of Existing Problems,” Journal of Education, Humanities and Social Sciences, vol. 48, pp. 51–56, Mar. 2025, doi: 10.54097/732kma16.

[8] D. Bogdoll, S. Orf, L. Töttel, and J. M. Zöllner, “Taxonomy and Survey on Remote Human Input Systems for Driving Automation Systems,” in Advances in Information and Communication, K. Arai, Ed., Cham: Springer International Publishing, 2022, pp. 94–108. doi: 10.1007/978-3-030-98015-3_6.

[9] T. Zhang, “Toward Automated Vehicle Teleoperation: Vision, Opportunities, and Challenges,” IEEE Internet of Things Journal, vol. 7, no. 12, pp. 11347–11354, Dec. 2020, doi: 10.1109/JIOT.2020.3028766.

[10] “Nuro Toolkit,” Nuro. Accessed: Dec. 30, 2025. [Online]. Available: https://www.nuro.ai/nuro-toolkit, [retrieved: February, 2026]

[11] S. Lu, R. Zhong, and W. Shi, “Teleoperation Technologies for Enhancing Connected and Autonomous Vehicles,” in 2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS), Oct. 2022, pp. 435–443. doi: 10.1109/MASS56207.2022.00068.

[12] F. Tener and J. Lanir, “Devising a High-Level Command Language for the Teleoperation of Autonomous Vehicles,” International Journal of Human–Computer Interaction, vol. 41, no. 9, pp. 5299–5315, May 2025, doi: 10.1080/10447318.2024.2359224.

[13] C. Kettwich, A. Schrank, and M. Oehl, “Teleoperation of Highly Automated Vehicles in Public Transport: User-Centered Design of a Human-Machine Interface for Remote-Operation and Its Expert Usability Evaluation,” Multimodal Technologies and Interaction, vol. 5, no. 5, p. 26, May 2021, doi: 10.3390/mti5050026.

[14] D. Majstorovic, S. Hoffmann, F. Pfab, A. Schimpe, M.-M. Wolf, and F. Diermeyer, Survey on Teleoperation Concepts for Automated Vehicles. 2022. doi: 10.48550/arXiv.2208.08876.

[15] Y. Wang et. al., Generative AI for Autonomous Driving: Frontiers and Opportunities. 2025. doi: 10.48550/arXiv.2505.08854.

[16] F. J. Jiang, J. Mårtensson, and K. H. Johansson, “Safe Teleoperation of Connected and Automated Vehicles,” in Cyber-Physical-Human Systems: Fundamentals and Applications, IEEE, 2023, pp. 251–272. doi: 10.1002/9781119857433.ch10.

[17] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust Speech Recognition via Large-Scale Weak Supervision,” Dec. 06, 2022, arXiv: arXiv:2212.04356. doi: 10.48550/arXiv.2212.04356.

[18] FastAPI - Introduction. (17:38:21+00:00). GeeksforGeeks. https://www.geeksforgeeks.org/python/fastapi-introduction/ [retrieved: February, 2026]

[19] N. F. Hutchins, L. Hook, W. Friedel, and Z. Kirkendoll, “Use of Unity in Scientific Simulation and Modeling for Research and Education,” Jan. 2017, Accessed: Jan. 02, 2026. [Online]. Available: https://www.academia.edu/115586868/Use_of_Unity_in_Scientific_Simulation_and_Modeling_for_Research_and_Education [retrieved: February 2026]

[20] U. Technologies, “Unity - Manual: Unity 6.1 User Manual.” Accessed: May 02, 2025. [Online]. Available: https://docs.unity3d.com/6000.1/Documentation/Manual/UnityManual.html [retrieved: February, 2026]

[21] “The Best Assets for Game Making | Unity Asset Store.” Accessed: Dec. 30, 2025. [Online]. Available: https://assetstore.unity.com/ [retrieved: February, 2026]