# ADAS FOR TWO WHEELERS USING NVIDIA JETSON NANO

***Project Reference No.:** 49S_BE_4538*


| | | |
|---|---|---|
| *College* | : | *Basaveshwar Engineering College, Bagalkote* |
| *Branch* | : | *Department of Electronics and Communication Engineering* |
| *Guide(s)* | : | *Dr. Somu Parande* |
| *Student(s)* | : | *Mr. Vaibhav Bangalore* |
| | | *Mr. Amar Patil* |
| | | *Ms. Banashree Rayanagoudar* |
| | | *Ms. Anusha Murgod* |



**Keywords**

ADAS, YOLOv8, Two-Wheelers, Nvidia Jetson Nano, Edge Computing Simulation, Forward Collision Warning, Blind Spot Detection, Lane Departure Warning, Overtaking Alert.


**Introduction:**

The global automotive landscape is undergoing a profound transformation driven by Advanced Driver Assistance Systems (ADAS). Historically, the development of ADAS has been heavily skewed towards passenger cars due to the high cost of sensor suites and computational hardware. However, the safety imperative for Powered Two-Wheelers (PTWs) is arguably far greater. In India, the world's largest two-wheeler market, PTWs account for the highest share of road accidents and fatalities due to the lack of a structural cage, dynamic instability, and chaotic mixed-traffic environments. This project aims to bridge this safety gap by developing a robust, purely vision-based ADAS targeted for edge deployment on the NVIDIA Jetson Nano. To ensure algorithmic integrity and bypass current hardware iteration bottlenecks, the core logic is currently being rigorously architected and validated in a high-fidelity software-in-the-loop simulation environment, proving that AI at the edge can process video feeds to warn riders of imminent threats.

**Objectives:**

1. **Edge-Optimized Vision Stack**: Development of a lightweight perception pipeline using YOLOv8 tailored for the NVIDIA Jetson Nano, achieving real-time performance for two-wheeler applications.
2. **Sensor-less Depth Estimation**: Implementation of a geometric heuristic using Inverse Perspective Mapping (IPM) to estimate Time-To-Collision (TTC) from monocular camera feeds, eliminating the need for expensive radar or LiDAR
3. **Dynamic Blind Spot Monitoring**: A robust tracking algorithm specifically designed to detect overtaking maneuvers and blind spot intrusions in chaotic, mixed-traffic environments
4. **Mathematical framework**: for estimating inter-vehicle distance and relative velocity (Time-To-Collision) using monocular depth estimation and Inverse Perspective Mapping (IPM) without relying on radar.

5. **Rider Behaviour Analytics:** A post-ride safety scoring mechanism evaluates lane discipline and reaction times to promote safer riding habits

**Methodology:**

The system follows a "Sense-Think-Act" robotic paradigm. The architecture is divided into two independent multithreaded modules to efficiently handle visual data processing. The Front-Camera Module (Forward Evaluator) resizes incoming frames to 640x640 for YOLOv8 inference, identifying vehicles and lane. Designed for the Jetson Nano, is executed and simulated on an RTX-powered environment to thoroughly test the limits of the detection logic using high-resolution video feeds.

1. **Front-Camera Perception (Forward Evaluator):** This module ingests forward-facing video feeds and utilizes the YOLOv8n (Nano) architecture to detect vehicles and obstacles in real-time. To calculate depth without relying on LiDAR, it applies an Inverse Proportional Height geometric heuristic. A Kalman Filter with a Constant Velocity Model (CVM) tracks objects across frames to calculate relative velocity and Time-To-Collision (TTC). If the TTC falls below a critical threshold, a Forward Collision Warning is triggered. Concurrently, an Inverse Perspective Mapping (IPM) transformation is applied to generate a Bird's-Eye View, utilizing color thresholding and polynomial fitting to monitor Lane Departure.
2. **Rear-Camera Monitoring (Rear Monitor):** This module tracks rear-approaching entities to mitigate the "blind spot paradox." It employs spatial logic gates to determine if an object's centroid falls within predefined rear blind spot coordinates. Furthermore, it calculates lateral velocity vectors to classify and alert the rider of high-speed overtaking maneuvers.
3. **Simulation and Edge Deployment:** To bypass hardware iteration bottlenecks and ensure safety, the entire software stack is initially validated using high-fidelity Software-in-the-Loop (SITL) simulation on an NVIDIA RTX platform. This environment processes real-world Indian traffic video feeds to tune detection thresholds and verify algorithmic correctness. Once validated, the models are optimized using NVIDIA TensorRT for low-latency deployment onto the physical NVIDIA Jetson Nano edge board.
4. **Safety Analytics & Driver Report Generation:** To promote safer riding habits, a dedicated background thread continuously logs critical telemetry, including the frequency of lane deviations, forward collision warnings, and blind-spot intrusions. Post-ride, a log parsing algorithm synthesizes this temporal data to generate a comprehensive "Driver Safety Evaluation Report." This virtual instructor mechanism evaluates the rider's risk exposure and compliance, assigning a cumulative safety grade (A-F) without interrupting the real-time processing of the core collision warning system.

SYSTEM REQUIREMENTS AND SPECIFICATIONS

| Category | Component | Specification |
|---|---|---|
| Sim. Workstation | Processor | AMD Ryzen 7 5800H |
| | GPU | NVIDIA RTX 3070 (8GB) |
| | RAM | 16GB |
| Edge | SBC | NVIDIA Jetson Nano (4GB) |
| Device Software Stack | Language | Python 3.8 |
| | Framework | PyTorch v1.10+, CUDA v11.x |
| | Vision Lib | OpenCV v4.5.5 |
| | Model | Ultralytics YOLOv8 |

**Result and Conclusion:**

In conclusion, the project successfully demonstrates a highly efficient, low-cost, vision-only ADAS tailored for two-wheelers. By executing a Hardware-in-the-Loop (HIL) simulation on an NVIDIA RTX platform, the logic for Forward Collision Warnings, Lane Departure, and Blind Spot Detection was rigorously validated against dynamic Indian traffic scenarios. The YOLOv8n architecture achieved a robust detection accuracy (mAP of 84%) and maintained high-speed inference, proving the algorithmic viability before edge deployment. Furthermore, the successful integration of the Safety Analytics Dashboard proved that the system can function as both an active threat-warning mechanism and a post-ride evaluation tool, automatically generating detailed driver performance reports and safety grades to encourage better riding discipline.

**References:**


[1] R. Nidamanuri, S. K. Nandyala, and R. S. Reddy, "A Progressive Review of Emerging Technologies for ADAS-Driven Solutions," *IEEE Access*, vol. 9, pp. 152311-152330, 2021.

[2] R. Chang, H. Wang, and Y. Liu, "Proactive Blind Spot Warning Systems for Powered Two-Wheelers Using Vision-Based Perception," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 5, pp. 4891-4902, May 2023.

[3] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," *Multimedia Tools and Applications*, vol. 82, pp. 9243-9275, 2023.

[4] G. Jocher, A. Chaurasia, and J. Qiu, "YOLOv8: A Unified, Anchor-Free Framework for Real-Time Object Detection," arXiv preprint arXiv:2408.15857, 2024.

[5] H. Chen, Y. Wang, and Y. Li, "Real-time Monocular Distance Estimation for Advanced Driver Assistance Systems," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 3, pp. 562-573, 2022

**Future Scope:**

The future scope of this project includes:

1. **Hardware Deployment and Field Testing:** Fabricating custom 3D-printed mounts to physically deploy the Jetson Nano and camera suite onto a motorcycle, transitioning from the Software-in-the-Loop (SITL) simulation to extensive physical field testing on real-world test tracks.
2. **TensorRT Optimization:** Converting the validated PyTorch YOLOv8 models into NVIDIA TensorRT engines to significantly reduce inference latency and achieve the targeted high-speed performance (<40ms per frame) natively on the Jetson Nano edge architecture.
3. **Sensor Fusion Integration:** Enhancing the system's robustness in low-light conditions and adverse weather by integrating low-cost Inertial Measurement Unit (IMU) data, adding a gyroscope for acceleration vectors and ultrasonic sensors, which will help eliminate false-positive "ghost detections" caused by shadows and poor visibility.