iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79903

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 IV Apr 2026- Available at www.ijraset.com

An Intelligent Traffic Detection Framework Using Deep Learning

Gouraveni Ramya[1], Koti Reddy[2], Juluri Ruchitha[3], Kanchumarthi Sai Ram Maruthi Nandan[4], JadalaRuthik[5], Durgapagari Bhavana[6]

[1, 3, 4, 5, 6]*Student, Department of CSE Hyderabad Institute of Technology and Management, Telangana, India*
[2]*Assistant Professor , Department of CSE, Hyderabad Institute of Technology and Management, Telangana, India*

Abstract: ***This paper introduces a deep learning–based traffic detection framework designed to monitor traffic scenes in real time and support adaptive responses to changing road conditions. The proposed system follows a multi-stage pipeline consisting of high-resolution video and image inputs captured from roadside cameras, preprocessing steps for noise removal and illumination normalization, and object detection using convolutional neural network (CNN) models such as AlexNet or Faster R-CNN. Detected vehicles and pedestrians are subsequently tracked across consecutive frames using effective tracking algorithms, enabling behaviour analysis including lane changes, stopping patterns, and unsafe manoeuvres. To improve performance across varying environmental conditions, the framework employs data augmentation and domain adaptation strategies. A decision module maps detection and tracking outputs to actionable alerts (e.g., congestion, accidents) or control commands (signal timing adjustments). Experiments conducted on standard datasets and real traffic video samples indicate strong performance, achieving precision and recall values above 90%, along with low false-alarm rates and latency suitable for real-time operation. The proposed framework supports efficient traffic management, enhances road safety, and contributes toward the development of smarter and more adaptive urban transportation infrastructure.***

Keywords: Deep Learning, YOLO, Computer Vision, Traffic Monitoring, Intelligent Transportation Systems, Vehicle Detection, Artificial Intelligence.

I. INTRODUCTION

Urban transportation in Indian cities is characterized by highly diverse traffic behavior, frequent rule violations, and uneven utilization of road infrastructure. In contrast to many Western cities where lane discipline and road markings are strictly enforced, Indian roads typically experience mixed traffic involving cars, two-wheelers, auto rickshaws, pedestrians, and animals , often lacking strict lane separation. Such heterogeneous traffic scenarios require perception systems capable of handling unstructured layouts, inconsistent illumination, frequent occlusions, and irregular events.



Fig : 01 Traffic congestion

To address these challenges, this work presents a deep learning–based traffic detection framework tailored to the characteristics of Indian road scenarios. The framework starts with video acquisition from roadside and overhead camera installations. Prepossessing modules adjust for nonuniform illumination (shadows, glare), weather perturbations (rain, haze), and image noise. Robust object detection models, such as YOLO variants or Faster R- CNN trained on Indian traffic datasets, are employed to identify vehicles, pedestrians, two- wheelers, auto rickshaws, and other road users. A tracking component links detection's frame to frame, allowing estimation of trajectories, speeds, and density. Further, interaction modules analyze lane-change, overtaking, and collision-prone behavior (for example using methods like Siamese interaction LSTMs for intersections in India)

©IJRASET: All Rights are Reserved



Fig :02 Traffic detetction

Finally, a decision engine triggers adaptive responses — dynamic signal timing, congestion alerts, or collision warnings — in real time. Experimental evaluation on Indian traffic scenarios demonstrates that the proposed framework achieves strong detection accuracy while preserving low processing latency even under dense, unstructured traffic. Overall, the proposed framework aims to support safer and more efficient urban traffic management in Indian cities.

II. LITERATURE REVIEW

1) Indian traffic sign detection and recognition using deep learning
Authors: Rajesh Kannan Megalingam, Konda Reddy thanugundala ,sreevatsava Reddy.
Dataset: Several studies developed medium- scale annotated Indian traffic sign datasets to improve detection performance; however, most of these works remain limited to sign-level analysis rather than full traffic scene understanding.
Out of these, 4544 images are Indian, including 1936 from police, with varied resolutions and visibility levels. Some images have multiple signs (up to 4), and 29.8% are low resolution, with 3% of signs barely visible.

2) Real-time Traffic Surveillance and Detection using Deep Learning and Computer Vision Techniques .
Author: Riju Tiwari , Abdul Hadi Rumney , M Saravanan.
•A customized data set of 6480 images and 7056 instances of Indian traffic signs was created.
•Data augmentation strategies were incorporated to improve robustness and generalization performance of the models.
•The dataset included 87 diverse traffic sign categories for robust model training

3) Real-Time Traffic Signal Optimization using Deep Learning and Reinforcement Learning
Authors : Pathipaka Sai Krishna, Gudem Varun.
•A curated data set of 6480 images and 7056 instances of Indian traffic signs across 87 categories.
•A customized annotated data set with bounding boxes and class labels for YOLOv5 training and evaluation.
•A real-time traffic dataset used in YOLOv8- based adaptive traffic control simulations and tests

4) Deep Learning Approaches for Vehicle Detection and Classification: A Survey (Saratham be-kai et al., IJARESM, 2025) The survey examines approximately 15 studies published since 2020, comparing datasets, model architectures, and evaluation metrics. Strength: gives a good overview of current trends; Limitation: less detail on real-world deployment (occlusions, lighting, cost of computation).

5) Traffic Detection and Recognition Using YOLO Object Detection Algorithm: A Systematic Review (Flores-Calero et al., 2024)Analyses YOLO variants (2016-2022) for traffic sign detection & recognition. Covers datasets, hardware, applications. Strength: systematic, broad; Limitation: Limitation: the scope is largely restricted to traffic sign analysis rather than comprehensive traffic scene understanding.

6) Indian Traffic Sign Detection and Recognition Using Deep Learning (Megalingam et al.)Proposes a Refined Mask R-CNN on Indian sign dataset (6480 images, 87 classes). The study reports precision close to 97%, highlighting the effectiveness of using datasets tailored to Indian road conditions; Limitation: primarily addresses sign detection without modeling broader traffic participants or interactions.

7) Automatic Vehicle Detection and Tracking Strategy Using Deep Learning Model (YOLO v2 & R-CNN) (Ankireddy et al., IJISAE)Combines YOLOv2, R-CNN & Fast-RCNN, does both detection & tracking. Strength: tracking + detection; Limitation: likely heavy computational cost, details of occlusion etc. may not be deeply addressed.

8) Deep Learning-Driven Pattern Recognition for Real-Time Traffic Incident Detection in Complex Urban Environments (IIETA)Proposes FCOS detector with Rep- VGG backbone, handles incident detection (12 classes) from HWID12 dataset; ~96.9% accuracy. Strength: multi-incident detection, real time, good accuracy; Limitation: dependency on large annotated dataset; possible latency under constrained hardware

9) Performance Analysis of YOLO-based Architectures for Vehicle Detection from Traffic Images in Bangladesh (Alamgir et al., 2022) Compares YOLOv3, YOLOv5s, YOLOv5x on datasets (Bangladesh + self-collected). Finds YOLOv5x best. Strength: local dataset, comparative; Limitation: perhaps less evaluation under extreme weather/occlusion.

10) Research on Multi-Target Detection and Tracking of Intelligent Vehicles in Complex Traffic Environments Based on Deep Learning Theory (World Electr. Veh. J., 2025)Improves with attention mechanisms, new suppression methods (Bot-NMS) in YOLOv7, for small/occluded targets. Strength: addresses occlusion, small object detection; Limitation: maybe not evaluated extensively on real-time or low resource settings.

11) Vehicle Detection in High Density Traffic Surveillance Data using YOLOv5 (Sneha & Kumar, 2024) Uses KITTI + Indian dataset DATS_2022. Reports ~93.5% precision, 90.7% recall on KITTI; lower on Indian dataset. Strength: addresses high density traffic; Limitation: performance drop on non-standard datasets, generalization issues.

12) Smart Traffic Monitoring with YOLOv9 Object Detection Algorithm (Sekar & Kartal, 2025)Applies YOLOv9 to classify and count types of vehicles. Cars/trucks detection >95% accuracy. Strength: recent algorithm, good results; Limitation: smaller objects (motorcycles etc.) accuracy drop; likely hardware dependency.

13) Highly Accurate Deep Learning Models for Estimating Traffic Characteristics from Video Data (Cai et al., 2024)Extracts traffic characteristics (speed, volume etc.) from video using DL (backbones + object detection + tracking). Strength: infers metrics beyond detection; Limitation: challenges in tracking accuracy, calibration, frame rate trade-offs.

14) Applications of Deep Learning in Traffic Congestion Detection, Prediction and Alleviation: A Survey (Kumar & Raubal, 2021)Surveys DL methods for congestion detection/prediction. Classification of recurring vs non-recurring congestion. Strength: identifies gaps; Limitation: older methods may have less relevance now with newer architectures.

15) Graph Neural Network for Traffic Forecasting: A Survey (Jiang & Luo, 2021)Reviews GNNs for traffic flow/speed/passenger demand etc., modeling spatial & temporal dependencies.

16) How to Build a Graph-Based Deep Learning Architecture in Traffic Domain: A Survey (Ye et al., 2020)Deals with graph-based architectures for multiple traffic tasks; discusses challenges & benchmarks.

17) Auto Vehicle Counts via Moving Car Observer & YOLOv3 (MOM-DL)Uses vehicle detection (YOLOv3) from moving observer (a vehicle) to estimate flow, speed, density. Validated on Italian highway; error <3%.

18) Wrong-Way Vehicle Detection using YOLO + Centroid Tracking (Rahman et al., 2022)Detects vehicles via YOLO, tracks via centroid method, to detect direction violation.

19) Traffic Anomaly Detection using Deep Learning (Arun Kumar et al.)Uses CNN / DenseNet etc. to detect anomalies (accident etc.) from video; ~80% accuracy.

20) Object Detection in Traffic Scenarios – Comparison of Traditional and Deep Learning Approaches (Erabati et al.)Compares HOG+SVM vs SSD & YOLO with various backbones; looks at mAP and speed.

21) Multimodal Deep Learning Information Fusion for Fine-Grained Traffic State Estimation and Intelligent Traffic Control (Joshi & Kulkarni, 2023)Fuses different data modalities (video, perhaps sensors) for traffic state estimation.

22) Ensemble Deep Learning Framework for Traffic Accident Detection in Smart Cities (Maneesh Kumar & Tahseen, 2025)Proposes I3D-CONVLSTM2D with RGB + Optical Flow; mAP ~87%.

23) Detection of Traffic Signs for Autonomous Driving with Deep Learning (Sinan Uguz & Hayati Akgün, 2023) Autonomous detection of critical signs (school, pedestrian crossing etc.) with DL methods.

III. METHODOLOGY

The proposed traffic detection framework follows a multi-stage structure consisting of data collection, preprocessing, object detection, and classification, tracking with behavior analysis, decision-making, and system evaluation. The methodology is designed to ensure accuracy, robustness, and real-time performance.

A. Data Collection

Video streams are collected from traffic cameras deployed at intersections, road segments, and other key monitoring locations.
Possibly also incorporate other sensors (e.g. radar, LiDAR) if available.

Collect diversified data: different lighting (day/night), weather (rain, fog), traffic density (light, heavy), types of vehicles (cars, two-wheelers, buses, trucks), pedestrians.

1) Preprocessing

Frame extraction from video at a fixed rate.
Preprocessing includes noise suppression using filtering techniques, illumination normalization, and mitigation of shadow effects.
Data augmentation: flip, rotate, scale images; adjust brightness/contrast to increase robustness
Labeling/annotation of objects: bounding boxes, classes (vehicle types, pedestrians), possibly behaviors (turning, stopping, etc.).

2) Detection & Classification

A suitable deep learning–based object detection model is selected, such as YOLO variants, Faster R-CNN, or SSD, based on accuracy– speed trade-offs , depending on the trade-off between accuracy and inference speed
Possibly incorporate small-object detection enhancements (multi-scale feature maps, attention modules) to handle occlusions and distant objects.

3) Tracking & Behavior Analysis

Object tracking algorithms such as SORT, Deep SORT, or ByteTrack are applied to associate detections across consecutive frames.
From trajectories, derive behavior attributes: speed, lane changes, turning, near collisions, braking, stopping times; compute traffic density, flow rate etc.

4) Decision Module / Actionable Insights

Based on detection + tracking outputs, build rules / machine learning classifiers for incident detection (accidents, congestion, anomaly), or feeding into an adaptive signal control system.
Possibly use temporal models (LSTM, GRU) or combined CNN + RNN pipelines to predict near-future traffic state, so proactive decisions can be made.

5) Performance & Optimization

Model optimization: tune hyper parameters (learning rate, batch size, anchor sizes etc.), prune model / quantize if deploying on edge devices.
Ensure latency is acceptable (e.g. < 200 ms) for real-time response. Possibly use GPU or edge accelerators.

6) Evaluation

Performance is assessed using common evaluation metrics such as precision, recall, F1- score, and mean average precision (mAP); IDF1, MOTA etc. for tracking.
Evaluate on held-out test set, including challenging conditions (night, rain, occlusion).
Also, evaluate system end-to-end: e.g. for signal control, measure reduction in wait time, throughput improvement etc.

B. Algorithms

3.2.1 Object Detection

YOLOv8 : It is a recent real-time object detection architecture that is effective for identifying vehicles and pedestrians in dynamic traffic environments.Its efficiency and accuracy make it ideal for deployment in real-time systems.
CNN: Convolutional Neural Networks (CNNs) are capable of learning discriminative visual features, making them suitable for detecting vehicles and pedestrians in complex urban scenes. It can be trained on datasets like KITTI and UA-DETRAC for robust performance.

3.2.2 Object Tracking

Deep SORT: Deep SORT integrates visual appearance descriptors with motion cues to enable consistent multi-object tracking across successive frames , making it suitable for monitoring vehicle trajectories in real-time.
Kalman Filter with Gaussian Smoothed Interpolation (GSI): An enhancement to traditional tracking methods, this approach improves tracking accuracy by addressing issues like occlusion and detection misses.

3.2.3 Semantic Segmentation

U-Net: A convolution network designed for semantic segmentation, U-Net performs pixel- level segmentation of road scenes, supporting tasks such as lane identification and surface analysis, useful for tasks like lane detection and road surface analysis.

3.2.4 Anomaly Detection

Dual-Modality Vehicle Anomaly Detection: This method utilizes bilateral trajectory tracing to detect anomalies such as accidents or stalled vehicles, enhancing the system's ability to respond to unusual traffic events.

3.3 Frameworks & Tools

TensorFlow / Keras / PyTorch: These deep learning libraries provide the necessary tools to build, train, and deploy models for object detection, tracking, and segmentation.

OpenCV / YOLOv5: OpenCV is used for video processing operations such as frame extraction and preprocessing, whereas YOLOv5 provides an efficient solution for object detection.

IV. ADVANTAGES

1) High accuracy in complex scenes: Deep learning approaches automatically extract hierarchical visual patterns, allowing accurate identification of multiple object classes even in partially occluded traffic scenes.
2) Robustness to variations: With good preprocessing (illumination correction, noise reduction) plus data augmentation (flips, brightness, weather effects), The framework demonstrates reasonable adaptability across varying lighting conditions, weather scenarios, and environmental settings.
3) Real-time capability (with optimization / edge computing): Modern detection networks (especially lighter ones or optimized ones) can run with low latency, enabling near real-time detection and action, which is critical for traffic signal control, safety alerts, etc.
4) Automatic feature learning: Unlike older methods that require manual feature engineering, deep learning learns features from data itself, which can reduce human effort and adapt more easily to new/unseen scenarios.
5) Scalability: Once trained, the framework can be scaled (different cameras, intersections, cities) with retraining / fine-tuning, and can also incorporate more classes / behaviors over time. Also, it could be deployed on cloud/edge/IoT infrastructure.
6) Enables higher-level analytics: With tracking, behaviour estimation etc., the framework can support not just detection but also prediction, anomaly detection, helping with traffic management planning, safety interventions etc. A richer output than simple counting.

V. LIMITATIONS

1) High data requirements: The proposed system relies heavily on large volumes of annotated image and video data, which makes data acquisition and labeling time-consuming and costly.. Collecting, annotating such data is expensive.
2) Computation / hardware demands: Training deep learning models demands significant computational resources, often requiring GPUs or high-performance hardware video / many cameras) can be heavy unless optimized or deployed on capable hardware.
3) Generalization to unseen conditions: Even with data augmentation, models often perform poorly under extreme or rare conditions (heavy rain, fog, glare, occluded views, very crowded traffic), or when camera viewpoint changes dramatically.
4) Latency / real-time constraints: To be useful for traffic control or alerts, detection, tracking, decision making must happen fast. Large models plus high resolution video can introduce delays. Optimisation, pruning, or edge computing may be needed.
5) Interpretability / explainability: From an explainability standpoint, deep learning models often function as opaque systems, making it difficult to interpret decision-making behavior. For safety-critical applications (e.g. identifying why a misdetection happened in accident context) it can be hard to trace errors or to convince stakeholders.
6) Maintenance & updating: Over time, as traffic patterns, camera positions, lighting, vehicle types etc. change, you'll need to retrain or fine-tune models. Also, models might degrade if not updated. Data drift.
7) Privacy / data concerns: Video capture implies potential privacy implications (pedestrians faces, number plates etc.). Regulatory, ethical constraints on what data can be used / stored. Also possible security of system (if data is transferred). ‹though less discussed in some papers›
8) Edge cases and rare scenarios: Rare events (e.g. animals crossing, broken- down vehicles, unexpected obstruction) may be poorly represented in training data and so the model might fail in those. Also behavior recognition (overtaking, unusual maneuvers) is harder.

©IJRASET: All Rights are Reserved

REFERENCES

[1] Smart traffic management of vehicles using Faster R-R-CNN based deep learning method — Arindam Chaudhuri et al., Scientific Reports (2024) This work investigates the use of Faster R- CNN with adaptive background modeling and extended topological active nets to segment vehicles and address illumination/shadow issues in traffic videos.

[2] Intelligent Traffic Analysis System Using Deep Learning — Ayesha Asif Pailwan & B. D. Jitkar (2024) A review/integration of traffic analysis systems using CNNs—covers object detection, tracking, transfer learning, and real-time adaptation.

[3] Indian traffic sign detection and recognition using deep learning — R. K. Megalingam et al. (2023) Focused on detection & recognition of Indian road signs using adapted deep learning models, dealing with varied lighting and sign classes.

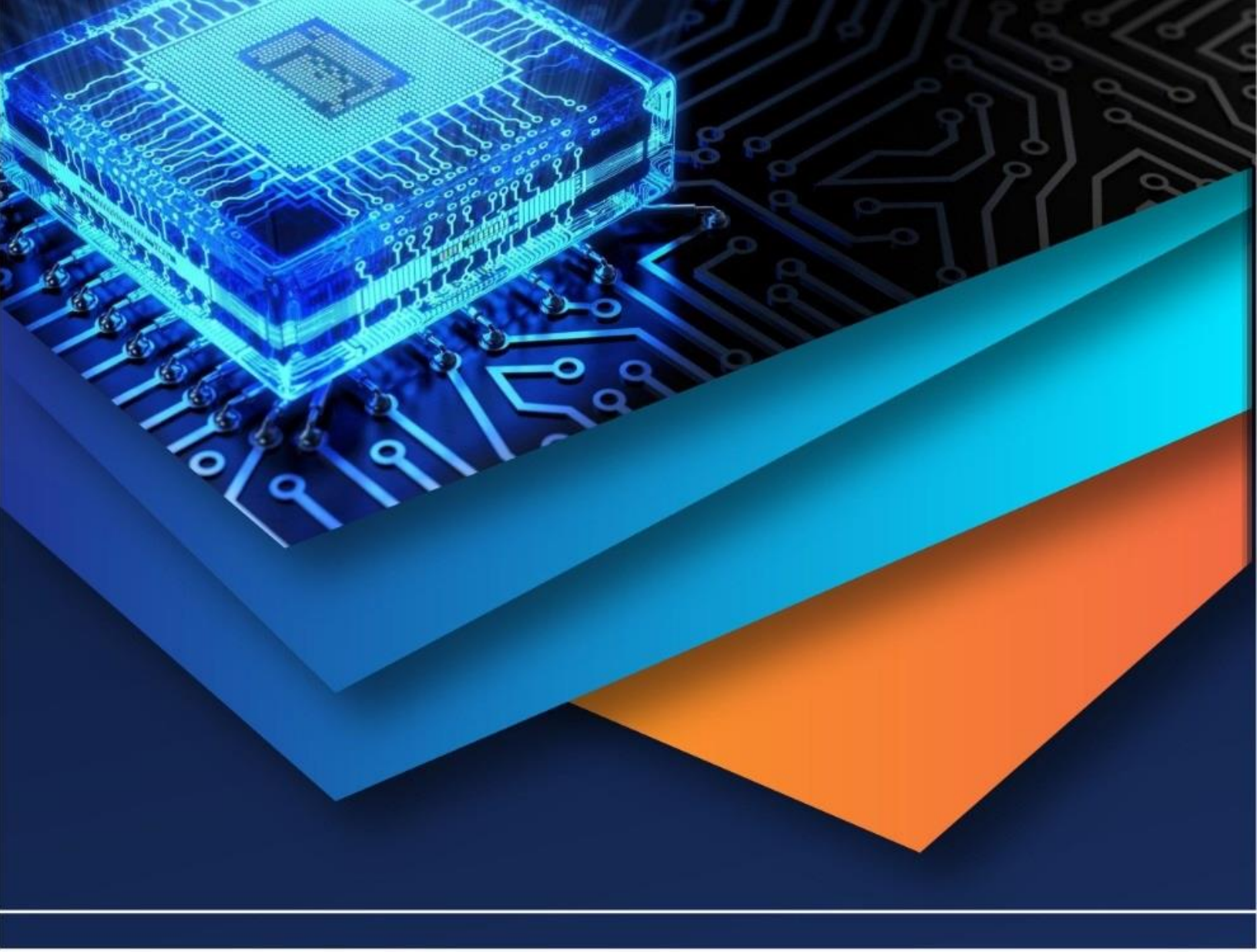
[4] A Vision-based System for Traffic Anomaly Detection using Deep Learning and Decision Trees — Armstrong Aboah et al. (2021) Combines YOLO (for object detection) with a decision-tree module for anomaly detection in traffic (e.g. accidents) and start/end time estimation.

[5] Intelligent Intersection: Two-Stream Convolutional Networks for Real-time Near Accident Detection — Xiaohui Huang et al. (2019) A two-stream CNN architecture (spatial + temporal) for detection, tracking, and near- accident prediction in traffic videos.

[6] Intelligent Infrastructure for Traffic Monitoring Based on Edge Computing and Computer Vision — J. Villa et al. (2024) Proposes a computer vision-based approach for real-time processing at the edge (i.e., embedding part of the system in edge devices) for traffic monitoring.

[7] Deep Learning-Based Detection for Traffic Control — Y. Yang et al. (2021) Presents a system to detect traffic density and adjust traffic light control strategy using deep learning.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)