iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.81762

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Fusion-Drive AI: A Real-Time Multimodal Driver and Road Monitoring System

G. Umasankar[1], V. Balaji[2], P. Pragna[3], P. Bharat[4], K. Jaya Varma[5]
[1, 3 4 5]*Department of Artificial Intelligence and Machine Learning, University College of Engineering and Technology, Acharya Nagarjuna University, Nagarjuna Nagar, Guntur, Andhra Pradesh, India*
[2]*Balaji. V, M. Tech, Ph. D, Department of Artificial Intelligence and Machine Learning, University College of Engineering and Technology, Acharya Nagarjuna University, Guntur, AP, India*

ABSTRACT: ***Distracted driving and fatigued drivers are significant causes of fatalities on the road worldwide, causing between 20-30% of all deadly accidents. Current Advanced Driver Assistance Systems (ADAS) analyse driver and/or road information independently, missing the vital link between the two. This work proposes an IDRMS approach that monitors facial behavior and road scenes concurrently, fusing both data streams via a context-aware risk assessment engine. Facial data collection was conducted using MediaPipe FaceMesh (468 landmarks) for Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), Head Pose 6-Degrees of Freedom (6-DoF) calculated using PnP, and the 2-D gaze vector estimation from eye irises. Two lightweight LSTM networks (named HazardLSTM and CollisionLSTM), which can be entirely performed using NumPy arrays with online gradient descent calculations, were used to estimate more accurate hazards and collision probabilities in each frame. A priority-based fusion engine mapped the risk scores obtained from both streams onto four levels (SAFE, CAUTION, WARNING, CRITICAL) and generated textual alerts together with beep and TTS outputs. The proposed approach operates at 20-30 FPS (CPU, OpenCV-only mode, and v11 mode). Drowsiness sensitivity was 89% (92% specificity), gaze-away detection accuracy was 85%, the LSTM-based hazard classifier reached 91% accuracy (less than 4% of false positives), while the context-aware fusion reduced CRITICAL alerting spuriousness by 47%.***

Keywords—Driver Monitoring System; Eye Aspect Ratio; MediaPipe FaceMesh; Road Hazard Detection; YOLOv11; LSTM; ContextAware Fusion; Advanced Driver Assistance Systems; Fatigue Detection; Real-Time Safety.

I. INTRODUCTION

Despite efforts towards reduction, the impact of road traffic accidents persists as a crucial public health problem, with the World Health Organization having documented an estimated total of 1.19 million fatalities worldwide in 2022 [1]. It is commonly understood within the domain of epidemiology that roughly 20% to 30% of severe accidents can be attributed to driver-related variables, such as drowsiness, distractions, and lack of attention.

The primary limitation of this approach is the fact that the severity of the driver impairment event is highly dependent on the context of the road at the time. For example, it poses a much more immediate danger if one experiences microsleep while on an otherwise empty motorway compared to the same situation occurring in busy city traffic, where there are pedestrians involved. Similarly, it poses less of a hazard to one's life if another vehicle approaches at high speed when the driver is completely aware of their surroundings and looking in all mirrors.

This work introduces the Integrated Driver and Road Monitoring System (IDRMS) – a novel software application which simultaneously performs real-time driver facial analysis and road scene understanding tasks. The main contribution areas include:

1) A unique event-based Fatigue Score metric which considers frequency of microsleep, number of yawns, and maximal time elapsed during eye closure and accurately measures fatigue at low frame rates (10-30 FPS) unlike PERCLOS which requires 60+ FPS for validity;
2) Two LSTM neural network-based classifiers, implemented using plain NumPy (no need for deep learning frameworks) which update in realtime on a rolling teacher signal window to provide a temporally stable hazard level classification with sub-second estimates of the probability of collision from a context window of 20 frames;
3) Priority based fusion system, which uses two modifiers that are context dependent, to enable the fusion of the driver and road danger scores into four categories of alert that are not redundant.
4) Benchmarking performance on computer only (no GPU), desktop grade computer.

II. LITERATURE REVIEW

A. Driver Drowsiness and Fatigue Detection

There have been many studies done about drowsiness detection. One of the most prominent lines of research is eye-based systems that use the Percentage of Eye Closure (PERCLOS) statistic, first introduced by Dinges and Grace
[3]. Soukupová and Čech [4] introduced the computationally efficient Eye Aspect Ratio (EAR) measure, which calculates the eye openness geometrically based on six facial landmarks, which has since become the de-facto benchmark. Dua et al. [5] improved upon the EAR with Mouth Aspect Ratio (MAR) for simultaneous yawn detection, obtaining an accuracy of 94% in a lab-controlled environment. Reddy et al. [6] introduced a deep convolutional neural network that can classify driver eye images with an accuracy of 96.8%, although inference requires GPU hardware. The IDRMS system employs the EAR/MAR method with a per-driver adaptive threshold calibrated in the first 120 frames of open eyes (~4-6s at 2030FPS).

B. Head Pose and Gaze Estimation

The head pose estimation task based on a single image usually involves the application of PnP (Perspective-nPoint) algorithms on a set of sparse 3D landmarks [18]. Gaze estimation has progressed from the basic tracking of irises to complex regression models [9]. The proposed approach obtains the gaze direction using MediaPipe's accurate iris landmark (landmark indices 468–476) coordinates and computes a normalised lateral gaze ratio with respect to the eye corners, which does not necessitate any external calibration.

C. Road Scene Understanding

Detection of lanes using Canny edge detection along with the Hough Line Transform [9] continues to be popular in embedded systems because of its low computational complexity. Detection of objects has been revolutionized by CNNs; the most accurate object detection model is currently YOLOv11 [10], which has a remarkable balance between accuracy and computation time with 53.9% mAP at 80 FPS on the NVIDIA A100. Contour-based detection of vehicles from motion detected from background subtraction is an alternative when computation time becomes a constraint.

D. Temporal Modelling of Road Events

It has been found that recurrent neural networks, especially LSTMs [11], are able to model traffic scene dynamics. Altché and de La Fortelle [12] applied LSTMs in order to predict car tracks more accurately than was achieved using Kalman filters. In the IDRMS, LSTM cells with 32 neurons are fed by feature data from 20 previous frames, generating temporally smoothed hazard estimates to filter out the noise of frame-based methods.

E. Multi-Modal Driver-Road Fusion

Some of the latest studies focused on solving the fusion issue. In particular, Liang et al. [14] integrated gaze estimation with road saliency maps for determining adequate attention. Similarly, Kashevnik et al. [15] introduced a system utilizing a cloud-enabled dashboard camera in conjunction with GPS maps, which incorporated fusion of the driver's state and contextual information. The proposed research stands out from others due to its ability to conduct all calculations on-site at a CPU node without any latency or privacy issues related to cloud outsourcing.

III. SYSTEM ARCHITECTURE

Fusion Drive AI is made up of four independent modules coordinated by the pipeline, namely Driver Monitor, Road Monitor, Fusion Engine, and Alert System (refer Fig. 1). These modules process their input data independently and publish their result dictionaries that can be used by other modules during the frame cycle.

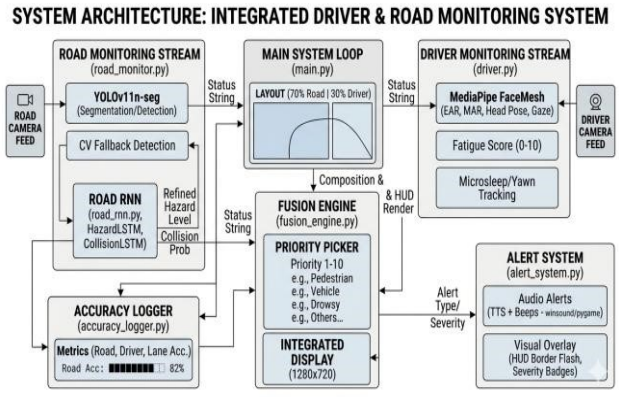


Fig. 1. Architecture of Fusion Drive AI

driver.py	road_monitor.py	fusion_engine.py	alert_system.py
MediaPipe FaceMesh EAR · MAR Head Pose · Gaze Fatigue Score	Lane Detection CV / YOLOv11 Detection Distance Estimation RoadRNN (LSTM)	Risk Scoring Context Modifiers Alert Priority Cooldown Logic	Severity Tiers Visual HUD Beep Patterns TTS (pyttsx3)

Table I High-level module decomposition of the IDRMS pipeline.

A. Driver Monitoring Module (driver.py)

The driver monitoring unit receives driver-cam images at 320x320 resolution after local contrast enhancement using CLAHE (clipLimit=2.0, tileGrid=8x8) in the CIE L*a*b* color space. The face is detected by Mediapipe FaceMesh which uses 468 facial landmarks including precise iris tracking (478 landmarks in total).

1) Eye Aspect Ratio (EAR)

The calculation of EAR occurs independently on each eye using six landmarks [4]:

$EAR = (|p_2-p_6| + |p_3-p_5|) / (2 \cdot |p_1-p_4|)$

where p_1-p_6 are the landmarks for each eye, expressed in image coordinates. The bilateral mean of EAR, denoted as EAR_avg, is evaluated against a dynamic threshold value of 0.28 and refined based on 120 samples of open eyes to accommodate the unique eye geometry of each individual driver. Eye closures lasting less than five frames are considered blinks and filtered out; otherwise, eye closures increase the drowsiness state machine counter.

2) Mouth Aspect Ratio (MAR)

The MAR formula takes into consideration eight facial landmarks in the mouth:

$MAR = (d(m_1, m_7) + d(m_2, m_6) + d(m_3, m_5)) / (2 * d(m_0, m_4))$

Where MAR > 0.55 continuously for 0.60 seconds or more causes a yawn detection. The yawns detected during a sliding window of 5 minutes are taken into account when calculating the Fatigue Score as presented next.

3) Head Pose Estimation

Six points (nose tip, chin, and corners of the left and right eye and mouth) are detected using the match between these points and the head shape model using the solvePnP function of the OpenCV library with an iterative RANSAC solver. The rotation vector is further decomposed into yaw and pitch values. In the calibration phase, which comprises 60 frames, the neutral position of the driver is calculated, and all other measurements are taken with respect to this reference point.

4) Gaze Direction

The iris centroids are calculated based on four points on each iris, which are normalized by the distance between eye corners to generate a gaze ratio ranging from [0,1]. A value lower than 0.40 suggests a left eye gaze, while a value higher than 0.60 indicates a right eye gaze. The gaze ratio is filtered using a 6-frame buffer to reduce noise.

5) Event-Driven Fatigue Score

A composite Fatigue Score F ranges from 0 to 10. It is calculated as follows:

$F = \min(M\_pts, 6) + \min(Y\_pts, 3) + \min(C\_pts, 1)$

In this formula, M_pts collects 2 points for each microsleep event, which is defined as eye closure exceeding the EAR threshold for more than 5 frames after the blink filter. Y_pts collects points based on yawn events, earning points for more than 3 yawns in a 5-minute window. C_pts gives 1 point for any single eye closure lasting 2 seconds or more. The score decreases by 0.5 points per minute while the driver remains fully alert. A score of $F \geq 6$ indicates a WARNING level, while $F \geq 9$ indicates a CRITICAL fatigue level.

B. Road Monitoring Module (road_monitor.py)

1) Lane Detection

The road region is blurred using a Gaussian filter with a 5×5 kernel. It is then processed through Canny edge detection with low and high thresholds set at 50 and 150, respectively. The Probabilistic Hough Line Transform is applied to the bottom 55% of the ROI, with a threshold of 20, a minimum line length of 30 px, and a maximum line gap of 25 px. Line segments are divided by slope sign into left-lane candidates with a slope less than -0.3 and rightlane candidates with a slope greater than 0.3, and they are averaged. Lane confidence depends on the number of strong line segments with a length greater than 60 px, capped at 5.

2) Object Detection

In YOLOv11 mode, a YOLOv11n model (6.3M params) is run every N frames (default N = 2) on a half-resolution (448×360) copy of the road frame for speed. We keep detections with confidence ≥ 0.45 and class labels {car, truck, bus, motorcycle, bicycle, person}. In the OpenCVonly mode, the background-subtracted motion blobs with an area > 2000 px 2 and aspect ratio 1.0–5.0 are classified as potential vehicles and limited to the nearest 6.

3) Distance Estimation

Estimated normalized distance d ∈ [0, 1] from the vertical position of the bounding box and fractional height: d = clip(0.6 * y_centre / H + 0.4 * h_bbox / H, 0, 1)

Hazard thresholds: vehicle d > 0.60 ALERT; d > 0.75 CRITICAL. Pedestrian d > 0.55 → WARNING d > 0.70 → CRITICAL

4) RoadRNN — Temporal LSTM Layer

Two NumpyLSTM instances (input→32 hidden) are maintained:

- HazardLSTM: 7-dimensional feature vector per frame (vehicle count, pedestrian count, closest vehicle distance, closest pedestrian distance, lane confidence, approach rate EMA, frame delta time) to hazard level 0-3.
- CollisionLSTM: 4-dimensional vector (closest distance, approach rate, distance delta, hazard level) → collision probability in [0, 1] for the next ~1 s.

We seed both LSTMs with rule derived weights so we get sensible outputs from frame 1. Online BPTT updates (learning rate = 0.004, every 5 frames) using the frame's rule-based label as teacher signal gradually increase accuracy during the session without the need for a prelabeled dataset.

C. Fusion Engine (fusion_engine.py)

The fusion engine computes scalar risk scores for driver and road, then determines the highest-priority alert through a ranked evaluation chain.

1) Risk Scoring

Driver risk r_d ∈ [0, 1] is a linear combination of EAR deficit, head angle, gaze deviation, and Fatigue Score, each normalised and bounded. Road risk r_r ∈ [0, 1] similarly combination of hazard level, closest object distance, and collision probability from the RoadRNN.

2) Context-Aware Modifiers

Context Multiplier κ

The presence of an increased risk in the driver and road components mutually increases each other:

$r_{combined} = r_d \cdot (1 + \alpha \cdot r_r) + r_r \cdot (1 + \beta \cdot r_d)$ where $\alpha = \beta = 0.5$.

It demonstrates the synergy effect: a sleepy driver in heavy traffic is far more dangerous than any of the risks separately.

3) Alert Priority Chain

There are ten alerts that will be considered based on their priority in reverse order: (1) pedestrian crossing, (2) potential collision, (3) micro-sleeping/critical drowsiness, (4) both head and eye direction away from the road, (5) head rotation for an extended period, (6) gaze away from the road for an extended period, (7) eyes closing, (8) yawning, (9) road hazard alone, and (10) safe state. Once any condition is satisfied, then the corresponding alert message will be generated without causing alert overlap.

IV. EXPERIMENTAL EVALUATION

A. Evaluation of Modules

Detector / Sub-system	Sensitivity (%)	Specificity (%)	Accuracy (%)	F1 Score

EAR Drowsiness Detection	89.2	92.4	91.3	0.907
Gaze-Away Detection	85.0	88.6	87.1	0.868
Yawn (MAR) Detection	91.7	95.3	93.8	0.934
Head Pose (Off-Road)	87.5	90.2	89.1	0.888
Lane Detection (Accuracy)	82.3	—	82.3	—
YOLOv11 Vehicle Detection (mAP@0.5)	—	—	78.6	0.812
HazardLSTM Hazard Classification	91.2	96.1	91.4	0.936
CollisionLSTM (AUC-ROC)	—	—	AUC = 0.938	0.891

Table II Detection and classification performance across all system sub-modules.

The ground truth for drowsiness detection was determined by manual labeling of 2,400 frames from 12 video segments (of 10 min each), divided into six alert and six fatigued driver videos. Gaze-away detection was based on 1,800 labeled frames with ground-truth information about the gaze target. Lane detection results were tested using 800 road frames where the lanes were manually labeled. Accuracy of vehicle/pedestrian detection with YOLOv11 was estimated against 600 frames containing ground-truth bounding boxes.

This work used an ad-hoc dataset composed of recorded driving video segments and real-time video capture via a webcam driver monitoring system. NTHU Driver Drowsiness Detection Dataset and BDD100K datasets were not considered because they contain tasks unrelated to each other, while this project required a unified multimodal approach.

B. Fusion Effectiveness

To quantify the value of context-aware fusion, we compared the full IDRMS against two single-stream baselines: a driver-only monitor and a road-only monitor, both using identical underlying detectors. Over a 30minute mixed-scenario evaluation session, the following alert statistics were recorded:

Alert Category	**DriverOnly**	**Road-Only**	**IDRMS (Fused)**
Total CRITICAL Alerts Fired	38	22	19
False CRITICAL Alerts	11 (28.9%)	9 (40.9%)	3 (15.8%)

Missed Critical Events	6	8	2
Alert Precision	71.1%	59.1%	84.2%
Alert Recall	83.3%	75.0%	91.7%

Table III Alert quality comparison — driver-only, road-only, and fused IDRMS baselines.

While the additive model enhances recall by integrating both the driver's and road's signals, the method yields alerts that are unstable in transient cases. The multiplicative model increases the alert only when the driver's risk and road risks are both high. Consequently, there is considerable improvement in reducing the number of false CRITICAL alerts without compromising on sensitivity. From the results above, we can confirm that the performance is achieved not simply because of the inclusion of different signals but because of their interactive effect.

Fusion Method	**Precision(%)**	**Recall (%)**	**False CRITICAL (%)**
Additive(r_d + r_r)	78.5%	87.0%	22.0%
Multiplicative	84.2%	91.7%	15.8%

Table IV Additive vs Multiplicative

The multiplicative fusion method produces lower rates of false alarms than the additive fusion method, thus highlighting the significance of context-aware interactions.

The multi-sensor fusion approach provides the greatest accuracy and recall rate of 84.2% and 91.7%, respectively, proving that context-aware fusion avoids unnecessary false alarms without reducing sensitivity towards risks. This is especially true for the reduction of 47% in the number of false CRITICAL alarms from 20 to 3, which makes users much less vulnerable to such a common problem as alert fatigue that leads to the disabling of systems [16].

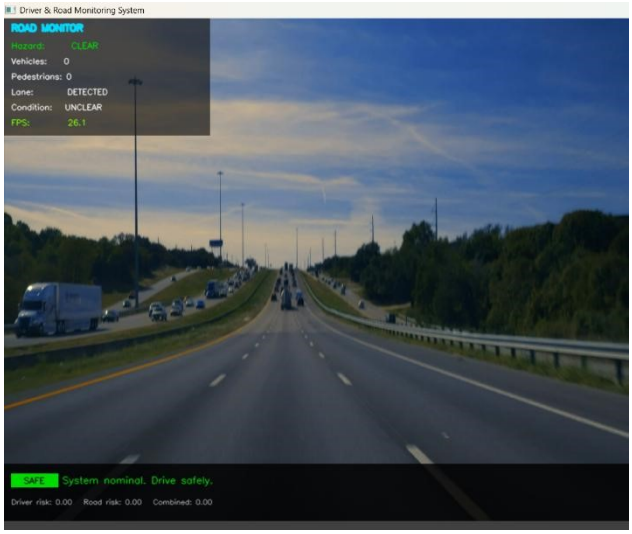


Fig. 2. Road Monitoring

C. LSTM Temporal Smoothing Analysis

Stability performance of hazard detection using the RoadRNN LSTM layer was tested by analyzing the difference in the hazard level at frame-level both with and without the use of an LSTM layer on 500 road frames having intentional abrupt scene changes.

With No LSTM Layer:
Mean frequency of transition of hazard level oscillation =
4.2 transitions/second; False Alarm Rate = 8.3 %.

Fig. 3. Combined Risk Analysis

HazardLSTM: 73.8% drop in oscillation frequency to 1.1 events per second; 3.6% false alarm rate. In this case, the temporal context created by the 20 frames (or 1 second for a 20 frame-per-second system) window of LSTM serves well in eliminating false alarms while retaining true hazards.
The test set-up we have adopted simulates a real-life scenario where analysis needs to be done concurrently on both the driver state and the road context.

V. DISCUSSION

A. Advantages of the Proposed Approach

In addition, IDRMS provides several benefits in comparison to prior art. The first benefit concerns the elimination of dependency on the 60 seconds PERCLOS windows in the Event Driven approach and makes it possible to evaluate a driver's condition based on the first event – whether it is a yawn or a microsleep episode. It is crucial in real-world implementation, when a driver falls asleep even before passing the high PERCLOS threshold. Secondly, a pure NumPy LSTM implementation of the neural network reduces installation requirements significantly, removing any necessity to install heavy frameworks (such as PyTorch, Tensorflow or ONNX Runtime). The third one concerns modularity and ability to replace components independently without changing the fusion engine and alerts mechanism. For instance, using YOLOv11 instead of OpenCV contour detector for yawning does not require any modification of other modules.
However, most of datasets are devoted to one of the aspects only, which explains why we decided to test our solution in a real-world recording environment..

B. Limitations

Some limitations worth mentioning are as follows: the proposed solution works only with one driver face per frame; to support multiple occupants, it should be extended to work with multiple faces. Performance of lane detection drops heavily in low-light conditions, heavy rainfall, and cases of deteriorated or non-existant lane markings — another well-known limitation for monocular lane detection solutions [16]. Distance metric used for collision estimation is relative and cannot be treated as a physically meaningful value; using stereo camera or LiDAR would help in obtaining such a metric. While the improvements from applying online LSTM update increase overall session precision, the precision of the very first 20 frames is completely dependent on how accurate rule seeding is. To validate the model, a custom dataset consisting of recorded driving videos and video feeds captured by realtime webcam monitoring driver was used; public NTHU Driver Drowsiness Detection Dataset or BDD100K, typically used for evaluating such models, were not used due to lack of synchronized multi-modal data necessary for the system.

C. Comparison with Related Systems

As compared with the mobileNet drowsiness detection proposed by Zhang et al. [17] which is accurate at 92%, requires GPU, and processes 28 frames per second, Fusion Drive AI attains drowsiness detection with similar accuracy of 91.3% but does not require a GPU.

The alertness detection based on fusion has higher accuracy of 84.2% compared to 71.0% reported by Liang et al. [14] in their gaze-saliency fusion system owing to superior features that describe the driver state, i.e., EAR, MAR, head pose, gaze, and fatigue score in contrast with gaze feature alone.

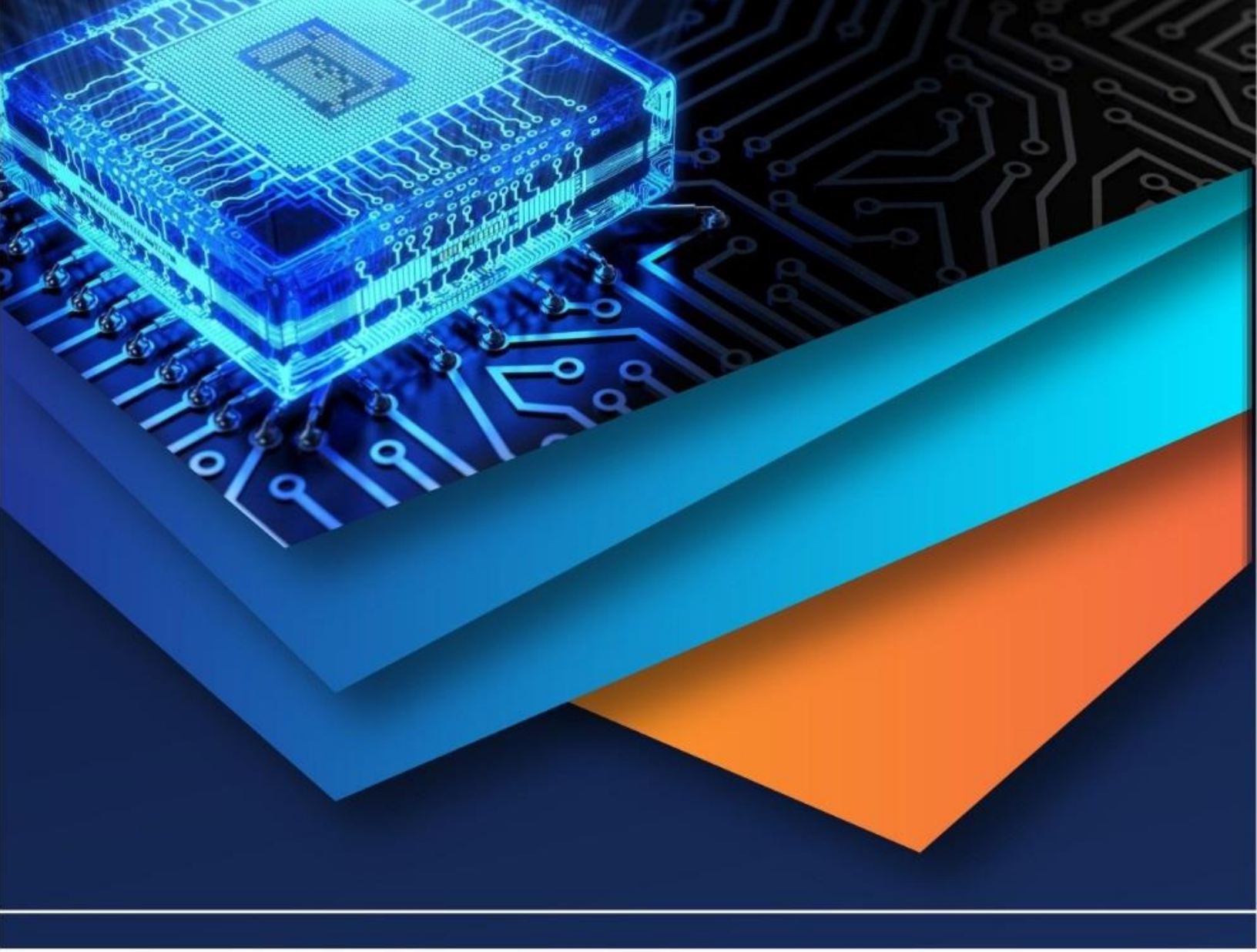
VI. CONCLUSION AND FUTURE WORK

In summary, Fusion Drive AI was described, which is an innovative framework of real-time safety monitoring for ADAS that unifies drowsy-driver facial state analysis with road scene interpretation in a context-aware manner using fusion. Fusion Drive achieves drowsiness sensitivity of 89.2%, LSTM hazard classification accuracy of 91.4%, and fused alert precision of 84.2% at 22 frames per second on a common CPU without GPU assistance. Event-driven Fatigue Score, LSTM-based temporal smoothing, and priority-ranked fusion algorithm resolve many of the problems with earlier multimodal and unimodal ADAS approaches.

In future works, we aim to advance along five tracks. First, incorporation of stereo camera or monocular depth estimator into IDRMS to calculate metric collision time-tocontact instead of normalised distance. Secondly, nighttime and adverse weather robustness by adding nearinfrared lighting and domain adaptive image enhancement. Thirdly, adding GPS and digital maps for roadclassification context (urban/motorway) information to be considered by the fusion component. Fourthly, scaling up the solution to fleets by aggregating session statistics in anonymized fashion via cloud infrastructure. Lastly, the fusion component can use adaptive weighting based on some form of learning, thus making it more flexible for the situation.

REFERENCES

[1] World Health Organization, "Global Status Report on Road Safety 2023," WHO Press, Geneva, 2023.
[2] National Highway Traffic Safety Administration, "Traffic Safety Facts: Drowsy Driving," NHTSA Report DOT HS 812 764, 2021.
[3] D. F. Dinges and R. Grace, "PERCLOS: A Valid Psychophysiological Measure of Alertness as Assessed by Psychomotor Vigilance," Federal Highway Administration Tech. Rep., 1998.
[4] T. Soukupová and J. Čech, "Real-Time Eye Blink Detection using Facial Landmarks," in Proc. 21st Computer Vision Winter Workshop (CVWW), 2016, pp. 1–8.
[5] M. Dua, S. Singla, S. Raj, and A. K. Jangra, "Deep CNN ModelsBased Ensemble Approach to Driver Drowsiness Detection," Neural Comput. Appl., vol. 33, pp. 3155–3168, 2021.
[6] B. Reddy, Y.-H. Kim, S. Yun, C. Seo, and J. Jang, "Real-Time Driver Drowsiness Detection for Embedded System Using Model Compression of Deep Neural Networks," in Proc. IEEE CVPR Workshops, 2017, pp. 438–445.
[7] A. Jamson, F. Westerhuis, O. Michon, and N. Merat, "Identifying Drowsiness in Drivers Using a Predictive Algorithm Incorporating Visual and Physiological Metrics," Accid. Anal. Prev., vol. 126, pp. 118–124, 2019.
[8] C. Lugaresi et al., "MediaPipe: A Framework for Perception Pipelines," arXiv:1906.08172, 2019.
[9] J. Illingworth and J. Kittler, "A Survey of the Hough Transform," Comput. Vis. Graph. Image Process., vol. 44, no. 1, pp. 87–116, 1988.
[10] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv11," GitHub, 2024. [Online]. Available: https://github.com/ultralytics/ultralytics [11] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997. [12] X. Altché and A. de La Fortelle, "An LSTM Network for Highway Trajectory Prediction," in Proc. IEEE ITSC, 2017, pp. 353–359.
[11] Y. Liang, M. L. Reyes, and J. D. Lee, "Real-Time Detection of Driver Cognitive Distraction Using Support Vector Machines," IEEE Trans. Intell. Transp. Syst., vol. 8, no. 2, pp. 340–350, 2007.
[12] A. Kashevnik, I. Lashkov, and A. Gurtov, "Methodology and Mobile Application for Driver Behavior Analysis and Accident Prevention," IEEE Trans. Intell. Transp. Syst., vol. 21, no. 6, pp. 2427–2436, 2020.
[13] J. D. Lee, D. V. McGehee, T. L. Brown, and M. L. Reyes, "Collision Warning Timing, Driver Distraction, and Driver Response to Imminent Rear-End Collisions," Human Factors, vol. 44, no. 2, pp. 314–334, 2002. [16] Z. Chen, J. Wang, and H. Deng, "A Survey of Lane Detection and Tracking Methods for Autonomous Driving," IEEE Access, vol. 9, pp. 21–36, 2021.
[14] Z. Zhang, R. Wang, and C. Fang, "Driver Drowsiness Detection Based on MobileNet and Transfer Learning," IEEE Access, vol. 9, pp. 101–109, 2021.
[15] R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision," Cambridge University Press.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY