Indian Journal of Modern Research and Reviews

This Journal is a member of the *'Committee on Publication Ethics'*

Online ISSN:2584-184X

Research Article

Design and Implementation of an IoT-Based Traffic Violation Detection Framework Using Image Processing

Dr. Manish Saraf [1], Prafulla Balshiram Jadhav [2*]

[1-2] Eklavya University, Damoh, Madhya Pradesh, India

Corresponding Author: * Prafulla Balshiram Jadhav

DOI: https://doi.org/10.5281/zenodo.21334525

Abstract

Rapid urbanisation, rising vehicle density and limited enforcement capacity have made traffic-rule compliance a major concern for public safety and urban administration. Conventional monitoring depends heavily on police personnel and manual inspection of closed-circuit television footage, which restricts coverage, delays response and increases the possibility of inconsistent decisions. This paper presents a compact design and implementation framework for an Internet of Things (IoT)-enabled traffic-violation detection system supported by image processing, computer vision, edge computing and secure event communication. The proposed architecture detects red-light jumping, absence of helmets, triple riding, wrong-way movement, lane misuse, illegal parking and over-speeding. Roadside cameras and sensors acquire traffic data, while an edge device performs frame enhancement, vehicle detection, tracking, behavioural analysis and automatic number-plate recognition. Once a probable violation is identified, the system generates a structured evidence package containing event images, a short video sequence, registration details, violation category, time, location and confidence score. Event-based communication reduces bandwidth consumption by transmitting only validated records rather than continuous video. A human verification stage is retained before administrative action in order to reduce false penalties and improve accountability. Since field observations and measured experimental results were not supplied, the paper defines a transparent validation plan based on precision, recall, F1-score, plate-recognition accuracy, latency, frame rate and bandwidth reduction. The framework offers a scalable foundation for smart-city traffic management while recognising the importance of privacy, cybersecurity, local calibration and legal oversight.

Manuscript Information

- **ISSN No:** 2584-184X
- **Received:** 07-06-2026
- **Accepted:** 10-07-2026
- **Published:** 13-07-2026
- **MRR:**4(7); 2026: 92-97

- **Plagiarism Checked:** Yes
- **Peer Review Process:** Yes

How to Cite this Article

Saraf M, Jadhav P B. Design and implementation of an IoT-based traffic violation detection framework using image processing. Indian J Mod Res Rev. 2026;4(7):92-97.

Access this Article Online

www.mrrjournal.in

KEYWORDS: Internet of Things, traffic violation detection, image processing, computer vision, automatic number-plate recognition, edge computing, intelligent transportation system.

1. INTRODUCTION

Road transport is essential for economic activity, mobility and access to public services, yet the rapid growth of motorised traffic has also produced congestion, unsafe driving behaviour and frequent violation of traffic rules. Red-light jumping, riding without a helmet, triple riding, wrong-way movement, unauthorised parking, lane indiscipline and speeding are common offences that increase accident risk and weaken public confidence in road administration. The challenge is particularly serious at busy intersections and on road corridors where continuous human observation is difficult.

Traditional enforcement methods include roadside police deployment, physical checkpoints, speed-control devices and manual examination of surveillance footage. These methods remain necessary, but their effectiveness is constrained by fatigue, limited manpower, poor visibility and the inability to observe every vehicle continuously. Passive cameras may record an incident, but the footage often requires lengthy review before the offending vehicle, exact time and supporting evidence can be identified.

IoT and computer vision provide an opportunity to transform surveillance from a passive recording function into an event-oriented decision-support system. Cameras, signal controllers, speed sensors, embedded processors, communication modules and cloud servers can operate as connected components. Image-processing algorithms can improve visual quality, detect vehicles and riders, recognise traffic signals, identify lane boundaries and estimate vehicle movement. Tracking algorithms then maintain vehicle identities across frames so that behaviour can be interpreted over time rather than from one isolated image.

The purpose of the proposed framework is not to replace legal authority or human judgement. It is designed to identify probable violations, organise relevant evidence and present cases to authorised personnel. This human-in-the-loop arrangement is necessary because emergency manoeuvres, temporary road diversions, police instructions, camera errors or unusual traffic conditions may otherwise lead to incorrect automated conclusions.

2. Problem Statement and Research Gap

Existing traffic-monitoring systems frequently operate as isolated solutions. One installation may detect red-light violations, another may record speed, while a separate application performs number-plate recognition. Such fragmentation makes it difficult to combine evidence, maintain audit trails and manage multiple types of violations through a common administrative platform. Systems that transmit continuous high-resolution video to a central server also consume substantial bandwidth and may respond slowly when network connectivity is weak.

A review of the technical requirements reveals five important gaps. First, many models are designed for a single offence and cannot interpret several traffic rules within one architecture. Second, vehicle detection is often separated from tracking, signal status, road geometry and plate recognition, although these elements must be jointly considered before an event can be treated as a probable violation. Third, insufficient attention is given to local edge processing, secure event transmission and offline buffering. Fourth, visual evidence is sometimes stored without a standard structure, integrity check or human verification stage. Fifth, privacy, access control and data-retention requirements are often treated as secondary issues rather than core design responsibilities.

The research therefore addresses the need for a unified, modular and secure IoT framework that can process traffic scenes near the source, detect multiple violations, generate reliable evidence and communicate only relevant events to a central platform.

3. AIM, OBJECTIVES AND RESEARCH QUESTIONS

The aim is to design and implement an IoT-based framework that integrates image processing, vehicle tracking, number-plate recognition and event-based communication for automated traffic-violation detection and evidence management.

3.1 Objectives

Design a layered IoT architecture for real-time traffic monitoring and event communication.

Develop an image-processing pipeline for vehicle, rider, helmet, signal, lane and number-plate detection.

Identify red-light jumping, helmet violations, triple riding, wrong-way driving, illegal parking, lane misuse and over-speeding.

Use multi-frame vehicle tracking to determine direction, stop-line crossing, lane position, speed and stationary duration.

Generate and securely transmit a structured digital evidence package for human verification.

Define a reproducible evaluation framework for accuracy, latency, reliability, bandwidth use and scalability.

3.2 Research Questions

How can IoT devices and image-processing modules be integrated into one traffic-enforcement workflow?

Which spatial and temporal features are required to distinguish lawful movement from a probable violation?

How can edge processing reduce response time and network load without weakening evidence quality?

How can number-plate recognition, confidence thresholds and human review improve the reliability of violation records?

Which technical and governance indicators should be used to validate the framework?

4. REVIEW OF RELATED LITERATURE

Digital image processing provides the foundation for extracting useful information from traffic scenes. Classical techniques include grayscale conversion, thresholding, edge detection, background subtraction, morphology and contour analysis (Gonzalez & Woods, 2018; Jain, 1989). Otsu's method is widely used for automatic threshold selection, while scale- and gradient-based feature descriptions have historically supported object recognition under changing image conditions (Otsu,

1979; Lowe, 2004; Dalal & Triggs, 2005). These methods remain useful for image enhancement, road-line extraction and plate preprocessing, although complex traffic scenes increasingly require learned models.

Modern object detectors based on convolutional neural networks can identify several classes in the same frame, including motorcycles, cars, buses, trucks, persons, helmets and number plates. Detection alone, however, does not explain movement. Vehicle tracking is therefore required to preserve identity across frames and estimate direction, speed and interaction with calibrated road regions. Kalman filtering provides a mathematical basis for predicting motion, while association methods connect detections over time (Kalman, 1960).

Automatic number-plate recognition commonly includes vehicle localisation, plate detection, geometric correction, enhancement, optical character recognition and format validation. Its performance is influenced by plate angle, camera distance, motion blur, illumination, damaged plates and partial obstruction. Multiple-view geometry and perspective transformation are useful when image coordinates must be related to actual road coordinates (Hartley & Zisserman, 2004).

IoT-based intelligent transportation systems connect cameras, signal controllers, sensors and administrative applications. Edge computing is especially relevant because continuous video transmission is expensive and vulnerable to network delay. Local processing allows probable violations to be identified near the camera and permits only metadata, selected images and short clips to be uploaded. This event-based arrangement supports faster response, local buffering and greater scalability. OpenCV and embedded computing platforms provide practical tools for implementing such pipelines (Bradski & Kaehler, 2008; Pajankar, 2019).

The literature indicates that technical accuracy alone is insufficient. Traffic evidence is sensitive because it can lead to penalties and may contain identifiable information. Secure storage, access control, traceable review and evidence integrity must therefore be built into the architecture. Information-security management principles, including role-based access, encryption and audit logging, are relevant to the design of such systems (International Organization for Standardization, 2018).

5. RESEARCH METHODOLOGY

5.1 Nature and Design of the Study

The study is technological, developmental, experimental and analytical. A prototype-based design is proposed in which hardware, image-processing modules, rule logic, IoT communication and an administrative dashboard are developed as interoperable components. The work is organised into five phases: requirement analysis, architecture design, model and rule development, system integration, and validation.

Phase	Main activity	Expected output
I	Traffic-rule analysis and requirement identification	Functional and legal requirement specification
II	Hardware, network and road-geometry design	Layered system architecture and calibrated regions
III	Detection, tracking and violation-rule development	Operational image-processing and rule modules
IV	IoT, database and dashboard integration	Secure event transmission and case-management interface
V	Testing and refinement	Performance report, error analysis and deployment recommendations

5.2 Data Sources and Sampling

Data may be collected from authorised traffic-camera streams, recorded intersection videos, manually captured road images, public datasets and signal-controller or speed-sensor records. Purposive site selection should cover controlled intersections, one-way roads, no-parking areas, high-density urban corridors, restricted lanes and speed-controlled zones. Recording conditions should include daylight, night, rain, glare, shadow and partial occlusion. The dataset must be divided into training, validation and test subsets at the video-sequence level so that near-duplicate frames do not appear in different subsets.

The primary unit of analysis is a vehicle trajectory associated with a traffic event. Each trajectory includes vehicle class, tracked position, direction, lane association, signal state, speed estimate, stop duration, number-plate output and violation label. Ground-truth annotations should be produced by trained reviewers using a documented labelling protocol.

5.3 Tools and Implementation Environment

A practical prototype may use high-resolution and infrared cameras, an edge-computing device, radar or another approved speed sensor, network connectivity, local encrypted storage and power backup. Python or C++ may be used for implementation, OpenCV for image processing, a deep-learning framework for detection, an OCR engine for plate recognition, MQTT or HTTPS for event communication, a relational or document database for evidence records and a web framework for the monitoring dashboard.

6. Proposed IoT Architecture

The proposed framework is divided into six layers. This separation improves maintainability because sensing, processing, communication, storage, application functions and governance can be upgraded independently.

Layer	Major components	Primary function
Sensing	Cameras, signal interface, radar/speed sensor, GPS, ambient sensor	Acquire visual, temporal, spatial and speed information
Edge processing	Frame processor, detector, tracker, rule engine, OCR	Interpret traffic events near the source
Communication	Wi-Fi, Ethernet, 4G/5G, MQTT/HTTPS	Transmit validated event records and device status
Cloud and storage	Server, database, evidence repository	Store cases, logs, media and system information
Application	Dashboard, search, alerts, reporting	Support review, verification and traffic analytics

Governance and security	Authentication, encryption, access roles, retention policy	Protect evidence and ensure accountable use

6.1 Operational Data Flow

Traffic video is captured by a fixed roadside camera and passed to the edge device. After preprocessing, the detector identifies vehicles, persons, helmets, traffic signals and number plates. A tracker assigns a persistent identity to each vehicle. The rule engine compares tracked movement with the signal state, stop-line position, lane geometry, road direction, speed threshold and restricted-zone configuration. When a probable violation is detected, the edge unit selects high-quality evidence frames, extracts the plate region, performs OCR, generates an event record and transmits the encrypted package to the central server. An authorised reviewer accepts, rejects or returns the event for further examination.

6.2 Image-Processing Pipeline

The pipeline begins with frame acquisition and region-of-interest selection. Processing only the stop line, lanes, parking zone, signal head and plate area reduces computation and false detections. Preprocessing may include resizing, denoising, contrast enhancement, shadow reduction, deblurring and perspective correction. Detected objects are represented by bounding boxes with class labels and confidence values. The centre of each bounding box is used by the tracking module to estimate displacement between frames. After camera calibration, pixel displacement may be converted into approximate real-world distance and direction.

A homography matrix maps image-plane coordinates to the road plane. This transformation is important for speed estimation, lane association and stop-line analysis. The method requires fixed reference points with known road positions. Calibration must be repeated whenever the camera angles, zoom or mounting position changes.

7. Traffic-Violation Detection Modules

7.1 Red-Light Jumping

The red-light module combines the verified signal state, stop-line boundary and tracked vehicle position. A probable event is created when a vehicle moves from the permitted side of the stop line to the prohibited side while the signal is red. Evidence should show the vehicle before crossing, during crossing and at a point where the registration plate is readable. Direct communication with the signal controller is preferable to visual signal-colour recognition because it provides the authoritative phase state.

7.2 Helmet Absence and Triple Riding

A motorcycle is first detected and associated with nearby persons. The head regions of the rider and passenger are examined for helmet presence. A helmet violation is generated only when the motorcycle-person association remains stable across several frames and the helmet confidence remains below a calibrated threshold. Triple riding is identified when more than two persons remain associated with one motorcycle over a minimum temporal window. Multi-frame confirmation reduces false decisions caused by temporary occlusion.

7.3 Wrong-Way and Lane Violations

Wrong-way detection compares the observed vehicle direction vector with the permitted direction assigned to the road segment. A strongly negative directional similarity indicates movement opposite to the allowed flow. Lane violations are detected by testing whether the tracked vehicle centre or footprint enters a prohibited lane polygon, crosses a continuous marking or remains outside its permitted lane beyond a specified duration. Semantic segmentation or manually calibrated polygons may be used to represent lane boundaries.

7.4 Illegal Parking and Over-Speeding

A no-parking area is represented as a calibrated polygon. When a tracked vehicle remains substantially stationary inside the polygon for longer than the permitted duration, the system records a probable parking violation. Short stops caused by congestion or signal queues must be excluded through contextual rules. Speed may be estimated from calibrated distance travelled over a known time interval; however, evidence used for legal enforcement should preferably be supported by an approved radar or speed sensor.

8. Number-Plate Recognition and Evidence Management

After a probable violation is detected, the system selects the clearest vehicle frame and localises the plate. The crop is rectified, sharpened and enhanced before OCR. The resulting text is checked against the expected registration format. A confidence-based policy is recommended: high-confidence outputs may proceed to preliminary acceptance, medium-confidence outputs require manual reading, and low-confidence outputs should be reprocessed or rejected. Several plate images may be stored so the reviewer can compare alternatives.

Evidence field	Description
Event and device ID	Unique case and source identifiers
Vehicle and plate data	Vehicle class, plate image, OCR text and confidence
Violation information	Rule category, signal/lane/zone context and confidence
Time and location	Timestamp, configured site and optional GPS coordinates
Media evidence	Selected images and a short event clip
Integrity data	Cryptographic hash, transmission status and audit record
Review data	Reviewer identity, decision, comments and decision time

A cryptographic hash should be generated for every evidence file so that unauthorised modification can be detected. The audit log should preserve status changes from detection to final review. Raw continuous footage should not be retained indefinitely; only evidence required by an approved retention schedule should be stored.

9. IoT Communication, Database and Dashboard

The framework adopts event-based communication. Instead of transmitting all video, the edge device sends metadata, selected images, a short clip, device-health information and periodic logs. MQTT is suitable for lightweight messaging, while HTTPS may be used for secure media upload and administrative services. When connectivity fails, encrypted events are stored locally and synchronised after the connection is restored. Each event should receive a server acknowledgement to prevent loss or duplicate processing.

The database should include device, event, user and audit entities. The device entity stores installation location, operational status, software version and last communication time. The event entity stores the violation type, plate information, confidence values, evidence paths and verification status. The user entity records roles and access levels, while the audit entity records every action affecting a case. The dashboard should support pending-case review, media display, plate search, device monitoring, statistical summaries and authorised export.

10. Evaluation Framework

No measured field results are claimed because a validated dataset and prototype test report were not supplied. After implementation, the system should be evaluated independently for each module and for the complete event workflow. Ground truth must be established through manual annotation and reviewer agreement.

Metric	Formula/meaning	Purpose
Precision	$TP / (TP + FP)$	Proportion of reported violations that are genuine
Recall	TP / (TP + FN)	Proportion of actual violations successfully detected
F1-score	2PR / (P + R)	Balanced measure of precision and recall
IoU	Overlap area / union area	Quality of object localisation
OCR accuracy	Correct characters / total characters x 100	Plate-reading performance
Latency	Server receipt time - event time	End-to-end response speed
Frame rate	Processed frames / processing time	Real-time processing capacity
Bandwidth reduction	(Cloud stream - edge events) / cloud stream x 100	Network savings from edge processing

Additional indicators should include tracking identity switches, full-plate accuracy, mean absolute speed error, packet-delivery rate, evidence retrieval success, dashboard response time and reviewer agreement. Performance should be reported separately for daylight, night, rain, dense traffic, partial occlusion and different camera distances. This stratified analysis is necessary because an overall average may conceal weaknesses under difficult conditions.

10.1 Validation and Reliability

Training, validation and test sequences must be separated before model tuning. Confidence thresholds should be calibrated only on validation data. The final test set should remain unseen until the design is fixed. Repeated trials, error categorisation and comparison with a baseline cloud-only system will strengthen reliability. For labelled violations, at least two reviewers should independently verify a sample, and disagreements should be resolved through an agreed protocol.

11. DISCUSSION

The major contribution of the framework is the integration of detection, tracking, rule evaluation, plate recognition, evidence generation and administrative review within one modular system. A single image may show a vehicle but cannot reliably establish wrong-way driving, stop-line crossing or excessive parking duration. Temporal tracking therefore provides the behavioural context required for traffic-rule interpretation.

Edge processing improves operational efficiency because high-resolution video does not need to be continuously transmitted. It also allows the system to continue detecting and buffering events during temporary network interruption. The design remains scalable because additional roadside units can publish standardised event records to the same central platform. Nevertheless, edge devices require sufficient computational capacity, thermal management, storage protection and secure software updates.

Number-plate recognition is a critical but vulnerable stage. Incorrect OCR can associate evidence with the wrong vehicle, so a plate result should never be accepted solely because one frame produced a plausible string. The framework therefore uses multiple images, format validation, confidence thresholds and human review. Similar caution is required for helmet detection, where caps, scarves, shadows and partial visibility may mislead a classifier.

The system should be regarded as a decision-support mechanism rather than an autonomous legal authority. Human verification protects citizens from penalties based on technical error and allows contextual exceptions to be considered. It also creates a traceable point of accountability. Transparent grievance procedures and controlled access to evidence are necessary for public trust.

12. Security, Privacy and Ethical Safeguards

Traffic surveillance can capture identifiable vehicles, faces and movement patterns. The design must therefore follow the principles of necessity, proportionality, limited retention and purpose restriction. Data should be encrypted in transit and at rest. Device identities should be authenticated; user access

should be role-based and sensitive actions should require multi-factor authentication. Signed software updates, tamper-resistant logs and periodic security audits are necessary to reduce cyber risk. Faces and unrelated number plates may be masked where they are not required for evidence. Complete footage should be accessible only to authorised personnel and retained only for the approved period. The system should provide a documented process for correcting plate-recognition errors, challenging an event and recording the final administrative decision. Model performance should also be checked across vehicle types, lighting conditions and locations to detect systematic bias.

13. Limitations and Recommendations

The framework may be affected by rain, fog, glare, darkness, camera vibration, dense traffic, occlusion, dirty or non-standard plates and changes in lane markings. Speed estimates may be unreliable without accurate calibration, while low-cost edge devices may not sustain the required frame rate. Traffic rules and evidence requirements also differ across jurisdictions. These limitations mean that deployment must be locally configured and tested before operational use.

Install cameras at a stable height and angle, with night support where required.

Integrate the authorised signal-controller state and approved speed sensor whenever possible.

Train and validate models on locally representative roads, vehicles, plates and weather conditions.

Use multi-frame confirmation, confidence thresholds and evidence-quality checks.

Maintain encrypted local buffering, secure updates, audit logs and a defined retention policy.

Require human verification before penalties or other legal action are initiated.

14. Future Scope

Future development may include transformer-based scene understanding, vehicle re-identification across intersections, privacy-preserving federated learning, mobile-phone and seat-belt violation detection, altered-plate detection, emergency-vehicle recognition, accident notification and cooperative analysis across multiple cameras. Explainable artificial intelligence may help reviewers understand why an event was flagged, while digital-twin simulation can support testing before field deployment. Research is also required on governance standards that balance enforcement efficiency with privacy, transparency and procedural fairness.

15. CONCLUSION

An IoT-based traffic-violation detection system can improve the reach and consistency of road monitoring when image processing, vehicle tracking, edge computing and secure evidence management are designed as one integrated workflow. The proposed framework detects multiple traffic offences, interprets movement over time, recognises number plates and transmits structured events to a central dashboard. Event-based communication reduces network load, while local buffering improves resilience during connectivity failure.

Technical automation must be accompanied by responsible governance. Environmental variability, OCR error, occlusion and model bias can produce incorrect results. Accordingly, the framework includes confidence-based filtering, multi-frame evidence, cryptographic integrity, access control, limited retention and human verification. With local calibration and rigorous field validation, the system can support safer roads, faster case review and evidence-based smart-city traffic administration.

REFERENCES

1. Bradski G, Kaehler A. Learning OpenCV: Computer vision with the OpenCV library. Sebastopol (CA): O'Reilly Media; 2008.
2. Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2005. p. 886-893.
3. Gonzalez RC, Woods RE. Digital image processing. 4th ed. Harlow: Pearson; 2018.
4. Hartley R, Zisserman A. Multiple view geometry in computer vision. 2nd ed. Cambridge: Cambridge University Press; 2004.
5. International Organization for Standardization. ISO/IEC 27001: Information security management systems. Geneva: ISO; 2018.
6. Jain AK. Fundamentals of digital image processing. Englewood Cliffs (NJ): Prentice Hall; 1989.
7. Kalman RE. A new approach to linear filtering and prediction problems. J Basic Eng. 1960;82(1):35-45.
8. Lowe DG. Distinctive image features from scale-invariant keypoints. Int J Comput Vis. 2004;60(2):91-110.
9. Otsu N. A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern. 1979;9(1):62-66.
10. Pajankar A. Raspberry Pi computer vision programming. Birmingham: Packt Publishing; 2019.
11. Rosebrock A. Deep learning for computer vision with Python. PyImageSearch; 2017.
12. Szeliski R. Computer vision: Algorithms and applications. 2nd ed. Cham: Springer; 2022.

About the Corresponding Author

Prafulla Balshiram Jadhav is affiliated with Eklavya University, Damoh, Madhya Pradesh, India. His academic interests include emerging technologies, engineering applications, and interdisciplinary research. He is committed to advancing innovative solutions through research and scholarly contributions, with a focus on practical applications and the dissemination of scientific knowledge in contemporary fields.