# Traffic Sign Recognition to Automatic Speed Control in IoT-Enabled Electric Vehicles

**Dr.Prosanjeet Sarkar[1], Sharda Ramkrushna Giripunje [2], Pradnya Kishor Sangolkar [3], Khushi Vishwajit Meshram[4], Arya Sunil Ayatwar[5], Sushma Ishwardas Shende[6]**

Dr. Prosanjeet Sarkar [1] Department of Electrical engineering
Tulshiramji gaikwad patil college of engineering and technology Mohgaon Nagpur, India
Sharda Ramkrushna Giripunje [2] Department of Electrical engineering
Tulshiramji gaikwad patil college of engineering and technology Mohgaon Nagpur, India
Pradnya Kishor Sangolkar [3] Department of Electrical engineering
Tulshiramji gaikwad patil college of engineering and technology Mohgaon Nagpur, India
Khushi Vishwajit Meshram[4] Department of Electrical engineering
Tulshiramji gaikwad patil college of engineering and technology Mohgaon Nagpur, India
Arya Sunil Ayatwar[5] Department / Institution / University, City, Country
Sushma Ishwardas Shende[6] Department of Electrical engineering
Tulshiramji gaikwad patil college of engineering and technology Mohgaon Nagpur, India
Corresponding Author Email: sarkarprosanjeet08@gmail.com | ORCID: https://orcid.org/0000-0002-4659-6271

**How to Cite this Article**:

Sarkar, P., Giripunje, S. R., Sangolkar, P. K., Meshram, K. V., Ayatwar, A. S. & Shende, S. I. (2026). Traffic Sign Recognition to Automatic Speed Control in IoT-Enabled Electric Vehicles. International Journal of Creative and Open Research in Engineering and Management, <i>02</i>(9), 1-9. https://doi.org/10.55041/ijcope.v2i9.146



https://doi.org/10.55041/ijcope.v2i9.146

**Abstract—**
Traffic sign recognition has been used as an input for managing speed of electric vehicles that are connected with their systems. This paper discusses how to integrate a visual recognition system with embedded propulsion control and IoT monitoring. Fifteen full-text studies published from 2021 to 2025 were reviewed. Literature contains deep learning techniques to detect and classify traffic signs; Indian roads dataset; IoT speed monitoring architecture; prototype-level speed control system. Recent detection studies show that end-to-end deep models can identify signs under complex conditions; for example, a refined Mask R-CNN study on 87 Indian sign categories reported 97.08% precision, while a YOLOv5 comparison reported 97.70% mAP@0.5 on its dataset. Control literature shows that pulse-width modulation (PWM) motor control, sensor feedback, and cloud logging can convert a known speed limit to an observable and traceable vehicle response. A low-cost prototype architecture is synthesized from the reviewed evidence: an ESP32-CAM captures signals, an ESP32 coordinates decisions and connectivity, an L298N applies pulse-width modulation to a DC traction motor, and a display/cloud service provides driver and supervisor feedback. The review finds that the limiting issue is no longer recognition accuracy alone: safe deployment requires confidence-aware decisions, low end-to-end latency, fail-safe override, locally representative data, and cybersecurity-aware IoT design. This model can serve as an academic and practical basis to develop an IoT-based electric vehicle speed control system.

**Keywords—** traffic sign recognition; electric vehicles; Internet of Things; automatic speed control; convolutional neural networks; intelligent transportation systems.

## I. INTRODUCTION

Traffic signs convey speed restrictions, restrictions on right-of-way and prohibitions, and warnings that must be understood quickly. For electric vehicles (EVs), the same information can be communicated to an assistance system which can suggest or limit a traction command. This is desirable since EV propulsion is already electronically controlled and can be adjusted using the motor controller. Based on the literature studied, this task is related to intelligent transportation systems, computer vision, embedded control, and connected sensing [1]-[15].

The two coupled decisions in a practical system are perception and control. In perception, a sign has to be sighted and categorized in a moving, cluttered image. In actuation, the identified sign must be sufficiently credible and relevant to the current path before it is converted to a bounded speed command. A correctly classified sign may still be out-of-date, associated with another lane, obscured, or otherwise inappropriate for immediate control.

This review synthesizes the provided literature and project material around traffic-sign recognition, automatic speed control, IoT-enabled monitoring, and a low-cost embedded EV prototype. It outlines the review method, examines traffic-sign perception and connected EV monitoring, synthesizes the prototype architecture, discusses control and safety, identifies research gaps, and proposes future directions.

## II. LITERATURE REVIEW

Previous research on traffic-sign recognition, electric-vehicle speed control, and IoT-based vehicle monitoring has primarily focused on three interconnected areas: deep-learning-based traffic-sign recognition, IoT-enabled speed monitoring and control, and integrated vehicle automation prototypes.

### A. Traffic-Sign Recognition Using Deep Learning

Qiao [1] investigated traffic-sign recognition using convolutional neural networks (CNNs), particularly under complex natural environments. The study highlights the importance of robust image datasets and suitable augmentation techniques for improving recognition performance. Traditional recognition approaches may rely on color, shape, edge detection, and region-of-interest processing; however, these methods can be affected by illumination, viewing angle, motion blur, occlusion, weather conditions, and background clutter. Deep-learning methods provide an alternative by learning visual features directly from labeled image data.

Megalingam *et al.* [2] specifically investigated Indian traffic-sign detection and recognition using deep learning. Their dataset contained 6,480 images, 7,056 traffic-sign instances, and 87 categories. The refined Mask R-CNN approach reported 97.08% precision under the stated evaluation conditions. This work demonstrates the importance of using datasets representative of the actual geographical environment in which a traffic-sign recognition system is intended to operate.

Kumar *et al.* [8] examined a traffic-sign recognition pipeline involving region-of-interest extraction and color/shape-based visual cues before classification. Their work indicates that preprocessing and temporal tracking can contribute to more reliable recognition in practical environments. Zhu and Yan [9] compared YOLOv5 and SSD for traffic-sign recognition and reported 97.70% mAP@0.5 for YOLOv5 compared with 90.14% for SSD under their experimental conditions.

Although these studies demonstrate high recognition performance, their reported metrics cannot be directly compared because the datasets, number of classes, hardware platforms, evaluation protocols, and environmental conditions differ. Therefore, recognition accuracy alone does not establish whether a system is suitable for automatic vehicle-speed control.

### B. IoT-Based Electric Vehicle Monitoring and Speed Control

IoT technologies have been increasingly applied to electric vehicles to enable continuous monitoring of vehicle and operating parameters. Putra and Ma'arif [3] proposed an IoT-based electric-car speed-monitoring system using an optocoupler, Arduino, LCD, SIM800L, and ThingSpeak. The architecture demonstrates how vehicle-speed information can be collected and transmitted to a remote monitoring platform.

Siva Kumar *et al.* [4] proposed an IoT-based monitoring and control architecture for electric vehicles. Their system incorporated onboard sensors, a microcontroller gateway, cloud analytics, and a user interface. Parameters such as battery condition, temperature, vehicle speed, GPS position, and motor efficiency were considered for monitoring.

Antoine *et al.* [10] investigated an IoT-based EV speed-management strategy incorporating PWM motor control and monitoring. Their reported random-forest voltage-estimation experiment achieved an $R^2$ value of 98.8% under the stated experimental conditions. The study demonstrates the potential of combining embedded motor control with IoT-based monitoring.

However, these studies also indicate an important architectural consideration: cloud connectivity should not be treated as a prerequisite for safety-critical control. A communication failure should not prevent a vehicle from responding to a locally detected speed restriction or emergency condition. Therefore, local control and remote monitoring should be separated, with IoT primarily supporting monitoring, logging, diagnostics, and supervisory functions.

### C. Integrated Traffic-Sign Recognition and Vehicle Control

Several studies have attempted to combine traffic-sign recognition with automatic speed adjustment and accident-avoidance functions. Arul Mozhi *et al.* [5] proposed an integrated prototype involving camera-based recognition, IoT sensing, obstacle warning, and speed response. Arunkumar *et al.* [6] examined IoT-based speed control for a semi-autonomous electric cargo vehicle using vehicle, battery, payload, GPS, and proximity-related information.

Latha *et al.* [7] proposed linking traffic-sign detection with vehicle parameters and speed control, highlighting the importance of measuring system latency. Goma *et al.* [11] investigated an embedded real-time recognition and response architecture, while Kamalathiyagarajan *et al.* [12] considered YOLO-based recognition of Indian traffic signs with live-video detection and speed adjustment.

Gunasekar *et al.* [13] investigated automatic vehicle speed control using zone-related information, including school and high-risk areas. Patil *et al.* [14] studied traffic-sign classification using a CNN/webcam arrangement with display-based feedback. Gayathri *et al.* [15] proposed an IoT-based traffic-sign detection and speed-control approach incorporating additional contextual information such as vibration, location, and driver state.

### D. Research Gap

The reviewed literature demonstrates considerable progress in traffic-sign recognition and IoT-enabled vehicle monitoring, but several limitations remain. First, recognition performance is frequently reported independently of the complete perception-to-control process. Consequently, parameters such as recognition-to-command latency, false intervention rate, missed intervention rate, speed-tracking error, and braking response are not consistently evaluated.

Second, correct classification of a traffic sign does not necessarily mean that the sign should immediately control the vehicle. A detected sign may belong to another lane, another road direction, or a different traffic context. Therefore, confidence assessment, temporal persistence, lane/context association, and operating-state validation are required before converting recognition output into a motor command.

Third, many IoT-based systems depend on cloud communication for monitoring, whereas safety-related decisions should remain locally executable. The reviewed literature therefore supports an architecture in which the ESP32-CAM performs visual sensing, the controller determines the appropriate command, the L293D interfaces with the motor, and IoT services provide monitoring and logging.

The proposed study builds upon these findings by integrating traffic-sign recognition, embedded motor-speed control, and IoT monitoring into a single low-cost prototype. The project specifically considers four predefined traffic-sign classes—START, STOP, SPEED LIMIT 20, and SPEED LIMIT 60—and maps these classes to corresponding prototype motor-control commands.

## III. METHODOLOGY

The proposed work follows a prototype-oriented embedded-system methodology combining computer vision, microcontroller-based control, PWM motor driving, and IoT monitoring. The overall system is designed to demonstrate the conversion of a visually detected traffic sign into an appropriate vehicle-speed command.

### A. System Architecture

The proposed system consists of an ESP32-CAM, NodeMCU, L293D motor driver, BO DC motor, power supply, and LCD/OLED or IoT monitoring interface. The basic functional sequence is:

**Traffic Sign → ESP32-CAM → Sign Recognition → Speed Command → L293D Motor Driver → BO DC Motor**

The ESP32-CAM provides the visual sensing interface and captures traffic-sign images. The NodeMCU provides additional control and communication functionality. The L293D acts as the interface between the controller and the BO DC motor, while PWM is used to control the motor command. An LCD/OLED and IoT service can provide non-critical monitoring information.

### B. Traffic-Sign Recognition

The system considers four predefined traffic-sign classes:

| Detected Sign | Prototype Command | System Interpretation |
|---|---|---|
| START | Start motor / starting PWM | Vehicle starts |
| STOP | PWM = 0% | Vehicle stops |
| SPEED LIMIT 20 | Lower PWM | Low-speed state |
| SPEED LIMIT 60 | Higher PWM | Higher-speed state |

These four classes provide a controlled demonstration of the relationship between visual sign recognition and motor control. The source project specifically defines these four traffic-sign categories.

### C. Motor-Control Method

The L293D motor driver is used to interface the controller with the BO DC motor. PWM is applied to control the motor's operating level. The prototype specifies example duty-cycle values of 0%, 30%, 60%, and 90%, corresponding to stop, low, medium, and high prototype operating levels.

These PWM percentages should not be interpreted directly as vehicle speeds in km/h. Actual motor speed depends on factors such as battery voltage, motor characteristics, vehicle load, wheel size, and mechanical configuration. Experimental calibration is therefore required before relating PWM duty cycle to a particular physical speed.

### D. Direction and Control Logic

For forward operation, the project specifies IN1 = HIGH and IN2 = LOW. Reverse operation is represented by IN1 = LOW and IN2 = HIGH, while IN1 = LOW and IN2 = LOW represents stop/coast. The BO DC motor is not connected directly to the ESP32 GPIO pins; instead, the L293D provides the required motor-driving interface.

For improved control reliability, the recognition output should not immediately determine the motor command from a single image frame. The recommended sequence is:

**Traffic-sign detection → Confidence verification → Temporal persistence → Context/lane verification → Target-speed determination → Current-speed comparison → PWM command → Motor response → Event logging**

This approach addresses the limitation that a correctly classified sign may not always be relevant to the vehicle's immediate trajectory.

### E. IoT Monitoring

The IoT subsystem is intended primarily for monitoring, logging, diagnostics, and supervisory functions. Information such as detected sign class, speed state, timestamp, and other available vehicle parameters may be displayed locally or transmitted to a cloud platform.

The safety-related control loop remains local so that loss of network connectivity does not prevent the vehicle from responding to a detected STOP or speed-limit condition. This separation between local control and cloud monitoring is consistent with the limitations identified in the reviewed literature.

### F. Prototype Implementation

The prototype is implemented using the hardware specified in the project material. The functional roles are summarized in Table I.

TABLE I. PROJECT HARDWARE AND FUNCTIONAL ROLE

| Component | Functional Role | Project Function |
|---|---|---|
| ESP32-CAM | Visual sensing | Captures traffic-sign images and performs/initiates recognition |
| NodeMCU | Control/communication | Provides additional control and IoT communication |
| L293D | Motor-driver interface | Applies PWM and interfaces the controller with the motor |

| | | |
|---|---|---|
| BO DC Motor | Prototype traction motor | Represents the EV drive motor |
| Power Supply | Electrical source | Provides appropriate controller and motor power |
| LCD/OLED / IoT | Monitoring | Displays sign, speed state, and non-critical telemetry |

The component roles correspond to the project architecture documented in the supplied paper.

## IV. RESULTS AND DISCUSSION

The reviewed evidence and the proposed prototype architecture demonstrate the feasibility of connecting traffic-sign recognition with embedded vehicle-speed control and IoT monitoring. Because the supplied work is a review and prototype-oriented study rather than a newly measured experimental dataset, the reported recognition values from individual publications are presented in their original study contexts and are not statistically pooled.

### A. Literature-Based Performance Comparison

TABLE II. COMPARATIVE SUMMARY OF SELECTED STUDIES

| Study | Main Focus | Reported/Identified Contribution | Design Implication |
|---|---|---|---|
| Qiao [1] | CNN recognition | Recognition under complex environments | Robust datasets and augmentation |
| Megalingam *et al.* [2] | Indian traffic signs | 87 categories; 97.08% stated precision | Local datasets are important |
| Putra and Ma'arif [3] | IoT speed monitoring | Arduino, LCD, SIM800L, ThingSpeak | Keep safety control independent of cloud |
| Siva Kumar *et al.* [4] | EV IoT monitoring | Sensors, gateway, cloud and interface | Keep safety logic local |
| Arul Mozhi *et al.* [5] | Integrated prototype | Camera, IoT, obstacle warning and speed response | Carefully fuse evidence |
| Zhu and Yan [9] | YOLOv5 recognition | 97.70% mAP@0.5 in stated experiment | Benchmark on target camera |
| Antoine *et al.* [10] | EV speed management | PWM + IoT; RF $R^2$ = 98.8% | Validate under different operating loads |
| Gunasekar *et al.* [13] | Zone-speed control | School/high-risk-zone speed reduction | Combine map and visual evidence |

The comparison indicates that different studies address different parts of the complete system. Deep-learning research concentrates primarily on recognition, whereas IoT studies emphasize monitoring and communication. Integrated prototypes attempt to combine these functions but do not consistently report complete end-to-end safety metrics.

### B. Prototype Command Mapping

The proposed prototype translates the recognized traffic signs into motor commands as shown below.

TABLE III.TRAFFIC-SIGN COMMAND MAPPING

| Detected Sign | Motor Command | Interpretation |
|---|---|---|
| START | Starting PWM | Vehicle starts |
| STOP | 0% PWM | Vehicle stops |
| SPEED LIMIT 20 | Lower PWM | Low-speed state |
| SPEED LIMIT 60 | Higher PWM | Higher-speed state |

This mapping provides a simple and demonstrable relationship between computer-vision output and motor-control action.

### C. PWM Control Levels

TABLE IV. PROTOTYPE PWM LEVELS

| PWM Duty Cycle | Prototype Action | Qualification |
|---|---|---|
| 0% | Stop | Motor command zero |
| 30% | Low speed | Example level; experimental calibration required |
| 60% | Medium speed | Depends on battery, motor, load and wheel size |

| | | |
|---|---|---|
| 90% | High speed | Not directly equivalent to km/h |

The PWM levels demonstrate the control concept but should not be considered calibrated road-speed values. The actual relationship between duty cycle and vehicle speed must be established experimentally for the particular motor, battery, load, and mechanical configuration.

**D. Discussion of Findings**

The literature indicates that high traffic-sign recognition accuracy can be achieved using modern deep-learning approaches. For example, the reviewed studies report 97.08% precision for a refined Mask R-CNN implementation on an Indian traffic-sign dataset and 97.70% mAP@0.5 for YOLOv5 in another stated experiment. However, these values are obtained under different experimental conditions and therefore cannot be interpreted as a direct performance comparison.

For the proposed prototype, the important issue is therefore not only whether the sign can be recognized but whether the recognition can be converted into a reliable and appropriate control response. Environmental conditions, camera placement, sign size, lighting, image quality, battery voltage, motor characteristics, vehicle load, and wheel size can all affect system behavior.

Accordingly, future experimental evaluation should measure:

- traffic-sign detection precision and recall;
- mAP and classification accuracy;
- recognition-to-command latency;
- speed-tracking error;
- false slow-down/intervention rate;
- missed-intervention rate;
- response during communication loss;
- motor and battery energy consumption; and
- behavior under different weather, road, and traffic conditions.

These measurements would provide a more complete evaluation than recognition accuracy alone.

**E. Overall System Outcome**

The proposed architecture demonstrates how ESP32-CAM-based visual sensing, embedded decision-making, L293D-based PWM motor control, and IoT monitoring can be integrated into a low-cost EV prototype. The system provides a practical research platform for investigating automatic speed adjustment based on predefined traffic signs.

However, the supplied study identifies the prototype as an educational/research demonstration rather than a certified road-going EV safety system. Real-world deployment would require representative datasets, quantified end-to-end performance, fail-safe mechanisms, cybersecurity measures, and extensive validation under realistic operating conditions.

Important formatting point: In your IEEE paper, place Table I immediately after the first paragraph where the hardware architecture is discussed, Table II after the literature-comparison discussion, and Tables III–IV in the Results/Discussion section immediately after their first references. The source paper already follows this general organization with four tables and one conceptual flow figure.

## V. CONCLUSION

Traffic-sign recognition, embedded speed control, and IoT monitoring can be connected into a unified EV automation architecture. The reviewed literature demonstrates progress in deep-learning sign recognition, PWM motor control, connected vehicle monitoring, and integrated prototype systems. The attached project material provides a practical low-cost platform using ESP32-CAM, NodeMCU, L293D, BO DC motor, and four predefined traffic-sign classes.

The central conclusion is that recognition accuracy alone is insufficient for safe automatic speed control. A useful system must combine confidence-aware perception, temporal and contextual validation, bounded local control, feedback, fail-safe behavior, and IoT monitoring that does not compromise local safety. The prototype provides an educational and research foundation, while representative datasets, quantified end-to-end performance, cybersecurity, and real-world validation remain necessary steps toward roadworthy deployment.

## ACKNOWLEDGMENT

The authors are thankful to the Department of Electrical Engineering, Tulsiramji Gaikwad Patil College of Engineering and Technology for academic support.

## REFERENCES

[1] Qiao X. Research on traffic sign recognition based on CNN deep learning network. Procedia Comput Sci. 2023;228:826-837. doi:10.1016/j.procs.2023.11.102.

[2] Megalingam RK, Thanigundala K, Musani SR, Nidamanuru H, Gadde L. Indian traffic sign detection and recognition using deep learning. J Traffic Transp Eng (Eng Ed). 2022.

[3] Putra RA, Ma'arif A. Internet of Things (IoT) based speed monitoring system for electric cars. Buletin Ilmiah Sarjana Teknik Elektro. 2024;6(2):182-189. doi:10.12928/biste.v6i2.11317.

[4] Siva Kumar JSV, Rohini D, Chandu B. IoT based monitoring and control system for electric vehicles. International Research Journal of Education and Technology. 2025;8(11):24-31.

[5] Arul Mozhi S, Rithvik B, Maya Krishnan S, Saran Kumar M, Saran J. IoT based speed control and accident avoidance using AI road sign. Alochana Journal. 2024;13(12):837-842.

[6] Arunkumar PL, Ramaswamy M, Murugesh TS. IoT based speed control for semi-autonomous electric on-road cargo vehicle. Int J Adv Comput Sci Appl. 2022;13(3):263-272.

[7] Latha R, Tamilselvan S, Vijay S, Vigneshwar G. IoT based speed control and accident avoidance using AI road sign detection system. International Journal of Creative Research Thoughts. 2025;13(2):g727-g735.

[8] Kumar N, Kurrey S, Singh S, Sahu D, Verma VP. Traffic sign recognition. International Journal for Research Trends and Innovation. 2022;7(6):361-366.

[9] Zhu Y, Yan WQ. Traffic sign recognition based on deep learning. Multimedia Tools and Applications. 2022;81:17779-17791. doi:10.1007/s11042-022-12163-0.

[10] Antoine G, Mikeka C, Bajpai G, Jayavel K. Speed management strategy: designing an IoT-based electric vehicle speed control monitoring system. Sensors. 2021;21(19):6670. doi:10.3390/s21196670.

[11] Goma E, Baraneedharan S, Chethan SN, Dinesh Reddy M. IoT based speed control monitoring and accident avoidance using AI traffic sign detection. International Journal of Innovative Research in Technology. 2025;11(10):3668-3673.

[12] Kamalathiyagarajan S, Kalidhsan S, Kavikuamr VJ, Sathish S, Manish S. IoT based speed control and accident avoidance using AI road sign detection system. International Journal of Advanced Research in Science, Communication and Technology. 2025;5(3):491-497. doi:10.48175/IJARSCT-26365.

[13] Gunasekar K, Manoj N, Rakesh T, Arthi P. Intelligent IoT based automatic vehicle speed control and alert system. International Journal for Research in Applied Science & Engineering Technology. 2025;13(4):2829-2835. doi:10.22214/ijraset.2025.68539.

[14] Patil AA, Suryavanshi SS, Patil SK, Choudhari BR, Budaragade AP. Traffic sign detection and recognition. International Journal of Advance Research and Innovative Ideas in Education. 2022;8(6):1107-1114.

[15] Gayathri T, Anbumani P, Prabakaran S, Varun S, Velumani K, Sudharshan M. IoT based speed control monitoring and accident avoidance using AI traffic sign detection. International Journal of Innovation Studies. 2025;9(2):429-438.