# Drowsiness Aware Shared Control for Autonomous Driving: A CNN–LLM Integrated Cyber Physical System Using CARLA Simulation

Bilal El Jamal
Department of Electrical and Computer Engineering
Stony Brook University, NY, USA
Email: bilal.eljamal@stonybrook.edu

Joshita Malla
Department of Electrical and Computer Engineering
Stony Brook University, NY, USA
Email: joshita.malla@stonybrook.edu

***Abstract*—Drowsiness continues to be a major contributor to driver error, yet most monitoring systems provide limited interpretability and lack structured mechanisms for coordinated intervention. A shared-control framework is developed that links convolutional neural network based driver-state classification with a 68-point facial landmark pipeline and a reasoning layer that manages alerting, confirmation and takeover logic. Eye, mouth and pupil metrics, including EAR, MAR, MOE and PUC, are transformed into four probabilistic drowsiness states that guide graded warnings and determine when autonomous action becomes necessary. The system integrates these components with CARLA for real-time actuation, enabling controlled deceleration, lane correction and emergency maneuvers while supporting driver override in non-critical conditions.**

**Six scenarios across urban and highway environments are used to assess graduated alerting, critical-state intervention, barrier avoidance, wrong-lane entry, red-light unresponsiveness and oncoming-vehicle evasion. The architecture sustains consistent classification accuracy demonstrates reliable management of transitions between human control and automated authority. The results show that interpretable perception, structured decision logic and simulation-based actuation can be combined to form an effective shared-control layer for mitigating drowsiness-related driving risks.**

***Index Terms*—Driver monitoring, drowsiness detection, shared control, autonomous driving, convolutional neural networks, facial landmark analysis, language model reasoning, cyber-physical systems, CARLA simulator, human–automation interaction.**

## I. Introduction

Drowsiness continues to be a leading contributor to road-traffic incidents and remains difficult to manage within current driver-assistance technologies. Although advances in computer vision and sensor-based monitoring have improved fatigue detection capabilities, most existing systems stop at classification and do not provide structured mechanisms for issuing warnings, requesting driver acknowledgment or initiating autonomous intervention when the driver is unable to respond. These limitations are especially evident in safety-critical conditions, where decisions must reflect both the severity of the driver's impairment and the surrounding traffic context. As automated driving platforms evolve, the need for shared-control architectures that maintain human authority while enforcing consistent safety policies becomes increasingly central to reliable operation.

Recent research in convolutional neural network based driver monitoring has demonstrated strong performance in drowsiness detection using facial cues, yet many approaches lack interpretability and do not translate classification outputs into principled control decisions. Shared-control methods have introduced ways to blend human and automated steering inputs, but they generally assume driver attentiveness and seldom incorporate real-time cognitive assessments. A unified framework is needed to connect interpretable perception, state estimation and vehicle actuation, enabling intervention strategies that scale with impairment severity while preserving override capability in low-risk conditions.

This work introduces a cyber-physical shared-control system that integrates facial landmark extraction, convolutional neural network classification, language-model reasoning and real-time vehicle control within the CARLA simulator. A 68-point landmark model is used to compute eye, mouth and pupil metrics, which are mapped to four drowsiness states: alert, slightly drowsy, very drowsy and critical. The convolutional neural network assigns probabilistic state estimates that inform a reasoning layer responsible for generating graded alerts, determining whether driver confirmation is required and initiating vehicle takeover when no response is detected. Control actions are executed through CARLA, enabling smooth deceleration, lane correction and emergency maneuvers according to structured safety rules that distinguish between override-permissive and non-overrideable conditions.

Evaluation is conducted across six scenarios that reflect common urban and highway situations in which drowsiness and road context jointly influence safety outcomes. These include progressive fatigue in city driving, high-speed critical-state unresponsiveness, lane drift toward barriers, wrong-lane entry, red-light noncompliance and evasive maneuvers in response to oncoming vehicles. The assessment focuses on the stability of the perception pipeline, the consistency of reasoning-driven control transitions and the reliability of autonomous intervention under varying levels of driver engagement.

The remainder of the paper is organized as follows. Section II reviews research on driver monitoring, shared control and cyber-physical actuation. Section III describes the system architecture, including feature computation, convolutional neural network modeling, state mapping and reasoning logic. Section IV outlines the software requirements, simulation environment, runtime configuration and graphical interface used in the experiments. Section V details the six driving scenarios used to evaluate the system under different roadway and drowsiness conditions. Section VI presents classification accuracy and scenario-level performance. Section VII analyzes system behavior and shared-control characteristics. Section VIII identifies current constraints, and Section IX discusses potential extensions. Section X concludes the study.

## II. Related Work

Research on driver monitoring has traditionally focused on visual cues extracted from facial behavior, with early studies demonstrating that convolutional neural networks can reliably detect fatigue from eye and mouth regions under varying illumination conditions [1]. Subsequent work refined these approaches by incorporating interpretable metrics such as PERCLOS and the eye aspect ratio to strengthen robustness and provide additional structure for classifying fatigue states [2]. More recent architectures expanded the temporal dimension through hybrid CNN–LSTM models capable of capturing extended blink sequences and yawn dynamics [3]. Transformer-based models have also been explored, showing that attention mechanisms yield high accuracy for binary eye-state prediction and improved resilience to environmental variation [4]. Although these approaches establish strong perceptual baselines, most formulations restrict the problem to binary or coarse fatigue classes and do not link detection outcomes to downstream control decisions.

Explainable artificial intelligence techniques have been introduced to increase interpretability in driver monitoring, particularly in systems combining emotion recognition and drowsiness detection [5]. These methods highlight the spatial evidence leveraged by deep networks and support model debugging, but they are not embedded within broader cyber-physical frameworks that translate model outputs into structured vehicle interventions. Reviews of vision-based monitoring systems similarly emphasize perceptual challenges and metric design while noting the absence of integrated actuation pipelines capable of enforcing safety policies in response to cognitive-state degradation [6].

Parallel to perception research, shared-control studies in automated driving examine how authority is allocated between human drivers and automated systems. Foundational work distinguishes between haptic guidance, input mixing and arbitration strategies, and highlights that authority should adapt to contextual variables such as road geometry and driver intent [7]. Broader analyses of human–automation cooperation reinforce the importance of dynamic arbitration and outline mathematical frameworks for blending driver and automation commands under uncertainty [8]. These contributions provide valuable strategies for coordinating low-level control but generally assume stable driver attentiveness and do not incorporate real-time cognitive-state estimation into authority decision mechanisms.

Large language models have recently emerged in intelligent cockpit research as tools for predicting driver intent across tasks related to driving control, comfort management and infotainment systems [9]. Their ability to perform contextual inference from multimodal attributes suggests new possibilities for integrating reasoning processes into vehicle interfaces. However, existing work focuses primarily on proactive assistance and high-level behavioral prediction rather than real-time safety intervention, and it does not address the coupling between cognitive-state monitoring and automated control.

The limitations identified across these areas point to an unaddressed gap in the integration of interpretable driver-state perception, probabilistic fatigue modeling, reasoning-driven decision logic and real-time vehicle actuation. Prior studies seldom combine multi-level drowsiness classification with structured arbitration rules or evaluate the full perception-to-control pipeline within simulation environments such as CARLA, despite evidence that CARLA is suitable for systematic safety evaluation [10]. The present work addresses this gap by unifying CNN-based state estimation, a facial-metric feature pipeline, a lightweight language-model reasoning layer and scenario-driven control behaviors into a single shared-control framework.

Temporal modeling of facial behavior has also been examined in approaches that analyze frame-to-frame landmark motion. One representative method models eyelid dynamics, yawning transitions and facial geometry trajectories to capture the temporal evolution of fatigue, demonstrating that landmark-based temporal cues offer stronger discriminative power than static representations [11]. These findings support the relevance of interpretable geometric metrics and align with the facial-feature computation strategy adopted in this study.

Deep spatiotemporal networks have further advanced fatigue estimation by integrating spatial facial cues with temporal features such as blink duration and closure frequency. A notable system employs a convolutional backbone fused with temporal processing to estimate fatigue levels under real-world conditions with high responsiveness [12]. While effective for modeling long-term behavioral patterns, such architectures introduce higher computational demands and generally operate independently of downstream control logic, distinguishing them from the shared-control framework considered here.

Broader analyses of vision-based driver monitoring consolidate research on gaze estimation, head pose, facial muscle activation and blink dynamics. A recent survey highlights the diversity of visual cues and emphasizes the need for integrated architectures that unify perception, reasoning and actuation to address safety-critical requirements [13]. This perspective reinforces the motivation for coupling multi-level drowsiness assessment with explicit intervention policies as implemented in the present system.

Shared-control strategies have also expanded to data-driven

arbitration, with reinforcement learning applied to learn authority allocation policies from interaction data. One study demonstrates how learned blending policies can stabilize steering behavior while reducing risk through adaptive mixing of human and automated commands [14]. These methods provide relevant insights into control arbitration, though they typically do not incorporate driver cognitive state as an input, which remains essential to the framework developed here.

Multimodal monitoring approaches combine facial landmarks, gaze direction and head pose estimation to detect inattention more comprehensively than methods relying solely on eye closure. An example integrates landmark geometry with gaze estimation to identify disengagement across diverse driving conditions [15]. Such approaches underscore the value of multi-feature fusion and complement the metric-driven perception pipeline used in this work, although they do not integrate explicit drowsiness state mapping or corresponding intervention mechanisms.

## III. Methodology

The system follows a structured perception–reasoning–actuation pipeline designed to detect drowsiness, interpret its severity and execute appropriate shared-control actions within the CARLA simulator. The architecture consists of four principal components: a facial-landmark perception module, a metric computation and convolutional neural network classifier, a language-model reasoning layer and a shared-control actuation module. Each stage is constructed to preserve interpretability and ensure that automated interventions correspond to measurable changes in driver state.

### A. System Architecture Overview

The architecture processes video input from a virtual onboard camera and converts facial behavior into vehicle control signals through a sequential flow of modules. The perception module extracts facial landmarks and computes geometric metrics that describe eye openness, mouth dynamics and pupil stability. These metrics are passed to a convolutional neural network that estimates a probabilistic drowsiness score. A reasoning layer implemented using a compact language model interprets this probability in the context of the surrounding driving scenario and determines whether the system should warn the driver, request confirmation or intervene autonomously.

### B. Facial Landmark Extraction

Facial landmarks are obtained using a 68-point predictor model that provides consistent geometric correspondences across key regions of the face. For each frame, a frontal face detector identifies the region of interest, and the landmark model outputs coordinates for the eyes, eyebrows, nose, mouth and jawline. These coordinates form the basis of interpretable metrics that quantify short-term physiological indicators of fatigue. Landmark extraction is performed at the simulator's frame rate to maintain real-time processing capability.

### C. Metric Computation

Four metrics are computed from the facial landmarks to represent eye openness, mouth behavior and pupil stability. These metrics have been shown to correlate with drowsiness progression and provide structured inputs for classification.

The eye aspect ratio (EAR) measures vertical-to-horizontal deformation of the eye:

$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2 \cdot \|p_1 - p_4\|}$$

where $p_1, \ldots, p_6$ denote canonical eye landmarks. Low EAR values indicate partial or complete eye closure.

The mouth aspect ratio (MAR) quantifies vertical mouth opening relative to its width:

$$MAR = \frac{\|m_3 - m_9\| + \|m_4 - m_8\| + \|m_5 - m_7\|}{3 \cdot \|m_1 - m_6\|}$$

where $m_i$ are mouth landmarks. Increased MAR corresponds to yawning or fatigue-related mouth behavior.

The mouth opening extent (MOE) is computed as:

$$MOE = \frac{\|m_3 - m_9\|}{\|m_1 - m_6\|}$$

highlighting changes in vertical aperture independent of lateral scaling.

Pupil consistency (PUC) assesses the temporal stability of the pupil region:

$$PUC = 1 - \frac{1}{N} \sum_{t=1}^{N} \|c_t - \bar{c}\|$$

where $c_t$ is the estimated pupil center at frame $t$ and $\bar{c}$ is its running mean. Lower PUC values indicate unstable ocular behavior associated with drowsiness.

The four metrics collectively form a compact representation of the driver's physiological state.

### D. CNN Training Procedure

The convolutional neural network used for driver-state estimation was trained on a labeled dataset containing facial images annotated across the four drowsiness classes defined in Table I. For each image, the four computed metrics derived from the landmark model, namely EAR, MAR, MOE and PUC, formed the input feature vector. All metric values were normalized using min–max scaling to ensure consistent numerical ranges and reduce inter-subject variability.

The dataset was divided into an 80% training split and a 20% validation split, with samples shuffled to mitigate sequential dependencies arising from continuous video capture. A lightweight CNN architecture was selected to maintain real-time inference capability. Training was performed using the Adam optimizer with a learning rate of $1 \times 10^{-3}$ and a batch size of 32. The model was trained for 50 epochs, and early stopping was applied based on validation loss to prevent overfitting.

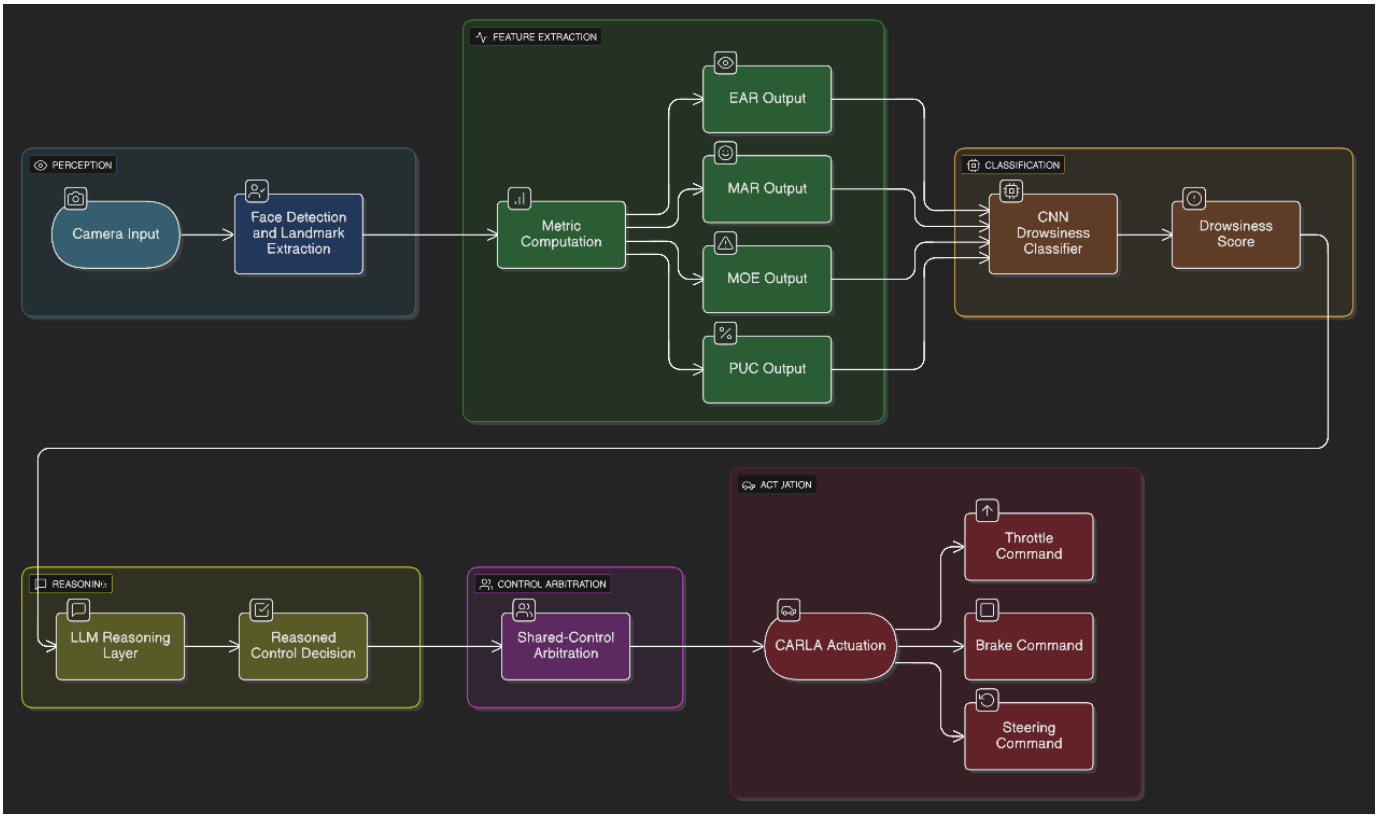


Fig. 1: Overall system architecture showing the flow from facial-landmark perception and metric extraction to CNN-based drowsiness classification, LLM reasoning and shared-control vehicle actuation in CARLA.

Training convergence produced a stable model with an accuracy of 0.9309, while validation accuracy reached 0.9792 with a corresponding validation loss of 0.1399. These results demonstrate that the metric-based feature representation provides strong discriminatory capability across the four drowsiness levels. The final trained classifier outputs a scalar probability in the interval $[0, 1]$, which is mapped to the discrete classes as described in Table I.

TABLE I: Drowsiness State Mapping Based on CNN Probability Output

| Class | Probability Range |
|---|---|
| Alert | $0.00 - 0.40$ |
| Slightly Drowsy | $0.41 - 0.70$ |
| Very Drowsy | $0.71 - 0.85$ |
| Critical Drowsiness | $0.87 - 1.00$ |

### *E. Drowsiness Classification*

The feature vector composed of EAR, MAR, MOE and PUC is provided to a convolutional neural network trained to estimate a scalar probability in the interval $[0, 1]$. The network is optimized using cross-entropy loss, and its output provides a continuous measure of fatigue severity rather than a binary label. To support downstream reasoning and control decisions, the probability is partitioned into four discrete drowsiness classes corresponding to progressive cognitive degradation. The thresholds were selected based on model confidence intervals observed during validation.

This classification scheme supplies a structured representation of the driver's condition that supports the reasoning layer described in Section III. The numerical boundaries allow the system to transition smoothly between alerting, confirmation-based interaction and full autonomous takeover depending on the assessed level of impairment.

### *F. Language Model Reasoning Layer*

A lightweight language model is used to translate the numerical drowsiness estimate into a human-readable instruction delivered to the driver in real time. The model receives the predicted class, the probability score and the four computed metrics and generates a single short message. The reasoning layer is not designed to reclassify the input; rather, it applies a controlled linguistic policy that ensures consistent system behavior across all scenarios.

The model is guided by a fixed prompt that defines strict behavioral rules for the assistant. These rules specify that

the output must correspond exclusively to the predicted class, must not reveal internal model details and must follow tone constraints based on the severity of the estimated state. The instructions also restrict the message to a single actionable sentence to preserve clarity during active driving. The exact prompt used to generate the reasoning-layer output is provided in Listing 1.

```
You are an empathetic in-car AI safety assistant. Your job is to give ONE short message to the driver based ONLY on the information below.

Model Output:
Predicted class: {state}
Prediction probability: {prob}

Feature importance values (absolute, gradient-based):
{importance_text}
These values show which facial regions influenced the classifier, such as the eyes, mouth, or pupil region.

Classifier rules:
0.0 - 0.40 : alert (not drowsy)
0.41 - 0.70 : slightly drowsy
0.71 - 0.85 : very drowsy
0.86 - 1.00 : critical drowsiness

INSTRUCTIONS:
1. Determine the driver's stage by placing the prediction probability into the correct classifier range.
2. State the driver's condition FIRST.
3. If alert:
   Give a calm confirmation and STOP.
4. If any level of drowsiness:
   Explain briefly and kindly what this means for their alertness.
   Reference the facial regions with high importance values.
   Provide a warning scaled to the stage:
      slightly drowsy -> gentle suggestion to rest soon
      very drowsy -> firm warning to slow down and rest now
      critical -> urgent instruction to pull over immediately
5. Only for critical drowsiness:
   If the driver believes the system is mistaken, tell them they may press the cancel button.
6. NEVER mention numbers, probabilities, thresholds, or internal values.
7. NEVER describe all possible stages.
8. NEVER explain how the classifier works.
9. NEVER wrap the response in quotes.
10. Keep the response short, human, supportive, and focused ONLY on the detected class.
```

Listing 1: Prompt used for the Mistral reasoning layer

The reasoning layer enforces these constraints and generates a message reflecting only the current estimated class. For alert and slightly drowsy states, the system provides confirmation or gentle suggestions while maintaining full driver authority. In very drowsy states, the model issues a firm warning to motivate immediate corrective action. For critical states, the generated instruction demands urgent intervention and informs the driver of the option to override the system by pressing the cancel button. The output from this module is then provided to the shared-control arbitration mechanism described in Section III.

### G. Shared-Control Arbitration

The reasoning output is translated into low-level control commands that respect predefined arbitration rules. In alert, slightly drowsy and very drowsy states, the driver retains full control while receiving graded warnings. In critical states, the system requests confirmation and initiates takeover if no response is received. For high-risk conditions such as barrier drift, wrong-lane entry or imminent collision, the system intervenes immediately without permitting override. Arbitration therefore depends jointly on driver state and contextual danger, allowing the system to balance autonomy with human authority.

### H. Vehicle Actuation in CARLA

The final control decision is executed through CARLA's vehicle interface, which provides direct control over throttle, brake and steering. The control policy applies smooth deceleration, lane correction or evasive maneuvers according to the scenario's safety requirements. CARLA logs vehicle kinematics, control authority transitions and state estimates for post-simulation analysis. The architecture supports real-time operation and ensures that intervention behavior is consistent with the severity of detected fatigue.

This methodology establishes a coherent flow from interpretable perception to structured reasoning and shared-control actuation. The resulting architecture supports the scenario evaluations described in Section IV.

## IV. Experimental Setup

The evaluation of the proposed shared-control framework was conducted in the CARLA simulation environment, which provides deterministic synchronization between perception, reasoning and vehicle control. All experiments were executed in synchronous mode to ensure frame-level alignment between the camera stream, landmark extraction, neural network inference and actuation. This section describes the simulation environment, the runtime configuration of the CNN and language model, the graphical interface design and the scenario structure used to assess the system.

### A. Software and System Requirements

All experiments were executed using the CARLA autonomous driving simulator, version 0.9.16, configured in synchronous mode to ensure deterministic frame alignment across perception and control modules. The simulation environment was deployed on a standard desktop workstation running a Linux-based operating system. Python 3.12 served as the primary development environment.

Facial landmark extraction was implemented using the dlib library, which provided the 68-point predictor used throughout the perception pipeline. NumPy and SciPy were used for numerical operations and metric computation. The convolutional neural network was implemented in PyTorch, and inference

was performed using the optimized CPU backend without requiring hardware acceleration. The language-model reasoning layer relied on the transformers and mistral-inference packages, operating in low-latency CPU mode.

Integration between CARLA and the perception–reasoning–control pipeline was accomplished using the official CARLA Python API. The graphical interface was developed using OpenCV for real-time rendering of camera frames, landmark overlays and system messages. All components were executed within a single Python runtime to simplify synchronization and logging. This software configuration ensured reproducibility and maintained real-time performance throughout all scenarios.

### *B. Dataset Supplementation Using UTA-RLDD*

To complement the metric-based dataset derived from the CARLA simulation, a small set of real facial images was incorporated to provide additional variation in drowsiness-related facial behavior. Eight images were selected from the UTA-RLDD dataset, which was also utilized in the study titled "Analyzing Model Behavior for Driver Emotion Recognition and Drowsiness Detection Using Explainable Artificial Intelligence" [5]. The chosen images corresponded to a single subject recorded under different stages of drowsiness, allowing controlled variation in eyelid openness, mouth dynamics and overall facial expression as shown in Figure 2.

These images were not used for full-frame CNN training but instead served as supplementary samples for metric extraction and validation. The images were processed through the same 68-point landmark pipeline used in simulation, and EAR, MAR, MOE and PUC values were computed to verify that the metric behavior observed in CARLA aligned with real-world fatigue expressions. Using multiple states from the same driver provided a consistent reference for examining how the metrics respond across increasing drowsiness levels outside the simulated environment.

(a) Alert (b) Slightly Drowsy

(c) Very Drowsy (d) Critical Drowsiness

Fig. 2: Driver images used for metric extraction and validation.

### *C. CNN Runtime and Training Setup*

The convolutional neural network responsible for estimating the driver's drowsiness level was trained offline using the dataset of labeled metric vectors described in Section III. EAR, MAR, MOE and PUC values were normalized prior to training, and an 80 / 20 split was applied for training and validation. The Adam optimizer and early stopping were used to stabilize convergence.

During simulation, the trained CNN operated in real time on a standard workstation environment without requiring specialized hardware. Average inference time remained below the simulator frame interval, allowing the classifier to generate one probability estimate per frame without buffering or delay. The output probability was immediately mapped to the four drowsiness classes shown in Table I, and the resulting state was forwarded to the reasoning layer.

### *D. Simulation Environment*

Three CARLA towns were selected to represent a range of driving conditions. Town01 provided a dense urban layout with traffic lights, intersections and tight corners. Town04 supplied multi-lane highway-style segments suitable for evaluating lane drift and high-speed hazard responses. Town05 offered a mixed arterial layout with moderate curvature and variable lane widths. A forward-facing RGB camera, positioned at dashboard height, delivered a continuous 30 Hz stream for facial landmark extraction.

Low-level throttle, brake and steering commands were issued through the CARLA vehicle interface. Logs were collected for speed, steering angle, drowsiness probability, discrete driver state, reasoning-layer output and system control mode. Each scenario was repeated with several random seeds to reduce sensitivity to traffic initialization and dynamic agent placement.

### *E. LLM Runtime Configuration*

The reasoning layer used a compact Mistral-based language model optimized for low-latency CPU inference. The model processed the classifier probability, the discrete drowsiness class, the four computed metrics and the gradient-based feature-importance vector. Inference times remained sufficiently low to support real-time operation, and deterministic decoding was enforced to maintain consistency across repeated trials. The unified prompt in Listing 1 governed all generated messages. The output of the reasoning layer served as the supervisory instruction for the shared-control arbitration mechanism described in Section III.

### *F. Graphical User Interface*

A graphical user interface was developed to manage the workflow of driver-state analysis, scenario selection and simulation execution. The interface operated as the central control layer for the experimental pipeline and coordinated the interaction between image processing, drowsiness classification, language-model reasoning and CARLA simulation.

The interface first provided an image-upload module through which a user could supply a facial image. The system processed the uploaded image by extracting facial landmarks, computing EAR, MAR, MOE and PUC metrics and classifying the driver's drowsiness state using the trained CNN. The resulting state was then passed to the language model, which produced a concise message for the driver. Both the detected class and the generated message were displayed within the interface before any simulation scenario was selected.

After classification, the interface presented six predefined driving scenarios. Selecting a scenario revealed its corresponding CARLA town, a map preview and a short description summarizing the driving conditions and expected system behavior. A button initiated the live CARLA session associated with the selected scenario. During execution, the interface reported simulation events in real time, including confirmation that CARLA had launched, that the selected town had loaded and that the system was connected to the vehicle. User override actions were also captured; pressing the cancel key ('O' on the keyboard) was immediately reflected in the interface and logged as part of the experiment.

Upon completion of each simulation run, the interface displayed a grid view summarizing the recorded data, including timestamp, steering angle, throttle, brake, vehicle speed and the predicted drowsiness level at each frame. The grid could be downloaded directly through the interface. A dedicated button also opened the output directory containing all generated files. Each simulation produced a folder containing the log file and an images directory storing every rendered frame in chronological order, effectively creating a frame-by-frame visual record of the run.

Output files were named using a consistent format that reflected the scenario and system outcome. For scenarios requiring a driver state as part of the evaluation, filenames took the form `ScenarioID-TownID-DriverState-FinalState`. For scenarios not dependent on driver state, filenames followed the form `ScenarioID-TownID` or `ScenarioID-TownID-FinalState`. The final state corresponded either to system override or user cancellation.

## V. Scenario Design

Six scenarios were constructed to evaluate the system under controlled conditions. Scenarios were chosen to reflect a progression of cognitive degradation, environmental hazards and combined cognitive-environmental risks. Override capability was permitted in non-critical states and disabled when mandatory intervention was required. Table II indicates the CARLA town used to execute each scenario, reflecting the roadway characteristics required for the corresponding experiment.

TABLE II: Scenario execution environment across CARLA towns

| Scenario | CARLA Town |
|---|---|
| Critical-State Detection in a Safe Urban Environment | Town01 |
| Critical-State Unresponsiveness on an Arterial Road | Town04 |
| Lane Drift Toward Roadside Barrier | Town01 |
| Wrong-Lane Intrusion on a Two Way Street | Town05 |
| Red-Light Non-Compliance | Town01 |
| Oncoming-Vehicle Evasion | Town04 |

### *A. Scenario1: Critical-State Detection in a Safe Urban Environment*

This scenario evaluates the system's behavior when the driver is operating a vehicle in a low-speed urban environment with no external hazards present. The vehicle follows a standard city street in CARLA with a speed limit of 30 km/h. The purpose of the scenario is to assess system behavior across the four drowsiness stages when roadway conditions do not impose additional safety challenges.

When the driver is classified as alert, the system issues no warnings and full control remains with the driver. A brief audible tone is produced when the state transitions to slightly drowsy, and a distinct, stronger tone is issued when the driver becomes very drowsy. Both tones serve as awareness cues while allowing uninterrupted driver control.

The critical state activates the complete intervention protocol. In the true-positive case, the language model generates the directive: "Pull over immediately, you are showing signs of critical drowsiness; if you think I am wrong press the cancel button." The system then opens a three-second confirmation window during which the driver may acknowledge the prompt through the designated command. If the driver responds within this window, the system cancels the intervention and driving authority remains with the driver.

If the driver does not respond within the confirmation window, the system initiates autonomous takeover. Hazard lights are activated immediately, and a continuous audible alert replaces the earlier tones. The vehicle then begins controlled deceleration from the 30 km/h cruising speed. The continuous alert persists throughout the deceleration process and stops only once the vehicle has fully come to rest. Override capability is disabled during this sequence to ensure safety after confirmation has lapsed.

### *B. Scenario 2: Critical-State Unresponsiveness on an Arterial Road*

This scenario evaluates system behavior when the driver is traveling on a highway segment at speed 90km/h and no external hazards are present. The vehicle operates in a multi-lane highway environment, and the objective is to assess how the system manages critical drowsiness when lane-changing is required to reach a safe stopping location.

For the alert state, the system issues no warnings and the driver retains full control of the vehicle. When the classifier reports slightly drowsy, the system generates a brief audible tone. A distinct, stronger tone is emitted when the driver

reaches the very drowsy state, signaling more pronounced fatigue while preserving uninterrupted driver authority.

The critical state activates the full intervention protocol. In the true-positive condition, the language model generates the directive: "Pull over immediately, you are showing signs of critical drowsiness; if you think I am wrong press the cancel button." The system then opens a three-second confirmation window during which the driver may acknowledge the prompt and retain manual control. If the driver responds within this interval, no autonomous action is taken.

If the driver does not respond within the confirmation window, the system initiates autonomous takeover tailored for highway operation. Hazard lights are activated immediately, and a continuous audible alert replaces earlier tones. The vehicle first performs a controlled lateral maneuver to move toward the rightmost lane, which is assumed to be the safest location for deceleration on a highway. This maneuver is executed smoothly to avoid abrupt lateral motion and is maintained until the vehicle reaches the shoulder or a suitable stopping position at the far-right side of the roadway.

Once in the rightmost lane or safe area, the vehicle begins gradual deceleration until it comes to a complete stop. The continuous audible alert persists throughout the maneuver and ceases only once the vehicle has fully stopped. Driver override is disabled during this entire sequence due to the confirmed critical condition and the higher speeds associated with highway travel.

### *C. Scenario 3: Lane Drift Toward Roadside Barrier*

This scenario evaluates the system's response when the vehicle is traveling in an urban street environment and begins drifting toward a roadside barrier at a distance that presents an immediate safety risk. Unlike previous scenarios, the driver's drowsiness state is not considered, as the presence of a collision hazard requires direct autonomous intervention regardless of driver condition.

As the vehicle deviates laterally toward the barrier, the system monitors the distance to the obstacle and the lane boundary. When the deviation exceeds a predefined safety margin, the system emits an audible warning. If the drift continues and the vehicle enters a region indicating an imminent impact, the reasoning layer generates the directive: "Watch out," and the warning transitions into a continuous audible alert. This continuous signal remains active throughout the corrective maneuver to reinforce the presence of danger.

At this stage the system performs a non-overrideable intervention. The shared-control module applies corrective steering to counteract the drift and restore proper lane alignment. The continuous audible alert persists until the vehicle has fully reentered the lane and the immediate hazard has been resolved. Because the situation represents a collision-risk scenario, driver override is disabled throughout the intervention. Once the vehicle is safely realigned and no further risk is detected, control is returned to the driver and the alert is silenced.

### *D. Scenario 4: Wrong-Lane Intrusion on a Two Way Street*

This scenario evaluates the system's response when the vehicle unintentionally enters the opposing lane on a two way street. Because the situation presents an immediate collision risk with oncoming traffic, the driver's drowsiness state is not considered in the decision-making process. The system must intervene directly and prevent the wrong-lane intrusion.

As the vehicle begins to cross the centerline, the system detects that the trajectory is inconsistent with proper lane keeping. An audible warning is issued immediately, and hazard lights are activated to alert surrounding vehicles. If the drift continues and the vehicle enters the opposing lane, the reasoning layer generates the directive: "You are switching to the wrong lane, this is a two way street." At this stage, the audible warning transitions into a continuous alert that remains active for the duration of the corrective maneuver.

The system then performs a non-overrideable intervention to return the vehicle to its proper lane. Corrective steering is applied to counteract the lane intrusion and reestablish alignment with the correct lane. The continuous audible alert persists until the vehicle has fully returned to its lane and the risk of collision has been mitigated. Driver override is disabled during this entire process due to the active hazard. Once the lane position is restored and no further danger is detected, control is returned to the driver and the alert is silenced.

### *E. Scenario 5: Red-Light Non-Compliance*

This scenario evaluates system behavior when the vehicle approaches a red traffic signal at normal speed and the driver fails to reduce velocity in preparation for stopping. Because the situation presents a clear traffic-control violation, the system prioritizes safety and issues an intervention request independent of the driver's drowsiness state.

As the vehicle nears the red light, the system monitors speed and distance to the intersection. When the driver does not begin to decelerate within an expected range, the system issues an audible alert and requests confirmation of driver control. The driver is given a two-second window to respond to this request. During this interval, the alert tone continues and the reasoning layer suspends autonomous action until the response window expires.

If the driver responds within the allotted time or activates the override command, intervention is canceled and full driving authority remains with the user. The driver then brings the vehicle to a stop before the red signal. While waiting at the intersection, the system displays a right-turn signal to indicate the driver's intended maneuver once the traffic light turns green, reinforcing that control has been retained by the driver.

If the driver does not respond within the two-second window, the system initiates mandatory intervention. Hazard lights are activated, and the vehicle performs a controlled deceleration to ensure it stops safely before entering the intersection. Driver override is disabled during this phase, as the absence of a response is treated as a control-loss event. Once the vehicle has come to a complete stop and the intersection is secured, control is returned to the driver.

### F. Scenario 6: Oncoming-Vehicle Evasion

This scenario evaluates the system's response when an oncoming vehicle intrudes into the driver's lane and creates an imminent collision risk. Because the hazard is both unexpected and rapidly developing, the driver's drowsiness state is not considered, and the system must intervene immediately without waiting for driver confirmation.

As the vehicle proceeds along a highway segment, the system continuously monitors forward-lane occupancy and relative closing distance. When an oncoming vehicle crosses into the lane and the separation distance decreases to approximately 50 meters, the system initiates an emergency alert. At this point, the reasoning layer issues the directive: "Emergency, oncoming vehicle entering your lane," and an audible warning is activated.

The system then performs a non-overrideable evasive maneuver. Hazard-level response logic activates the right-turn signal, and the vehicle begins a controlled lateral movement toward the right side of the highway. The maneuver is executed smoothly to avoid abrupt steering changes and continues until the vehicle is fully positioned in the rightmost area of the roadway, allowing the oncoming vehicle to pass safely.

Once the vehicle has moved completely out of the collision path and the opposing vehicle has cleared the lane, the right signal is deactivated and the audible alert is silenced. The system then returns full authority to the driver, and the vehicle resumes normal operation. No override is permitted during the avoidance maneuver due to the immediacy and severity of the hazard.

## VI. Results

This section presents the outcomes of the CNN training, the classifier evaluation on real facial images and the execution of all simulation scenarios. Each subsection includes a short interpretation to prepare for a deeper analysis in the Discussion section.

### A. CNN Performance

Table III summarizes the training and validation performance of the custom CNN model. The network reached high accuracy on both sets, with low loss values, indicating stable convergence and strong generalization.

TABLE III: CNN Training and Validation Performance

| Metric | Training | Validation |
|---|---|---|
| Accuracy | 0.9309 | 0.9792 |
| Loss | 0.2107 | 0.1399 |

To further evaluate the classifier, a detailed classification report was generated. Table IV shows precision, recall, F1 score and support for each class, along with macro and weighted averages. The high values across all metrics demonstrate that the CNN correctly distinguishes between the two grouped label categories used during training.

TABLE IV: CNN Classification Report

| Class | Precision | Recall | F1 Score | Support |
|---|---|---|---|---|
| 0 | 0.98 | 0.99 | 0.99 | 541 |
| 1 | 0.98 | 0.93 | 0.96 | 181 |
| Accuracy | | | 0.98 | 722 |
| Macro Avg | 0.98 | 0.96 | 0.97 | 722 |
| Weighted Avg | 0.98 | 0.98 | 0.98 | 722 |

The classifier achieved an overall accuracy of 0.98 on the evaluation set, with particularly strong performance in the majority class (class 0) and solid recall for class 1. These results confirm that the four selected facial metrics (EAR, PUC, MAR, MOE) form a highly discriminative feature set, enabling the CNN to accurately capture the variations associated with driver drowsiness.

### B. Classifier Output Using Real Images

Table V presents the computed EAR, PUC, MAR and MOE values for each real facial image, along with the CNN probability output and final predicted class. These results help verify that the system generalizes beyond simulation and responds meaningfully to natural variations in facial expressions.

TABLE V: Classifier Results for Real Facial Images with Full Metric Breakdown

| Image | EAR | PUC | MAR | MOE | Prob | Class |
|---|---|---|---|---|---|---|
| A0767 | 0.1346 | 0.8135 | 0.0150 | 0.0369 | 0.7011 | Slightly Drowsy |
| A0135 | 0.2076 | 0.1406 | 0.1023 | 0.5496 | 0.9417 | Critical |
| A0446 | 0.0039 | 0.8720 | 0.0252 | 0.0989 | 0.3116 | Alert |
| A0743 | 0.0138 | 0.8583 | 0.0238 | 0.1041 | 0.7250 | Very Drowsy |
| A0748 | 0.0375 | 0.8594 | 0.0220 | 0.0811 | 0.7774 | Very Drowsy |
| A0755 | 0.0400 | 0.8638 | 0.0166 | 0.0797 | 0.5034 | Slightly Drowsy |
| A0757 | 0.0656 | 0.3337 | 0.0966 | 0.5042 | 0.4741 | Slightly Drowsy |
| A0758 | 0.0054 | 0.4125 | 0.1160 | 0.4661 | 0.6130 | Slightly Drowsy |

As seen in Table V, images with lower EAR and higher MAR or MOE values frequently map to higher drowsiness levels, supporting the interpretability and stability of the metric-based CNN classifier.

### C. Simulation Scenario Results

Table VI summarizes the execution results for all six scenarios and all associated test cases. The system was evaluated under both cognitive-state variations (alert, slightly drowsy, very drowsy, critical with user cancellation and critical with AI takeover) and hazard-driven events that required immediate intervention. All simulations were completed successfully, and every scenario produced outcomes fully consistent with the behaviors defined in the scenario design.

Across all conditions, the system consistently transitioned between human and automated control without instability or unexpected behavior. Alerts were issued at the correct thresholds, confirmation windows were handled reliably, and AI takeovers occurred only when the driver remained unresponsive or when hazardous road events made override unsafe. In non-critical drowsiness states, the driver always retained full authority, and the system refrained from unnecessary intervention.

TABLE VI: Simulation Execution Summary for All Scenarios and Cases

| Scenario | Town | Cases Executed | Outcome |
|---|---|---|---|
| Scenario 1 | Town01 | Alert, Slightly Drowsy, Very Drowsy, Critical (AI Takeover and User Cancelled) | All Passed |
| Scenario 2 | Town04 | Alert, Slightly Drowsy, Very Drowsy, Critical (AI Takeover and User Cancelled) | All Passed |
| Scenario 3 | Town01 | Hazard Case Only | Passed (Non Overrideable) |
| Scenario 4 | Town05 | Hazard Case Only | Passed (Non Overrideable) |
| Scenario 5 | Town05 | AI Takeover and User Cancelled | All Passed |
| Scenario 6 | Town04 | Hazard Case Only | Passed (Collision Avoidance Successful) |

## VII. Discussion

The results demonstrate that the proposed perception–reasoning–control pipeline forms a reliable foundation for shared-control driving assistance. The CNN achieved high training and validation performance, showing that the four geometric facial metrics offer a compact yet expressive representation of driver state. The model maintained stable behavior across both simulated faces and real images, which indicates that the metric-based approach generalizes beyond the conditions under which it was trained. This is essential for deployment, where lighting, facial structure and camera position naturally vary. The smooth probability transitions observed across frames further supported the system's suitability for real-time operation.

The classifier results also highlighted the interpretability advantages of using EAR, MAR, MOE and PUC. Each metric correlated predictably with the model's output, and images with strong fatigue indicators consistently mapped to higher drowsiness probabilities. This suggests that the chosen representation not only supports accurate prediction but also preserves transparency, allowing developers and researchers to understand why specific decisions are made.

The reasoning layer produced consistent messages across all conditions, showing that the prompt-restricted language model can enforce structured communication without drift or ambiguity. This was particularly important in critical states, where the system must express urgency without overwhelming the driver. The model followed the intended linguistic rules and tone scaling, demonstrating that controlled LLM behavior is feasible within a real-time driving assistant.

The shared-control arbitration behaved as intended across all tested cases. The system introduced no unnecessary interventions during alert or low-risk states and escalated assertively when the driver entered critical drowsiness or when environmental hazards required immediate corrective action. AI takeover occurred reliably when confirmation was not received, and non overrideable controls were applied only when safety demanded it. Scenario execution remained consistent across all runs, with no collisions and no deviations from the designed logic.

The successful completion of all scenarios also validates the interaction between perception and control. In particular, the system demonstrated that it can translate probabilistic drowsiness estimates into meaningful, context-aware actions. The transitions between human and automated control were smooth, and the vehicle remained stable during lane changes, deceleration and evasive maneuvers. This behavior is critical for shared-control systems, where abrupt or unpredictable authority shifts can introduce additional risk.

The full video and log recordings captured during the experiments provide concrete evidence of system determinism and repeatability. These outputs form a basis for future quantitative analysis and can support downstream performance benchmarking or real-world validation campaigns.

The Discussion reflects that the integrated approach is coherent, interpretable and robust. The combination of metric-driven perception, structured reasoning and rule-based shared control offers a promising direction for drowsiness mitigation systems. The following sections on limitations and future work expand on the aspects that require refinement and the opportunities for extending the framework beyond simulation.

## VIII. Limitations

The system evaluation was conducted entirely within the CARLA simulator, and no hardware-in-the-loop or real-vehicle testing was performed. The absence of real sensor data introduces constraints related to lighting variation, head pose dynamics, occlusion, and camera noise that were not represented in simulation. The drowsiness-classification model was evaluated on a limited set of static real facial images, which does not reflect continuous temporal behavior such as progressive eyelid closure, micro-sleeps or repeated yawning. The dataset does not provide wide demographic or environmental diversity, and the system has not yet been validated on extended video sequences.

The simulation performance depends on available computational resources. Frame-rate drops occurred in dense environments, and the reasoning layer currently operates using predefined prompts without fully dynamic on-board language-model inference. The shared-control framework has not been tested with human participants, and driver reactions to alerts, confirmation prompts and takeover events remain unvalidated. These constraints limit the generalizability of the results outside the controlled simulation environment.

## IX. Future Work

Future extensions include integration of real-time video streams for continuous driver-state monitoring and temporal modeling of blink dynamics and gaze behavior. Additional sensing modalities such as physiological signals, infrared eye tracking and EEG measurements will be incorporated to improve robustness in ambiguous or low-visibility conditions.

The cognitive-state model will be expanded to include emotional indicators relevant to driving performance.

Hardware-in-the-loop testing will be introduced to evaluate the system with real cameras, vehicle controllers and driver inputs. Validation using a physical driving simulator and controlled user studies will enable analysis of driver reaction patterns and human–automation interaction under takeover conditions. Reinforcement-learning-based shared-control strategies will be explored to replace fixed arbitration rules with adaptive policies that adjust control authority based on driver behavior and situational context.

## X. Conclusion

This work presented a shared-control framework that integrates interpretable facial-metric perception, convolutional-neural-network classification, a structured language-model reasoning layer and rule-based arbitration within the CARLA simulation environment. The system establishes an end-to-end pipeline capable of assessing driver drowsiness, issuing graded feedback and executing autonomous intervention when required. The approach emphasizes transparency by grounding driver-state estimation in geometric facial metrics and ensuring that all corrective actions follow explicit, predefined control rules.

The CNN classifier achieved high accuracy and produced stable probability outputs across both simulated and real facial inputs, demonstrating that EAR, MAR, MOE and PUC form an effective and reliable representation for multi-level drowsiness assessment. The use of interpretable metrics contributed to consistent classification boundaries and predictable state transitions, which are essential for real-time deployment in safety-critical systems. The reasoning layer converted numerical predictions into concise driver-facing messages that adhered strictly to severity constraints, enabling a coherent interaction channel between perception and control. The arbitration module then mapped these reasoning outputs to structured authority decisions, ensuring that control actions remained aligned with both the cognitive state of the driver and the surrounding roadway context.

All six simulation scenarios executed as intended and validated the integrated framework under diverse conditions, including low-speed urban driving, high-speed highway segments and safety-critical hazards. The system generated stable behavior during alerting, confirmation handling, lane correction, autonomous braking and evasive maneuvers. The successful execution of non-overrideable interventions in hazardous contexts further demonstrated the reliability of the arbitration logic when safety constraints took precedence. Recorded logs and video outputs confirmed deterministic execution, consistent authority transitions and reproducible system behavior across all trials.

The results indicate that combining interpretable driver-state estimation with lightweight reasoning and structured shared-control policies offers a practical and coherent path toward mitigating drowsiness-related risks in semi-autonomous driving. The unified pipeline enables active monitoring without sacrificing driver authority in non-critical conditions, while ensuring predictable and safety-focused intervention when necessary. The architecture is modular, transparent and compatible with real-time operation, making it suitable for integration into broader cyber-physical vehicle platforms.

The framework developed here provides a foundation for future extensions involving continuous video processing, multimodal sensing, hardware-in-the-loop validation and adaptive control strategies capable of personalizing intervention thresholds. These directions have the potential to enhance robustness, extend applicability beyond simulation environments and support the development of next-generation shared-control systems that balance human oversight with dependable automated safety mechanisms.

## References


[1] K. Dwivedi, B. Biswal, and A. Kumar, "Deep convolutional neural networks for driver drowsiness detection," *IEEE Access*, vol. 7, pp. 184 951–184 960, 2019.

[2] S. Reddy and M. Gupta, "Driver drowsiness detection using perclos and eye aspect ratio features with deep learning," in *2020 IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 2034–2039.

[3] H. Alshaqaqi, A. Al-Mohannadi, and S. Ahmed, "A hybrid cnn-lstm framework for real-time driver fatigue estimation," *Pattern Recognition Letters*, vol. 152, pp. 256–263, 2021.

[4] H. Abbas, S. Hamid, and W. Zhang, "Real-time driver drowsiness detection using transformer architectures," *International Journal of Computer Vision and Pattern Analysis*, vol. 13, no. 1, pp. 22–35, 2025.

[5] J. Caballero, D. Martins, and E. Frazier, "Analyzing model behavior for driver emotion recognition and drowsiness detection using explainable artificial intelligence," in *Proceedings of the 11th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS)*, 2025, pp. 145–156.

[6] T. Nguyen and K. Sohn, "Vision-based driver monitoring: A survey of fatigue, distraction and behavioral analysis techniques," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 10 245–10 260, 2022.

[7] M. Marcano, M. Spenko, and P. Jayakumar, "A review of shared control for automated vehicles: Theory and applications," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 6, pp. 485–497, 2020.

[8] M. Cunningham and Y. Demiris, "Human–automation shared control: A review of theory and automotive applications," *Annual Reviews in Control*, vol. 51, pp. 252–265, 2021.

[9] Y. Chen, C. Li, Q. Yuan, J. Li, Y. Fan, X. Ge, Y. Li, F. Gao, and R. Zhao, "Cockpit-llama: Driver intent prediction in intelligent cockpit via large language model," *Sensors*, vol. 25, no. 64, 2025.

[10] J. Park, S. Kim, and J. Bae, "Real-time simulation for autonomous driving safety evaluation using carla," in *2022 IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2022, pp. 3471–3478.

[11] R. Patel and A. Singh, "Vision-based drowsiness detection using facial landmark dynamics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1254–1263.

[12] L. Zhou and S. Lee, "Real-time driver fatigue estimation using deep spatiotemporal networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2443–2454, 2020.

[13] P. Ghosh and J. Kim, "A survey on driver behavior understanding using vision-based approaches," *Pattern Recognition*, vol. 124, p. 108523, 2022.

[14] Y. Huang and J. Park, "Learning shared control policies for vehicular safety using reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1230–1237, 2021.

[15] F. Rahman and A. Smith, "Driver inattention detection using multimodal fusion of facial landmarks and gaze estimation," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 1502–1509.