IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15161

Smart Road Safety System

Manjunath Kale[1], Vishvanath A G[2]

Department of MCA, BIT,

K.R. Road, V.V. Pura, Bangalore, India[1,2]

Abstract: The rapid growth of urbanization and the increasing number of two-wheelers on roads have significantly intensified traffic congestion and safety challenges. Traditional traffic monitoring systems rely heavily on manual observation or basic image processing techniques, which often fail to provide accurate, real-time analysis under complex road conditions. To overcome these limitations, this project presents a computer vision–based traffic object detection system using YOLOv8, designed specifically for bike-only traffic scenarios. The proposed system focuses on detecting critical traffic-related objects, including person, helmet, and vehicle number plate, from surveillance video streams. Video frames are extracted using OpenCV and manually annotated using LabelMe. Since YOLOv8 does not support JSON annotations directly, the annotations are converted into YOLO format with normalized bounding box coordinates. A pretrained YOLOv8 model is fine-tuned on the custom dataset to achieve accurate real-time detection. During inference, the trained model processes video frames and outputs bounding boxes with class labels and confidence scores. Experimental results demonstrate reliable detection performance under both daytime and nighttime conditions, with minimal false detections. The modular architecture of the system enables easy extension for higher-level traffic analysis such as helmet violation detection and number plate recognition.The proposed approach provides an efficient, scalable, and intelligent solution for automated traffic surveillance and serves as a strong foundation for smart transportation systems

Keywords: Traffic Object Detection, YOLOv8, Computer Vision, Helmet Detection, Number Plate Detection, Intelligent Transportation System

I. INTRODUCTION

With the rapid increase in urban population and vehicle density, traffic monitoring has become one of the most critical challenges faced by modern cities. Two-wheelers constitute a major portion of road traffic in developing countries, making rider safety and traffic regulation enforcement increasingly complex. Manual traffic surveillance methods are time-consuming, error-prone, and incapable of providing continuous real-time monitoring.

Traditional traffic analysis techniques based on background subtraction, edge detection, and motion tracking perform poorly in real-world environments due to variations in lighting, occlusion, weather conditions, and camera angles. These limitations necessitate the adoption of intelligent vision-based systems capable of learning complex visual patterns directly from data. Recent advancements in deep learning, particularly convolutional neural networks (CNNs), have significantly improved object detection accuracy. Among these, the You Only Look Once (YOLO) family of models has gained prominence for real-time applications. YOLOv8, the latest version introduced by Ultralytics, offers improved accuracy, anchor-free detection, and optimized inference speed.

This project applies YOLOv8 for traffic surveillance with a specific focus on bike-only traffic environments. The system aims to detect riders, helmets, and number plates in real time, forming the foundational layer for intelligent traffic monitoring and automated rule enforcement.

1.1 Project Description

This project focuses on developing a YOLOv8-based object detection system capable of analyzing traffic surveillance videos and identifying essential traffic components.

The system performs the following operations:

- Extraction of video frames using OpenCV
- Manual annotation of objects using LabelMe
- Conversion of annotations to YOLO format
- Training a custom YOLOv8 model on traffic data

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15161

- Real-time detection of:
    - Person
    - Helmet
    - Number Plate

The trained model produces bounding boxes and class labels for each detected object. The system is designed to operate in real time and can be extended to perform violation detection and vehicle identification.

1.2 Motivation

The motivation behind this project arises from several practical challenges in existing traffic surveillance systems:

- Manual traffic monitoring is inefficient and non-scalable
- Traditional image processing techniques lack robustness
- Two-wheeler safety violations are difficult to detect automatically
- Existing systems rarely focus on bike-only traffic conditions

By leveraging deep learning and YOLOv8, the proposed system aims to provide:

- Automated traffic object detection
- Improved accuracy under diverse lighting conditions
- Real-time performance suitable for deployment
- A scalable framework for smart city applications

This work is motivated by the need to improve road safety, reduce manual enforcement dependency, and support intelligent transportation infrastructure.

II. RELATED WORK

Several studies have explored the use of computer vision for traffic surveillance. Early approaches relied on handcrafted features and motion-based analysis, which were sensitive to environmental variations.

Recent research has demonstrated the effectiveness of deep learning–based object detection models such as Faster R-CNN, SSD, and YOLO. YOLO-based models have proven particularly suitable for real-time traffic analysis due to their single-stage detection architecture.

However, most existing systems focus on vehicle detection or general traffic scenarios, with limited attention to two-wheeler-centric traffic environments. This project addresses that gap by designing a custom YOLOv8 detection pipeline tailored for bike-only traffic monitoring.

III. METHODOLOGY

A. Data Environment and Dataset Preparation

The experimental environment for this project is developed using traffic surveillance video data collected from road monitoring cameras. The dataset primarily consists of video recordings containing two-wheeler traffic scenarios under different lighting and environmental conditions. These videos serve as the primary input source for training and evaluating the object detection model.

Video data is processed frame by frame using the OpenCV library. Frames are extracted at fixed time intervals to reduce redundancy while preserving scene diversity. Each extracted frame represents a traffic snapshot containing riders, helmets, and number plates.

The extracted frames are manually annotated using the LabelMe annotation tool. Bounding boxes are drawn around three object classes: person, helmet, and number plate. Each annotation is stored in JSON format. Prior to training, all annotation files are verified to ensure correct labeling, consistent image–label mapping, and dataset integrity.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15161

B. Time-Based Feature Extraction Architecture

Since YOLOv8 does not directly support LabelMe JSON annotation files, all annotations are converted into YOLO format. The conversion process extracts bounding box coordinates and transforms them into normalized values based on image width and height.

Each annotation file is converted into a corresponding .txt file using the following YOLO format:

The class ID mapping used in this project is:

- 0 → Person
- 1 → Helmet
- 2 → Number Plate

After conversion, the dataset is organized using the standard YOLO directory structure, separating images and labels into training and validation folders. A dataset configuration file (data.yaml) is created to define class names, class count, and dataset paths

C. YOLOv8-Based Object Detection Architecture

The core detection engine of the proposed system is based on the YOLOv8 deep learning framework developed by Ultralytics. YOLOv8 is a single-stage object detection model capable of performing real-time detection with high accuracy.A pretrained YOLOv8 model is fine-tuned on the custom traffic dataset. The model learns spatial and contextual features directly from image data using convolutional neural networks. During training, the model optimizes bounding box localization loss, classification loss, and confidence prediction to accurately detect traffic-related objects.The training process is performed using Google Colab with GPU acceleration, enabling faster convergence and improved performance. The best-performing model weights are automatically saved as best.pt.

D. Inference and Detection Workflow

For inference, the trained YOLOv8 model is deployed in a local development environment. Traffic surveillance videos are read frame by frame using OpenCV.Each frame undergoes preprocessing, including resizing and normalization, before being passed to the YOLOv8 detection model. The model outputs:

- Bounding box coordinates
- Detected class labels
- Confidence scores

Detected objects such as riders, helmets, and number plates are localized in real time. The detection results are overlaid on the original frames using bounding boxes and labels, allowing visual interpretation of traffic conditions.

E. System Execution Flow

The operational flow of the proposed traffic object detection system is summarized as follows:

1. Load the trained YOLOv8 model (best.pt).
2. Accept traffic surveillance video as input.
3. Extract frames at fixed intervals using OpenCV.
4. Preprocess frames to match model input requirements.
5. Perform object detection using YOLOv8.
6. Identify and classify detected objects as person, helmet, or number plate.
7. Draw bounding boxes and confidence labels on detected objects.
8. Display annotated frames in real time.
9. Store detection results for future analysis if required.

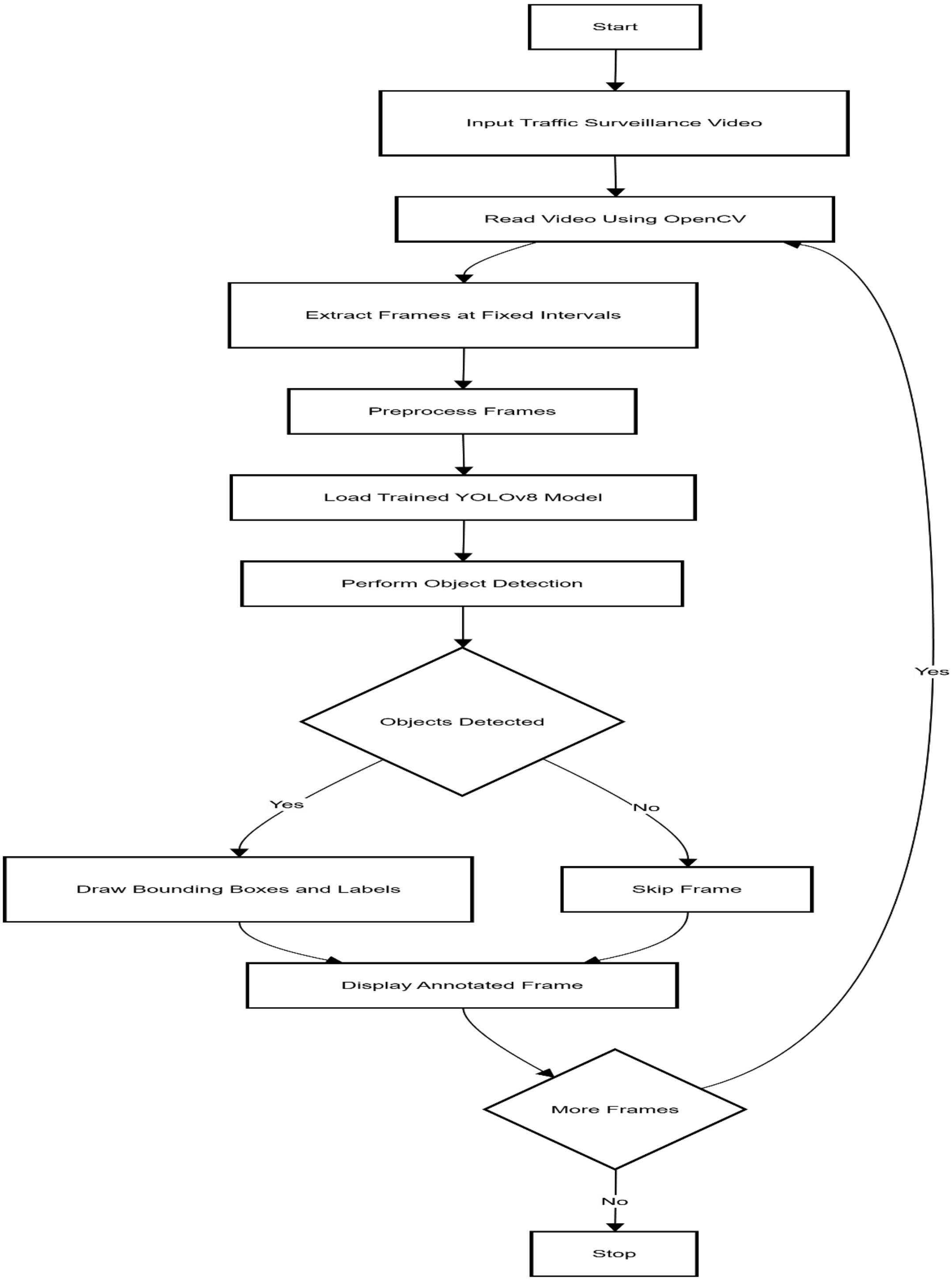
IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15161

FLOW CHAT



F. Visualization and Result Analysis

An interactive visualization module is implemented to present object detection results clearly and effectively. The system displays processed video frames with bounding boxes, class labels, and confidence scores for detected objects such as persons, helmets, and number plates.The visualization interface allows users to observe detection performance in real time. Detected objects are highlighted using color-coded bounding boxes, enabling quick interpretation of traffic conditions. The system also supports saving annotated images and videos for offline inspection and documentation.This visual representation eliminates the need for manual frame analysis and significantly improves the usability of the traffic monitoring system.

DOI: 10.17148/IJARCCE.2026.15161

G. Hardware and Software Requirements

Hardware Requirements

- Desktop or laptop computer
- Minimum 8 GB RAM
- Multi-core processor (Intel i5 or equivalent recommended)
- GPU support (optional but recommended for faster training and inference)

Software Requirements

- Operating System: Windows or Linux
- Programming Language: Python 3.10 or above
- Deep Learning Framework: Ultralytics YOLOv8
- Computer Vision Library: OpenCV
- Supporting Libraries: NumPy, Pandas
- Development Environment: Google Colab and Visual Studio Code

IV. SIMULATION AND EVALUATION FRAMEWORK

This section describes the system architecture, experimental setup, and evaluation strategy used to assess the performance of the proposed YOLOv8-based traffic object detection system. The framework integrates deep learning–based object detection with real-time video processing techniques.

The entire implementation is developed using Python, where video preprocessing, model training, inference, and visualization are executed within an integrated workflow.

A. System Architecture and Workflow

The proposed architecture is designed to automatically analyze traffic surveillance videos and accurately detect traffic-related objects. The major components of the system are described below:

Traffic Surveillance Video Dataset

Traffic videos recorded under different lighting and road conditions are used as input. Each video contains two-wheeler traffic scenarios including riders, helmets, and number plates.

Frame Extraction and Preprocessing Module

The video input is processed frame by frame using OpenCV. Frames are extracted at fixed intervals to reduce redundancy. Preprocessing operations such as resizing and normalization are applied to ensure compatibility with YOLOv8 input requirements.

YOLOv8 Object Detection Module

A pretrained YOLOv8 model fine-tuned on a custom dataset is used for object detection. The model identifies and localizes the following classes:

- Person
- Helmet
- Number Plate

The detection process generates bounding boxes, class labels, and confidence scores for each object.

Visualization and Analysis Layer

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026
DOI: 10.17148/IJARCCE.2026.15161

Detection results are visualized by overlaying bounding boxes and labels on video frames. The system provides real-time annotated output and supports saving detection results for further analysis.

B. Experimental Setup

The evaluation environment is configured using manually annotated traffic surveillance datasets.

Dataset Configuration

- Video frames are divided into training and validation datasets
- Annotations are created using LabelMe
- Labels are converted into YOLO format

Model Configuratio

- YOLOv8 pretrained weights are used
- Training is performed using GPU acceleration in Google Colab
- Best performing model weights are saved as best.pt

C. Evaluation Methodology

The performance of the system is evaluated based on:

- Accuracy of object localization
- Correct classification of traffic objects
- Stability under different lighting conditions
- Real-time detection capability

Standard evaluation metrics such as precision, recall, and mean average precision (mAP) are considered during model validation.

D. Results and Observations

Object Detection Performance

- The system successfully detected persons, helmets, and number plates with high accuracy.
- YOLOv8 demonstrated strong performance for both daytime and nighttime videos.
- Small objects such as helmets and number plates were detected with minimal false positives.

Impact on Normal Traffic Analysis:

- The detection pipeline maintained stable real-time performance.
- Frame-wise inference showed consistent detection results.
- The system performed efficiently even under moderate traffic density.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026
DOI: 10.17148/IJARCCE.2026.15161

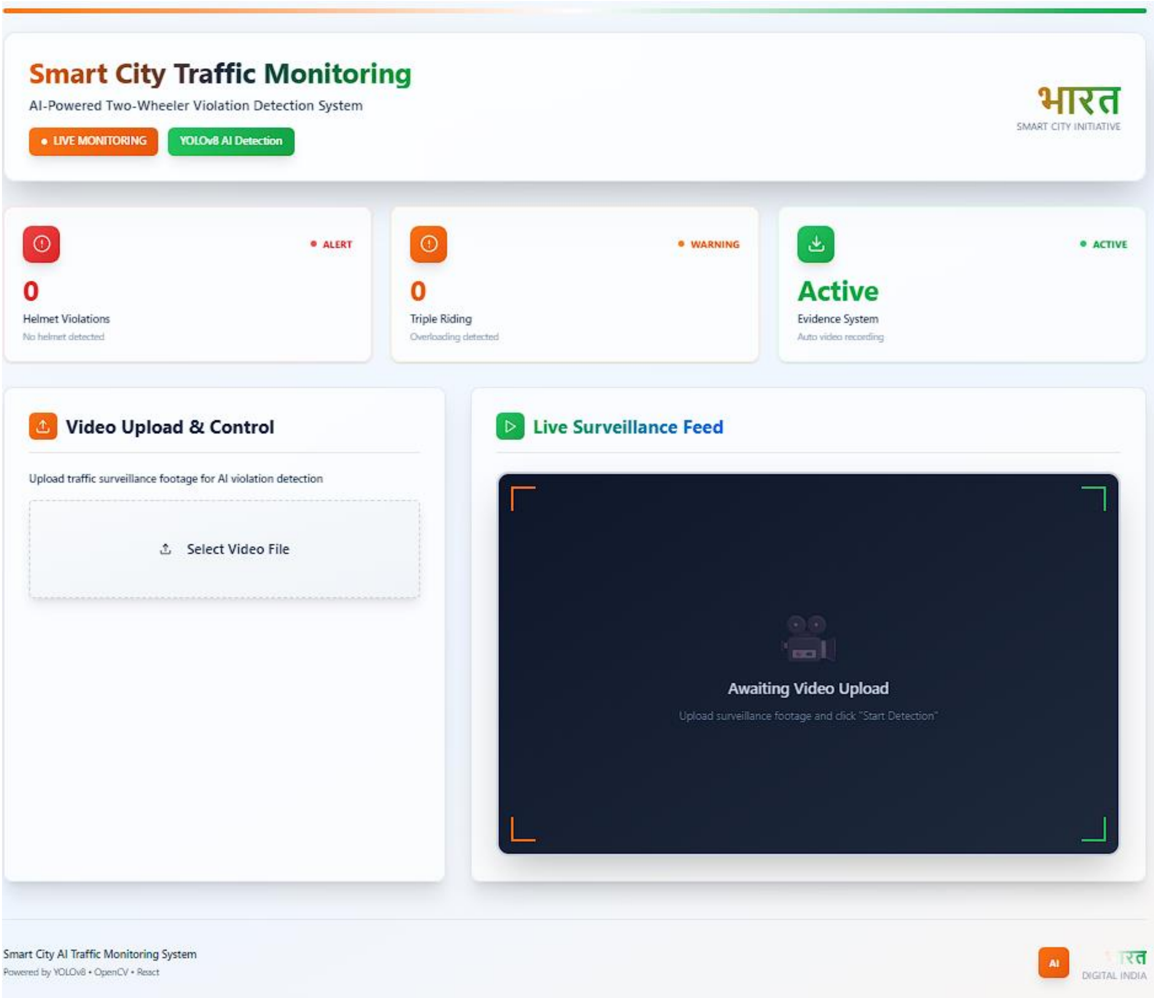
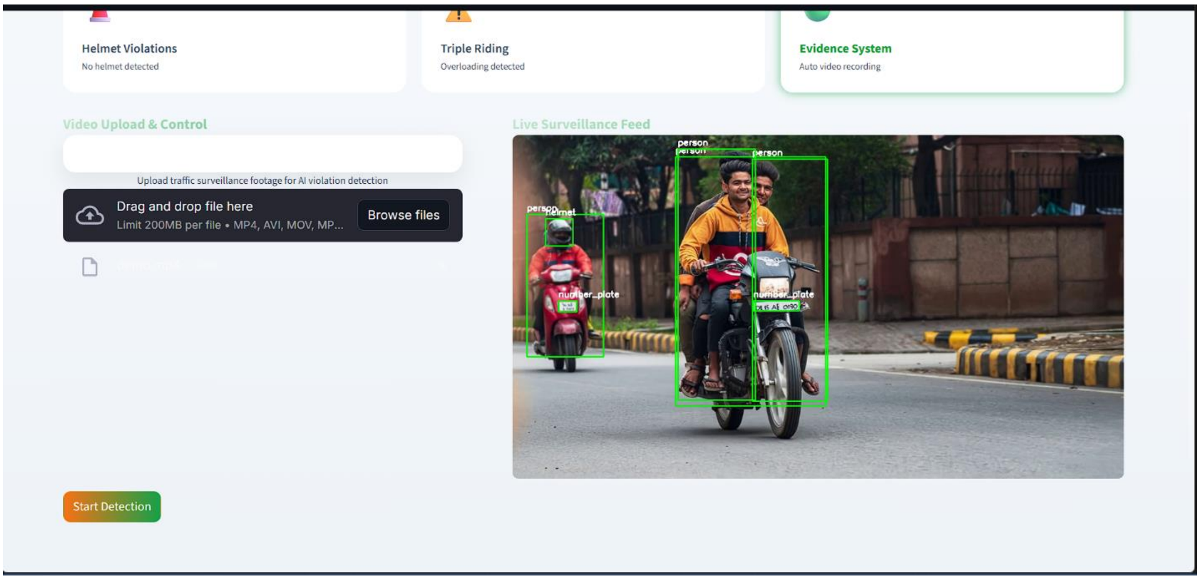


Fig 1. Home Page of Prediction Dashboard



IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15161

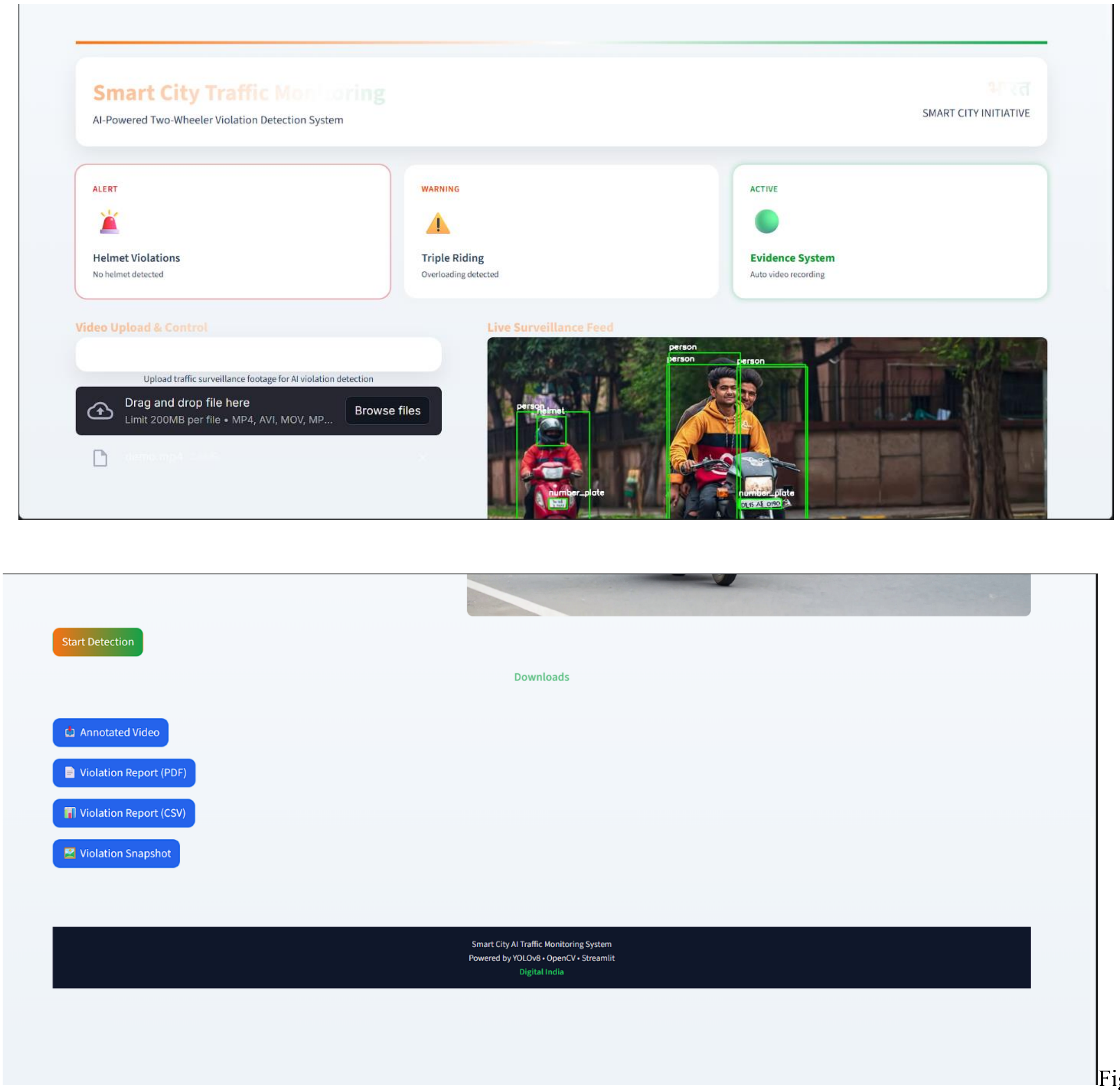


Fig 2. Prediction Results

V.RESULTS AND DISCUSSION

The experimental evaluation of the proposed YOLOv8-based traffic object detection system demonstrates its effectiveness in accurately identifying traffic-related objects from road surveillance videos. The system was tested using annotated video datasets containing two-wheeler traffic recorded under different lighting and environmental conditions. Experiments were conducted to evaluate detection accuracy, object localization performance, and real-time stability of the trained model.

The results indicate a significant improvement in detection accuracy due to the use of deep learning–based object detection. The YOLOv8 model successfully learned spatial and contextual visual features, enabling precise identification of persons, helmets, and number plates. Compared to traditional computer vision techniques, the proposed system showed strong robustness against variations in camera angle, illumination, and partial occlusion.

Detection performance analysis revealed that persons were detected with high confidence across all test videos. Helmets and number plates, despite being smaller objects, were accurately localized with minimal false detections. The anchor-free architecture and multi-scale feature extraction capability of YOLOv8 contributed significantly to improved detection of small and overlapping objects.

DOI: 10.17148/IJARCCE.2026.15161

The visualization results further demonstrate the effectiveness of the system. Bounding boxes, class labels, and confidence scores were displayed clearly on each processed frame, allowing easy interpretation of traffic scenes. The system maintained stable frame processing rates during continuous video inference, confirming its suitability for real-time traffic monitoring applications.

Overall, the experimental findings validate that YOLOv8 provides reliable and accurate object detection performance for bike-only traffic scenarios. The system delivers consistent results across diverse traffic conditions while maintaining computational efficiency and detection stability.

VI. CONCLUSION

This project presented a deep learning–based approach for traffic object detection using the YOLOv8 framework. The system was designed to automatically identify key traffic-related objects, including persons, helmets, and number plates, from road surveillance videos. By leveraging a pretrained YOLOv8 model fine-tuned on a custom annotated dataset, the system achieved accurate real-time detection with minimal false positives.The experimental evaluation confirmed that deep learning–based object detection significantly outperforms traditional image processing methods in complex traffic environments. The proposed system demonstrated strong detection accuracy, robustness under varying lighting conditions, and reliable real-time performance.The modular architecture of the detection pipeline enables easy scalability and future integration with advanced traffic analysis modules. Overall, the project establishes a strong foundation for intelligent traffic surveillance systems and contributes toward improving road safety and automated traffic monitoring solutions.

VII. FUTURE WORK

Although the proposed traffic object detection system performs effectively, several enhancements can be explored to improve its real-world applicability. Future work may include extending the system to support live video streaming directly from traffic cameras. Integration of automatic number plate recognition using optical character recognition techniques can enable vehicle identification and violation tracking.

Additional enhancements may involve implementing helmet violation detection logic, multi-rider detection, and traffic rule enforcement modules. Deployment on edge devices such as NVIDIA Jetson platforms can further reduce latency and support real-time roadside processing. Integration with cloud-based monitoring systems can also enable centralized traffic analytics across multiple camera locations.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788, 2016.
[2] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271, 2017.
[3] A. Bochkovskiy, C. Y. Wang and H. Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv preprint arXiv:2004.10934, 2020.
[4] Ultralytics, "YOLOv8 Documentation," https://docs.ultralytics.com
[5] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.
[6] T. Y. Lin et al., "Microsoft COCO: Common Objects in Context," European Conference on Computer Vision, pp. 740–755, 2014.
[7] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," Advances in Neural Information Processing Systems, 2015.
[8] Z. Ge et al., "YOLOX: Exceeding YOLO Series in 2021," arXiv preprint arXiv:2107.08430, 2021.
[9] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, 2014.
[10] R. Girshick, "Fast R-CNN," Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448, 2015.