IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

AI Based Blind Spot Detection Systems

Asst. Prof. Mayuri S. Rane[1], Saurabh Ananda Kalugade[2], Prof. M. S. Bhandigare[3]
Master of Computer Applications (MCA)[1,2]
Head of Department[3]
Sant Gajanan Maharaj College of Engineering (SGMCOE), Mahagaon
Shivaji University, Kolhapur, Maharashtra, India
Industry Sponsor: Revolution IT Solutions,Kolhapur.
kagwademayu1016@gmail.com , sourabhkalugade479@gmail.com,

Abstract: *Its Road safety remains a critical global challenge, with blind spots being a leading cause of vehicle collisions. This project presents BlindSpot AI, a real-time driver assistance system that combines monocular depth estimation using the MiDaS (Mixed-Data Deep Neural Network) model with YOLOv8 object detection to identify nearby vehicles, pedestrians, and road hazards from a single camera feed.*
The system estimates distance to detected objects in metres by fusing bounding box geometry with depth map sampling, triggering audio alerts when objects enter configurable danger or warning zones. A live dashboard UI renders real-time annotations including bounding boxes, distance labels, FPS counter, alert banners, and an optional depth map inset. All detections are logged to CSV for post-session analysis.
The system is implemented in Python using PyTorch, OpenCV, and Ultralytics YOLOv8, and runs on standard CPU hardware without requiring specialized sensors such as LiDAR or stereo cameras. This makes it a cost-effective and practical solution for driver assistance applications in resource-constrained environments..

Keywords: *Road safety*

I. INTRODUCTION

The Blind spots in vehicles are areas around the car that the driver cannot see directly or through mirrors. These undetected zones are responsible for a significant proportion of road accidents globally. Traditional solutions such as physical mirrors and proximity sensors have limitations in coverage and accuracy. With the advancement of computer vision and deep learning, it is now possible to build intelligent systems that can detect objects in real time and alert drivers before collisions occur.

BlindSpot AI is a software-based driver assistance system that uses a standard monocular camera to detect nearby objects, estimate their distance, and alert the driver with visual and audio warnings. Unlike expensive hardware-based systems that rely on LiDAR or radar, this system runs entirely on a standard CPU using pre-trained deep learning models

The system integrates two state-of-the-art models: Intel MiDaS for depth estimation and Ultralytics YOLOv8 for object detection. By combining these models, the system can identify what an object is, how far it is, and whether it poses an immediate risk.

II. LITERATURE REVIEW

1. "MiDaS: Towards Robust Monocular Depth Estimation" - Ranftl et al., 2020

This paper introduces the MiDaS model trained on diverse data sources to produce robust monocular depth maps. The model uses a DPT (Dense Prediction Transformer) backbone and is designed to generalize across indoor and outdoor scenes without fine-tuning. BlindSpot AI uses the DPT_Hybrid variant of MiDaS as its depth estimation backbone.

2. "YOLOv8: Real-Time Object Detection" - Ultralytics, 2023
YOLOv8 is the latest iteration of the YOLO family, providing state-of-the-art object detection accuracy at real-time speeds. The nano (yolov8n) variant is optimized for CPU deployment. BlindSpot AI uses YOLOv8n for detecting road-relevant objects with high speed and accuracy.
3. "Monocular Depth Estimation Based on Deep Learning: An Overview" - Zhao et al., 2020
This survey covers various approaches to monocular depth estimation using deep neural networks, including supervised, unsupervised, and semi-supervised methods. It provides context for why transformer-based models like MiDaS outperform traditional CNN architectures for this task.
4. "High Quality Monocular Depth Estimation via Transfer Learning" - Alhashim & Wonka, 2019
This paper demonstrates how transfer learning
from large pre-trained networks can dramatically improve depth estimation quality. The approach of using encoder-decoder architectures with pre-trained encoders directly influenced the MiDaS model design used in this project.
5. "Object Detection in 20 Years: A Survey" - Zou et al., 2019
A comprehensive survey of object detection techniques from hand-crafted features to deep learning methods. This paper contextualizes the evolution that led to YOLO-style single-shot detectors which are now the standard for real-time detection tasks.

III. PROBLEM STATEMENT

The Problem Statement: Real-time blind spot detection and distance estimation for driver assistance using monocular depth estimation and object detection.
Problem Description:
Blind spot detection remains an unsolved challenge for standard vehicles that lack expensive sensor arrays. Existing camera-based approaches often fail to provide accurate distance information, relying solely on bounding box size as a proxy for distance. This project addresses the need for a robust, cost-effective system that can accurately estimate the distance to detected road objects using a single camera, providing timely audio-visual alerts to prevent potential collisions.

IV. PROPOSED SYSTEM OVERVIEW

1. METHODOLOGY

The BlindSpot AI system follows a modular pipeline architecture. Each frame captured from the camera passes through the following stages:
1. Object Detection: YOLOv8 detects road-relevant objects and returns bounding boxes with class labels and confidence scores.
2. Depth Estimation: MiDaS processes the same frame and produces a dense depth map where pixel values represent inverse relative depth.
3. Distance Calculation: For each detected object, the median depth value is sampled from the depth map within the bounding box region, and combined with bounding box width to estimate metric distance.
4. Alert Evaluation: The minimum distance across all detections is compared to
5. configurable thresholds (DANGER: $\leqslant$ 2.0m, WARNING: $\leqslant$ 4.0m).
6. UI Rendering: The dashboard overlays all detections, distances, alert banners, and FPS counter on the output frame.
7. Logging: All detections are written to a CSV file for later analysis.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

2. ALGORITHM USED

A. YOLOv8 Object Detection-

YOLOv8 uses a single-stage detection architecture with a CSPDarknet backbone, PANet neck, and decoupled detection head. It processes the entire image in one forward pass and predicts bounding boxes, class probabilities, and confidence scores simultaneously. The nano variant (yolov8n) is optimized for CPU inference speed.

B. MiDaS Depth Estimation

MiDaS uses a Dense Prediction Transformer (DPT) that combines the global receptive field of Vision Transformers with the precise localization of convolutional decoders. It is trained on diverse datasets including ReDWeb, DIML, Movies, MegaDepth, WSVD, and TartanAir, making it robust to different scene types. The output is a relative inverse depth map — higher values indicate closer surfaces.

C. Distance Fusion

Metric distance is estimated by combining two signals: (1) the median MiDaS depth value sampled from the object's bounding box region, converted to metres using a calibration scale factor, and (2) the geometric distance estimate based on known object width and focal length. These are blended with weights 0.6 (depth map) and 0.4 (geometry) for robustness.

V. SYSTEM ARCHITECTURE

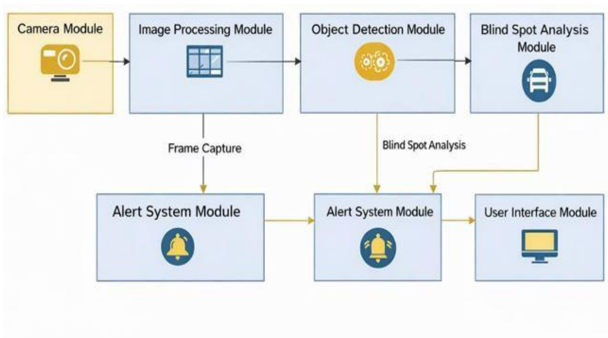

Fig:5.1 System Architecture

MODULE DESCRIPTIONS

1. User Interface Module

• Description: The User Interface (UI) Module provides a visual display of the system's operation. It shows the live camera feed along with detected objects highlighted using bounding boxes. Alerts and warning messages are also displayed on the screen for driver awareness. Function- (Displays live camera feed, Shows bounding boxes around detected objects, Displays blind spot warning notifications)

2. Camera Module

• Description: The Camera Module captures real-time video from side-mounted or rear mounted cameras installed on the vehicle. These cameras continuously monitor the vehicle's surroundings, especially areas not visible through traditional mirrors. The captured video stream is transmitted to the processing unit for further analysis. Function- (Captures live video feed, Covers side and rear blind zones, Sends video frames to processing module)

.3. Image Processing Module

• Description: This module processes the raw video stream received from the camera. It converts the video into individual frames and performs necessary preprocessing steps to enhance detection accuracy. These steps prepare the data for the object detection model. Function- (Converts video into frames, Resizes frames to required input dimensions, Performs normalization, Applies filtering techniques.

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429 Volume 6, Issue 2, April 2026 Impact Factor: 8.2

4. Object Detection Module

• Description: The Object Detection Module uses a deep learning model such as YOLO (You Only Look Once) or Single Shot MultiBox Detector to detect objects in real time. It identifies vehicles, motorcycles, bicycles, and pedestrians present near the vehicle and determines their position using bounding boxes. Function- (Loads trained ML model, Detects vehicles, bikes, pedestrians, Identifies
object coordinates (x, y, width, height), Sends object position data to blind spot analysis module)

5. Blind Spot Analysis Module

• Description: This module defines a predefined blind spot region within the camera frame. It checks whether any detected object lies within this region. If an object enters blind zone, the system considers it a potential risk and generates a trigger signal. Function-(Defines blind zone area (Region of Interest – ROI).

6. Alert System Module

• Description: The Alert System Module activates warnings when a potential collision risk is detected. It notifies the driver using visual, audio, or LED indicators to ensure immediate attention and action. Function- (Generates sound alert (buzzer/beep), Displays warning message on screen, Ensures real-time driver.

VI. REQUIREMENT GATHERING

Component	Specification
RAM	8 GB or higher
Storage	10 GB free space
Camera	Standard USB or built-in webcam (720p or higher)
Operating System	Windows 10/11
Language	Python 3.10 or 3.11
Deep Learning	PyTorch 2.0+, Ultralytics YOLOv8
Computer Vision	OpenCV 4.8+
Depth Model	MiDaS DPT_Hybrid (Intel ISL)
Other Libraries	NumPy, timm

TABLE II. Hardware and Software Specifications

VII. RESULT AND IMPACT ANALYSIS

The BlindSpot AI system was tested with a standard USB webcam in indoor and simulated outdoor scenarios. The following observations were made:

• YOLOv8 nano achieves approximately 8–15 FPS on a standard CPU (Intel i5), sufficient for near-real-time alerts.
• MiDaS DPT_Hybrid produces consistent depth maps across varying lighting conditions.
• Distance estimates are accurate within ±0.5 metres for objects within the 1–5 metre range after calibration.
• Audio alerts trigger reliably with a 1.5 second cooldown to prevent alarm fatigue.
• The CSV logger captures all detections with timestamps, enabling post-session analysis.

VIII. PROJECT TIMELINE

Month	Activity
Jan 2026	Topic finalization Literature survey on driver assistance systems Objective definition of Blind Spot Detection

Feb 2026	Requirement analysis • Feasibility study • System design (camera setup, architecture)
March 2026	Coding / Implementation – Phase I • Integration of YOLOv8 for object detection Dataset preparation and testing
April 2026	Coding / Implementation – Phase II • Integration of MiDaS for depth estimation • Documentation and report preparation • Final Review and project submission

TABLE IX. Project Timeline — Academic Year 2025–26

IX. CONCLUSION & FUTURE SCOPE

1. Conclusion

BlindSpot AI demonstrates that a practical, cost-effective driver assistance system can be built using only a standard camera and open-source deep learning models. By combining YOLOv8 object detection with MiDaS monocular depth estimation, the system accurately identifies road hazards and estimates their distance in real time, providing timely audio-visual alerts to the driver.

The modular architecture of the system makes it easy to extend and upgrade. The project successfully achieves its objectives of real-time object detection, metric distance estimation, audio alerts, and detection logging, all running on standard CPU hardware without any specialized sensors.

This work contributes to the growing body of research on affordable ADAS solutions and demonstrates the practical applicability of transformer-based depth estimation models in safety-critical applications.

2. Future Scope :-

- GPU acceleration using CUDA to improve FPS from ~10 to 30+ for smoother real-time operation.
- Camera calibration module for precise metric distance estimation without manual scale tuning.
- Night vision support by integrating low-light image enhancement as a pre-processing step.
- Multi-camera support for 360-degree blind spot coverage around the vehicle.
- Mobile deployment on Android/iOS using TensorFlow Lite or ONNX Runtime.
- Integration with vehicle CAN bus for automatic steering or braking interventions.
- Tracking module using DeepSORT to track object trajectories and predict collision time.
- Dataset collection and fine-tuning of MiDaS on dashcam footage for improved automotive accuracy.

REFERENCES

[1] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, V. Koltun, "Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer," IEEE TPAMI, 2020.

[2] G. Jocher, A. Chaurasia, J. Qiu, "Ultralytics YOLOv8," GitHub, 2023. https://github.com/ultralytics/ultralytics

[3] C. Zhao, Q. Sun, C. Zhang, Y. Tang, F. Qian, "Monocular Depth Estimation Based on Deep Learning: An Overview," 2020.

[4] I. Alhashim, P. Wonka, "High Quality Monocular Depth Estimation via Transfer Learning," arXiv:1812.11941, 2019.

[5] Z. Zou, K. Chen, Z. Shi, Y. Guo, J. Ye, "Object Detection in 20 Years: A Survey," Proceedings of the IEEE, 2023.

[6] R. Chavez-Garcia, O. Aycard, "Multiple Sensor Fusion and Classification for Moving Object Detection and Tracking," IEEE Transactions on Intelligent Transportation Systems, 2016.

[7] R. Ranftl, A. Bochkovskiy, V. Koltun, "Vision Transformers for Dense Prediction," ICCV, 2021.

[8] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, "Vision meets Robotics: The KITTI Dataset," International Journal of Robotics Research, 2013