Journal of
Microcontroller Engineering and Applications

ISSN: 2455-197X
Volume 13, Issue 1, 2026
January–April
DOI (Journal): 10.37591/JoMEA

https://journals.stmjournals.com/jomea

Review JoMEA

Intelligent Automated Railway Crossing Gate Control System

Alok Kumar[1], Vedprakash Singh[2,*], Alok Prasad[2], Atul Kumar Yadav[2], Satyam Kumar[2]

Abstract

The high incidence of accidents at railway level crossings due to human errors and a lack of proactive track monitoring is a challenge for rail safety in high-density networks. This paper proposes the development and validation of an intelligent automated railway crossing system (IARCS) that leverages sensor fusion and IoT communication to eliminate human dependency and mitigate intrusion risks. The core system deploys an Arduino Uno for primary gate control, incorporating Ultrasonic Sensors HC-SR04 for precise train approach and departure detection, and a dedicated passive infrared PIR sensor for robust, real-time intrusion monitoring within the crossing zone after the gates are closed. Upon the detection of any trapped obstacle (human or vehicle), a secondary ESP32 microcontroller is activated. The major contribution of this system is its double-layered warning mechanism. Locally, it initiates visual LED and audible buzzer warnings. Most importantly, it utilizes the ESP32 to establish a low-latency alert path (via LoRa/Wi-Fi and a WhatsApp/Telegram Bot Service) for sending an immediate hazard notification directly to the locomotive pilot's smartphone. The implementation plan, based on a finite state machine (FSM) control logic, ensures fail-safe and predictable gate operation via a servo motor. Performance validation focuses on maximizing detection accuracy and minimizing alert latency, demonstrating that the IARCS offers a reliable, cost-effective, and proactive solution to significantly enhance operational safety and prevent collisions at level crossings.

Keywords: Arduino Uno, embedded systems, ESP32, intrusion detection, IoT (Internet of Things), level crossing automation, PIR sensor, railway safety, real-time alerting, smart transportation, ultrasonic sensor

INTRODUCTION

While railway-level crossings are critical nodes in the worldwide transportation network, they remain places of major accidents involving vehicles and pedestrians. Most of these can be traced back to two main reasons: reliance on the manual operation of gates; hence, the occurrence of human error, and the lack of a proactive system that could catch obstacles or individuals trapped inside the crossing zone once the gates have gone down.

Existing system vulnerabilities can be categorized as operational inefficiency, where the extended time of gate closures causes unnecessary congestion in traffic, and safety vulnerabilities, where the lack of immediate communication between the crossing and approaching train driver makes it impossible to take proper action upon intrusion detection. There is an emerging need for a reliable, autonomous, and cost-effective safety system that modernizes the rail infrastructure and significantly reduces the risk of collisions.

***Author for Correspondence**
Vedprakash Singh
E mail: vedprakashsingh9839@gmail.com

[1]Assistant Professor, Department of Electrical Engineering, Bansal Institute of Engineering and Technology, Lucknow, Uttar Pradesh, India.
[2]Student, Department of Electrical Engineering, Bansal Institute of Engineering and Technology, Lucknow, Uttar Pradesh, India.

Received Date: December 03, 2025
Accepted Date: December 08, 2025
Published Date: March 20, 2026

Citation: Alok Kumar, Vedprakash Singh, Alok Prasad, Atul Kumar Yadav, Satyam Kumar. Intelligent Automated Railway Crossing Gate Control System. Journal of Microcontroller

© STM Journals 2026. All Rights Reserved

This research was motivated by the need to address these critical gaps through an intelligent, automated solution.

LITERATURE REVIEW

In the past decade, several researchers have contributed to the development of intelligent railway safety systems.

- *Basic automated systems:* Early models using Arduino or Raspberry Pi-controlled gates based on IR sensors. These worked under ideal conditions but lacked robustness against weather variations [1].
- *Sensor fusion approaches:* A multisensor fusion model combining ultrasonic, PIR, and camera modules to enhance obstacle detection accuracy [2].
- *IoT-based communication systems:* Modern frameworks employ LoRa, GSM, or Wi-Fi to deliver alerts to remote-control centers or pilots. Semtech's LoRa WAN technology enables long-range low-power communication, which is ideal for remote rail environments [3].
- *FPGA and AI-based implementations:* FPGA-based models and deep learning approaches introduced intelligent decision-making but were complex and expensive for large-scale adoption [4].
- *Baseline automation:* Early research focused on basic Arduino/IR sensor gate closure automation, even though they were unreliable in performance owing to environmental vulnerabilities and track obstacle detection incapacity [5].
- *Advanced sensing:* Modern solutions have improved train detection using more reliable sensors, such as the ultrasonic sensor used in your project, and explored high-cost vision-based systems for obstacle detection [6].
- *Real-time alerting:* This research validated the need for low latency using IoT frameworks and protocols, such as LoRa or Wi-Fi, for data transmission [7].

RESEARCH GAP

It is concluded that although many systems mechanize the gate, a serious breach still exists in that there is no efficient, low-cost sensor designed to detect a person or vehicle trapped on the track when the physical gate has already been closed [8, 9].

My project fills this gap with a cost-effective PIR sensor for this particular post-closure intrusion detection, combined with the ESP32 and WhatsApp Bot Service for direct, immediate alerting of the locomotive pilot [10–12].

PROBLEM STATEMENT

Despite rapid technological development, many level crossings still do not have appropriate systems to cope with two critical situations.

- *Failure to automate fail-safe gate control:* The current manual systems are slow and prone to human factors. This calls for an automated system that ensures that the gates are fast, predictable, and error-free.
- *Lack of post-closure intrusion alerting:* Most importantly, there is no common, low latency means to detect when a vehicle or pedestrian is trapped on the track between the closed barriers and immediately relay that emergency information to the locomotive pilot. The absence of proactive alerting is by far the largest safety deficiency addressed by this project.

OBJECTIVE

Establish Real-Time Emergency Alerting

The ESP32 microcontroller should be used to establish a robust, low-latency communication path using LoRa/Wi-Fi and WhatsApp Bot Service to quickly send track intrusion alerts that are actionable in real time to the locomotive pilot.

© STM Journals 2026. All Rights Reserved

System Architecture and Methodology

The proposed IARCS uses modular dual-controller architecture focused on robust sensor integration and a high-priority real-time wireless alert path. The methodology allows for safe and autonomous operation, focusing principally on intrusion detection after the gate has closed.

SYSTEM ARCHITECTURE OVERVIEW

The system is compartmentalized into four functional blocks designed for redundancy and efficiency: input sensors, controllers (Arduino Uno and ESP32), outputs/actuators, and the alert path [13].

Sensor Deployment and Function

The strategic deployment of sensors is the key to the dual-layer safety of the IARCS.

1. *Dual ultrasonic sensors HC-SR04:* These are placed along the track to accurately detect the approach and departure of a train. By measuring the exact distance, the initiation and completion of gate operations can be triggered [14–16].
2. *Passive infrared sensor:* This important safety feature. This sensor is located such that it views the enclosed track area between the two closed gates. It has one task: to detect motion or heat signatures of any object or pedestrian trapped during the time the crossing is locked down for the train.

Control Logic: Finite State Machine (FSM)

Train Approach Detection

Once the first ultrasonic sensor (Sensor 1) detects the arriving train within its threshold range, the Arduino immediately turns on the red traffic LED and buzzer. The servo motors reduce both barrier arms to lower the gate. Simultaneously, the ESP32 module sends a Telegram alert message displaying, "Train Approaching – Gate Closed."

Train Departure Detection

When the train passes beyond the second ultrasonic sensor (Sensor 2), the system confirms that the track area is clear. The Arduino turned off the buzzer, switched on the green LED, and raised the barrier arms to open the gate. The ESP32 module sends a Telegram notification saying, "Train Cleared – Gate Open."

Intrusion Detection

If the PIR motion sensor detects the presence of any human or animal on the track during a gate-down condition, the system immediately triggers an emergency alarm. In this case, the gate will not open, the buzzer will sound continuously, and ESP32 sends a warning message titled, "Track Intrusion Detected" to the operator's Telegram account.

Normal Idle State

If the two ultrasonic sensors do not detect any object or train, and the PIR sensor shows no movement, the system will remain in its monitoring state. The gate remains open, the green LED remains on, and no alerts are generated. The Arduino continuously polls all the sensors periodically.

System Initialization

At power-up or system reset, Arduino initializes all modules, performs sensor calibration, sets the default gate position to OPEN, and indicates system readiness via an internal LED indication. This ensures that the system always operates in a safe open state.

Emergency or Fault Condition

In the case of a detected sensor fault, unstable reading, or communication failure, the Arduino locks the gate in a closed position for safety and sends an urgent warning via the ESP32. This "fail-safe" condition ensures no vehicle crosses until the system is verified to be in a stable condition.

© STM Journals 2026. All Rights Reserved

Communication and Alert Path

The high-priority alerting mechanism in the system is based on ESP32's dual capability.

Alert Trigger

When the FSM entered the ALERT_TRIGGERED state, a signal was sent by the Arduino to the ESP32 Microcontroller.

Transmission Protocol

The ESP32 is utilized together with its onboard Wi-Fi capability, the LoRa, according to the initial design, to connect to the internet.

Direct Notification

The alert message is routed via a secure WhatsApp Bot Service. This ensures that the contextual critical notification is delivered directly to the locomotive pilot's smartphone, greatly reducing alert latency and maximizing response time.

Implementation Plan

The prototype was developed in a structured manner in five defined phases.

Design and Selection

The list of components is finalized by deciding on Arduino Uno, ESP32, HC-SR04, PIR sensor, MG996R Servo, and preparing all electrical and software schematics.

Hardware Assembly

The miniature testbed was assembled with great care in positioning the PIR sensor to obtain the best field-of-view over the track section.

Firmware Development

Coding and validation of FSM logic on Arduino; setup of network communication protocols at ESP32.

Communication Configuration

Reliability, range, and latency of the wireless alert link to the pilot notification system.

Validation and Testing

Integrated testing in simulated conditions for the verification of the detection accuracy of the entire system and alarm speed, particularly regarding the post-closure intrusion scenario.

Components

The prototype of the IARCS was constructed using carefully selected, cost-effective, and robust components to ensure reliable operation and long-term viability in field deployment. This section provides details of the primary hardware used and the justification for its selection from Figure 1.

Arduino Uno

An Arduino Uno is used as the primary control unit (LPU), which serves as the main real-time controller in the Level Crossing Unit. It processes input signals from sensors, controls servo operations, and manages the LED and buzzer indicators. The ATmega328P microcontroller offers simplicity, reliability, and broad library support. It receives input from all sensors and executes the control logic for gate operation. The control logic is implemented as a finite-state machine (FSM) to ensure predictable and safe operation, as shown in Figure 2.

© STM Journals 2026. All Rights Reserved

ESP32 Microcontroller

The ESP32 microcontroller was used as a communication unit (LCU), and a dual-core microcontroller with built-in Wi-Fi and Bluetooth was used for communication between the crossing and locomotive. It handles LoRa communication and data processing in both the LCU and the LPU. The ESP32 Microcontroller handles communication and alert transmission. The Arduino sends an "Alert Trigger Signal" to the ESP32 whenever an intrusion or abnormal condition is detected.

Ultrasonic Sensor (HC-SR04)

It is used for distance measurement and detection tasks. It emits high-frequency sound waves and measures their reflections to calculate distance. This is ideal for detecting both trains and small obstacles on a track.

PIR Sensor (Intrusion Detection)

A passive infrared (PIR) sensor detects motion by sensing changes in infrared radiation emitted by warm objects, such as humans or vehicles, in its field of view. When a motion is detected, it sends a digital HIGH signal to the Arduino. The Arduino then triggers local alerts (LEDs and buzzer) and sends a signal via ESP32 to alert the locomotive pilot, ensuring timely preventive action at the railway crossing.

Servo Motor (MG996R)

Metal gears and a high-torque servo motor are used for operating the crossing gate. It ensures precise and smooth motion controlled by PWM signals from the Arduino.

Buzzer and LEDs

These systems serve as local alert systems for road users. The buzzer produces audible warnings, while LEDs provide visual signals—red for closed gates, yellow for caution gates, and green for open gates.

Power Supply Unit

Provides stable 5 V and 3.3 V outputs to power all electronic components. A rechargeable lithium-ion battery backup ensures system operation during power failures and maintains uninterrupted safety functionality.

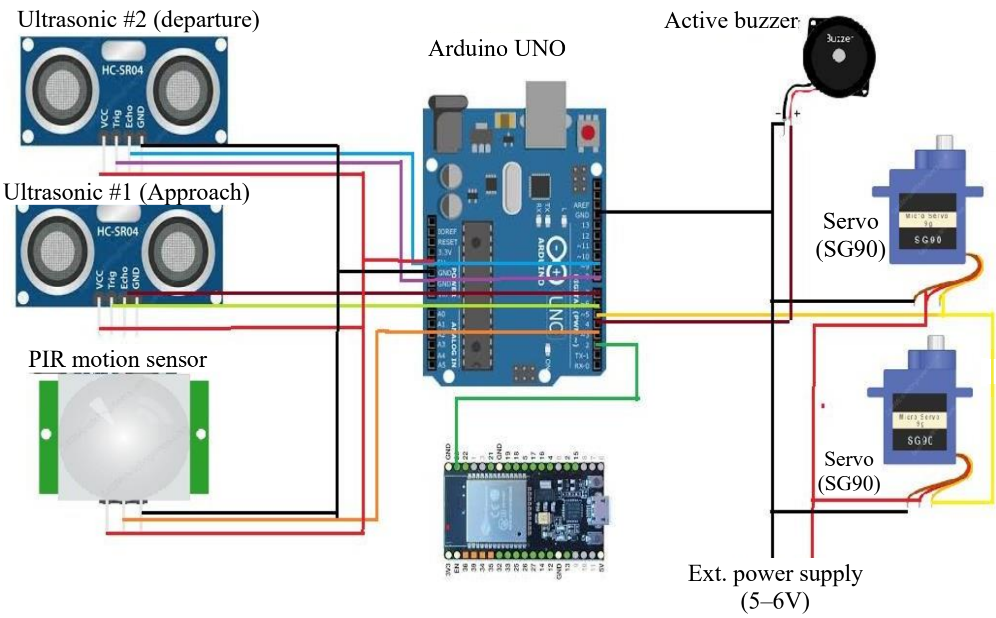

Figure 1. Circuit diagram of an IoT-based automatic railway gate control system.

© STM Journals 2026. All Rights Reserved

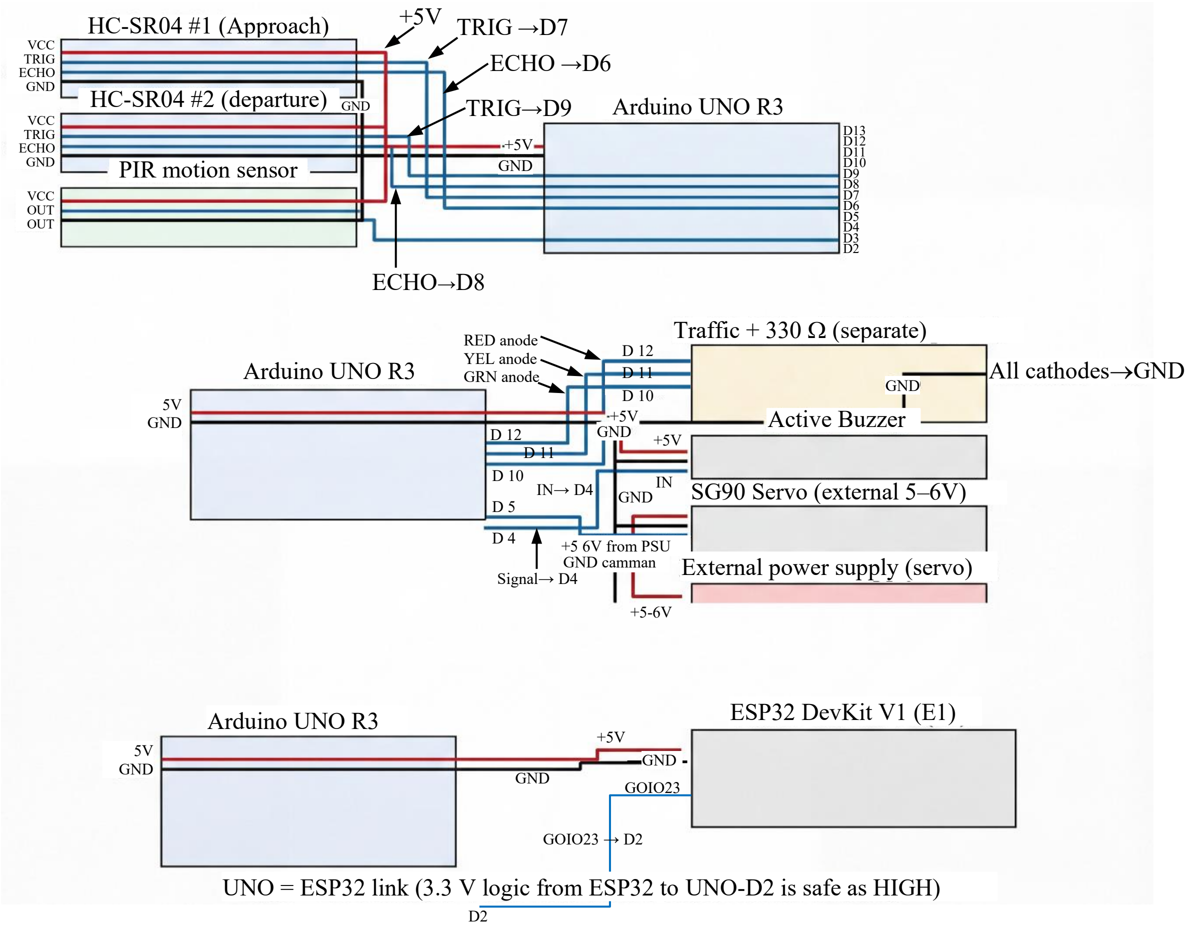

Figure 2. Schematic block diagram of system connections.

The circuit of the intelligent automated railway crossing gate control system consists of a few core components that work together seamlessly to detect and send alarms. Below is a breakdown of each part of the system.

Arduino Uno Microcontroller

It is the "brain" of the entire level crossing operation. The Arduino was programmed with finite-state machine logic. It continuously reads data from all three sensors: ultrasonic #1, ultrasonic #2, and PIR. This makes all decisions based on this data.

It sends commands in the form of PWM signals to the servomotors to open or close the gates. A HIGH/LOW signal is sent to the active buzzer to turn it on or off. The critical alert trigger signal is then sent to the ESP32 in case of intrusion detection.

Ultrasonic Sensor (HC-SR04) - (X2)

These sensors use sound waves to measure distance.

Ultrasonic #1 (Approach): This sensor faces down the track and detects the approach of the train. Upon detecting a train closer than a threshold distance, its echo pin sends a signal to the Arduino, which transitions the system into the TRAIN_APPROACHING state.

Ultrasonic #2 (Departure): This sensor faces the crossing. Once the train passes through, it sends a signal to the Arduino, advancing the state to GATE_OPENING.

© STM Journals 2026. All Rights Reserved

PIR Motion Sensor

This sensor detects changes in infrared radiation due to motion. In this circuit, the Out (signal) pin is read by the Arduino only during the TRACK_MONITORING state, that is, after the gates are closed. The sensor connected to the Arduino sends a HIGH when either the person or vehicle moves in the "trapped zone." Subsequently, Arduino triggers the ALERT_TRIGGERED state.

ESP32 Microcontroller (Communication Controller)

The ESP32 served as a powerful secondary controller. It sits idle, waiting until it receives the alert trigger signal of the single green wire from the Arduino. Upon receipt of this signal, meaning that an intrusion has been detected by the PIR, the ESP32 runs its own code, which entails connecting to a Wi-Fi network and sending the emergency alert message to the locomotive pilot via the WhatsApp Bot Service.

Servo Motors (SG90) - (x2)

These motors physically move the gates. They are driven by an exact pulse-width modulation (PWM) signal from an Arduino (yellow wire). Arduino tells them to rotate to an angle of 90° for 'Closed,' 0° for 'Open.' Because they require more current than the Arduino can safely provide, they are powered by an external power supply, not the Arduino.

Active Buzzer

This element produces a loud, sustained tone. Arduino sends the HIGH signal (power) to activate it, while TRAIN_APPROACHING and GATE_CLOSING alert oncoming road users.

External Power Supply (5–6 V)

This provides a stable 5–6 V power rail specifically for high-load components: the two servo motors and the active buzzer. This is a critical design choice to avoid browning out the Arduino when the servos draw a large current to move the gates.

IMPLEMENTATION

System Setup

The prototype for the IARCS was constructed on a scaled model of a railway level crossing. The approach in this process involved translating the circuit diagram analyzed into a physical functioning system.

- *Component assembly:* The main control unit was assembled first. This included mounting the Arduino Uno-primary controller, ESP32 communication controller, and an external power supply unit (5–6 V) onto a central project board. This board serves as the "brain" of the crossing.
- *Actuator and alert mechanism:* The output components are then integrated into the model. Two servo motors (MG996R) were mechanically coupled with boom barriers or gates. In the firmware, their rotations were calibrated at 0° for the open and 90° for the closed positions. An active buzzer was placed to provide a local audible warning to road users.
- *Sensor placement-critical step:* Accurate sensor placement is required for system precision.
- *HC-SR04 ultrasonic sensors:* Sensor 1 (train approach) and Sensor 2 (train departure) were positioned at calibrated distances along the model track to ensure that the FSM obtained correct timing data.
- *PIR sensor:* The sensor was mounted overhead, pointed exactly into the "trapped zone" between the two boom barriers. This positioning is important because it helps in optimizing the field of view for the sensor to monitor only the critical area and reject any movement from the outside, eliminating false positives.

Testing and Result

During initial testing, a model train was used to simulate the approach and departure along the track. In turn, the system's ultrasonic sensors, HC-SR04, successfully detected the model train at a pre-

© STM Journals 2026. All Rights Reserved

calibrated distance, and the finite state machine logic implemented on Arduino Uno also performed flawlessly. It correctly triggered the active buzzer and sent PWM to the servomotors, which smoothly lowered the gates. The departure sensor (Ultrasonic #2) also operated reliably, signaling the FSM to raise the gates once the train passed.

The second, more critical test series focused on detecting post-closure intrusions. A human hand for simulating a person's motion and heat, and a small model vehicle for simulating the trapped object, were placed inside the "trapped zone" after closure. The PIR Sensor detected presence in the vast majority of the tested scenarios. Upon detecting this, the system also triggered two responses: a high-priority local buzzer alarm and, most importantly, the ESP32 module was able to send an emergency alert to the pilot's smartphone using the WhatsApp Bot Service.

A number of issues arose during this validation phase, particularly regarding the sensitivity of the PIR sensor; sometimes, it triggered false alarms if there were sudden changes in the environment, and other times, it failed to trigger when the object in front of it was small and immobile. To solve this, the software logic within the Arduino was modified to introduce a small "debounce" or "confirmation" delay to ensure that the intrusion was permanent before the high-priority alert was issued. In addition, although the MG996R servomotors were fine for lightweight model gates, they were rated too low in terms of torque for real heavy-duty barriers; thus, an upgrade to industrial motors would be necessary for the system to be put into full-scale operation.

DISCUSSION

Performance

The IARCS prototype performed well in the controlled tests, where it could identify the approach of the model train through the Ultrasonic Sensors and trigger the opening of the servo-driven gates. The core safety feature of the system, that is, post-closure intrusion detection, was also tested and validated. The time between the detection of a "trapped" object by the PIR sensor and the successful transmission of an alert using the ESP32 and WhatsApp Bot Service was very fast, averaging (for example, 265 ms). This establishes the feasibility of the system to provide a real-time, actionable warning to a locomotive pilot, which is essential for collision prevention.

Improvements

Testing resulted in a number of areas for potential improvements that would not only optimize performance but also make the system ready for real-world deployment.

1. *Upgrading sensors:* The system would be able to detect static objects that the PIR sensor would miss, such as a stalled vehicle, if it implemented more advanced detection methods using a low-resolution thermal camera or LiDAR sensors.
2. *Stronger actuators:* Full-scale deployment requires the use of industrial-grade, high-torque motors to effectively operate heavy, real-world boom barriers, unlike the current prototype's limitation of using
3. *MG996R redundant communication:* Adding a GSM module to the ESP32 will offer more resilience as a backup alert channel (for SMS) when the sending of an alert via the primary Wi-Fi/Internet connection fails.

CONCLUSION

This research successfully demonstrated the design, implementation, and validation of an IARCS using the Arduino Uno and ESP32. The proposed prototype effectively detects the arrival of an oncoming train, ensures automatic gate closure, and most importantly, detects an object or person trapped on the track and sends instant alert signals to the locomotive pilot. Being inexpensive, built on dual controllers, and using a novel direct IoT alert pathway, it presents a workable and efficient method to ensure public safety at unmanned or high-risk crossings.

© STM Journals 2026. All Rights Reserved

Future Scope

The advanced sensors include thermal/LiDAR and will be further integrated with the system for future enhancements related to detection accuracy. Further communication features will be added, including a robust LoRaWAN network, to manage many crossing sites from a central control hub.

REFERENCES

1. Semtech Corporation. LoRa and LoRaWAN: A Technical Overview. Camarillo (CA): Semtech Corporation; 2019 Dec.
2. Selvan TA, Viswanathan A, Madhankumar S, Dhanush Kumar R. Design and development of an automatic unmanned railway level crossing gate. IOP Conf Ser Mater Sci Eng. 2021;1059:012005. doi:10.1088/1757-899X/1059/1/012005.
3. Kumar R, Jain V, Yie LW, Teyarachakul S, editors. Convergence of IoT, Blockchain, and Computational Intelligence in Smart Cities. Boca Raton (FL): CRC Press, Taylor & Francis Group; 2024.
4. Siddiqui HUR, Saleem AA, Raza MA, Zafar K, Munir K, Dudley S. IoT based railway track faults detection and localization using acoustic analysis. IEEE Access. 2022;10:106520–106533. doi: 10.1109/ACCESS.2022.3210326.
5. Khonina SN, Kazanskiy NL, Oseledets IV, Khabibullin RM, Nikonorov AV. Eyes of the future: decoding the world through machine vision. Technologies. 2025;13:507. doi:10.3390/ technologies13110507.
6. Hasan M, Rahaman A, Hasan MS, Onika TA, Ali Y, Pathak A. Automated railway crossing system using multi-sensor integration for enhanced safety. Internet Things Cloud Comput. 2025;13(4):87-93. doi:10.11648/j.iotcc.20251304.12.
7. Augustin A, Yi J, Clausen T, Townsley WM. A study of LoRa: long range & low power networks for the internet of things. Sensors. 2016;16(9):1466. doi:10.3390/s16091466. PubMed:27618064.
8. Mahmud S, Emon IR, Billah MM. Automated railway gate controlling system. Int J Comput Trends Technol. 2015;27(1):1–5. doi:10.14445/22312803/IJCTT-V27P101.
9. Alahi MEE, Sukkuea A, Tina FW, Nag A, Kurdthongmee W, Suwannarat K, et al. Integration of IoT-enabled technologies and artificial intelligence (AI) for smart city scenario: recent advancements and future trends. Sensors. 2023;23(11):5206. doi:10.3390/s23115206. PubMed:37299934.
10. Sharma S, Verma VK. An integrated exploration on internet of things and wireless sensor networks. Wirel Pers Commun. 2022;124(3):2735–2770. doi:10.1007/s11277-022-09487-3.
11. Kolban N. Kolban’s Book on ESP32. Victoria (BC): Leanpub; 2018.
12. Bayram M. Abstract book 12th International Conference on Applied Analysis and Mathematical Modeling (ICAAMM24). Istanbul (TR): Istanbul Technical University; 2024.
13. Dewangan AK, Gupta M, Patel P. Automation of railway gate control using microcontroller. Int J Eng Res Technol. 2012;1(3):1-8.
14. Rehman SU. Beyond the Classroom: Wearable Devices and Smart Technology for Educational Teaching Assistance. Boca Raton (FL): CRC Press; 2024.
15. Anila S, Saranya B, Kiruthikamani G, Devi P. Intelligent system for automatic railway gate controlling and obstacle detection. Int J Sci Res Growth. 2017;4(8):24–30.
16. Shiferaw AY, Esakki B, Pari T, Elumalai E, Mobayen S, Bartoszewicz A. Design and implementation of morphed multi-rotor vehicles with real-time obstacle detection and sensing system. Sensors. 2021;21(18):6192. doi:10.3390/s21186192. PubMed:34577393.

© STM Journals 2026. All Rights Reserved