Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2025) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

4th International Conference on Evolutionary Computing and Mobile Sustainable Networks

Smart Platform Connectivity Interface: Train detection and Distance Prediction Using IoT And Machine Learning

Muhammed Tufayl Dalvi*[a], Aditi Narkar[b], Dr. Sujata Kadu[c]

[a] *UG Scholar, Terna Engineering College, Nerul Navi Mumbai, Maharashtra 400706, India, Email: tufayldalvi007@gmail.com*
[b] *UG Scholar, Terna Engineering College, Nerul Navi Mumbai, Maharashtra 400706, India, Email: aditinarkar2004@gmail.com*
[c] *Assistant Professor, Terna Engineering College, Nerul Navi Mumbai, Maharashtra 400706, India, Email: sujatakadu@ternaengg.ac.in*

Abstract

This paper addresses the design of an intelligent IoT system interface that handles automated and motorized horizontal passenger transfer from one railway platform to another while at the same time increasing overall efficiency of the railway systems. The system employs NodeMCU as the main controller with a motorized rolling stage (interface) for motion control, various LEDs for displaying status and alarm signals and one screen display for displaying the incoming train information. Through computerised opening and closing of the platform edges, the passengers are able to have a smooth and secure passage between the platforms. The railway train detection is implemented in Python with the help of OpenCV and YOLOv5 object detection model. The design also accommodates wheelchair users by providing an accessible transfer platform, hereby enhancing mobility and safety for all passengers. To enable recognition of the train under different environmental conditions, a custom dataset of train images with annotations was developed specifically for training. When the approaching train is detected, the platform interface movement mechanism is activated and the information on the display together with the signal lights is changed.

Keywords: IoT in Railway Systems; OpenCV for Real-time Train Detection; Platform Crossing Interface; Distance Prediction Algorithm; Passenger Safety Automation.

1. Introduction

Railway transport is a fundamental part of combined world economies and is responsible for rendering mobility of both commodities and people over extensive distances. But, safety and operation efficiency of railway are the difficult issues which are hard to ensure with high traffic density and the demand for quick decision making. One of the most

* Corresponding author. Tel.: +91-845-287-9881.
E-mail address: tufayldalvi007@gmail.com

important factors which affect the railway station safety and efficiency is the design to control passengers transfer between the platform and the capability to deliver real-time information about the train arrival to the passengers for their efficient commute [1-2]. Some previous techniques used to manage the platform intersection are through using footbridges as well as tunnels or hand operations which may take time and are sometimes hazardous in crowded stations [6]. To overcome these difficulties, this paper proposes the design of the Smart Platform Interface that controls the movement of passengers between railway platforms using IoT technology and real-time train identification system for passenger automation [10-11]. Our system employs NodeMCU as the central processor which has incorporated a motorized platform interface, signalling LED and an LED display for train information and safety precautions in real-time [16]. The train detection mechanism uses advanced computer vision technologies, which include OpenCV and YOLOv5 which is a fast convolutional neural network (CNN) designed to detect objects in real time. This system proposes advances safety and operation of railway stations by automating the process of platform access based on real-time information and detection of trains [4-9]. In implementing the project, several specific steps are taken starting from generation and labelling of train image dataset with environmental conditions of interest. This dataset is employed to fine-tune of the YOLOv5 model in terms of train detection accuracy [4]. This real time response system makes it possible to open or close the special platform interface depending on the input detected train events and hence enhances the security of passengers and efficient station operations [13].

1.1. Use Case Diagram

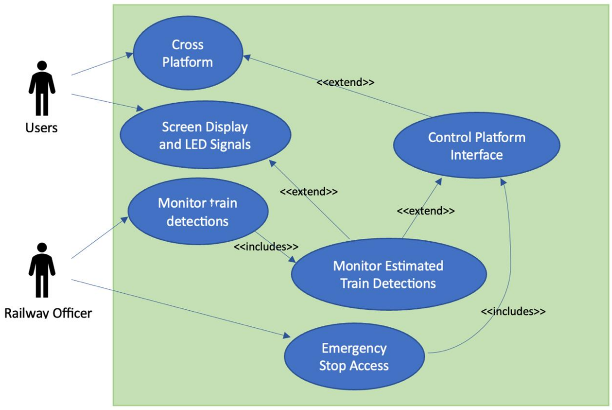

Fig. 1. Use Case Diagram

The use case diagram in fig 1 depicts the main functions of a smart platform-crossing interface system.

- Actors
  - *Users*: Represents passengers or individuals who want to cross the platform.
  - *Railway Officer*: A staff member responsible for monitoring and controlling platform access, especially in emergency situations.
- Use Cases
  - *Cross Platform*: Represents the action of passengers crossing from one platform to another. This action depends on whether the platform is safe and accessible.
  - *Screen Display and LED Signals*: Manages the visual signals (e.g., screen displays and LED indicators) that inform users about platform accessibility and the approach of any train. This feature likely communicates "safe to cross" or "do not cross" messages to users.
  - *Monitor Train Detections*: Detects the presence or absence of trains approaching the platform. This detection triggers safety mechanisms that may restrict access or update the signals accordingly. It is essential for real-time platform management and user safety.
  - *Control Platform Interface*: Manages platform access controls, likely opening or closing platform gates based on whether a train is detected. It acts as the central control for restricting or allowing passage.

  – *Emergency Stop Access*: Allows the *Railway Officer* to immediately stop platform access in emergencies, providing an extra safety layer.

2. Major Contribution

- *Enhanced Passenger Mobility and Safety*: The proposed system facilitates efficient passenger flow between platforms by automating crossing access, significantly reducing congestion and minimizing risks, especially during peak hours.
- *Cost-Effective and Scalable Solution*: Unlike traditional footbridges or tunnels, which are costly and time-consuming to build, this system provides an affordable and scalable solution that can be deployed at multiple railway stations without extensive infrastructure changes.
- *Accessibility for All Passengers*: With a focus on inclusivity, the system ensures smooth transitions for all passengers, including those with disabilities, by eliminating the need for stairs or ramps, which are often challenging for some passengers.
- *Real-Time, Low-Power Detection and Control*: Leveraging the low-power NodeMCU with the efficiency of YOLOv5 and OpenCV, the system offers real-time responses to train detection, operating seamlessly on a low-resource setup suitable for the constraints of railway environments.

3. Related Work

In the current Indian railway system, platform management primarily relies on manual operations and static infrastructure, with limited automation. The transfer of passengers between platforms is typically managed through footbridges, subways, and manually operated level crossings. Such methods are slow, and pose potential safety risks, especially during peak travel hours when stations are congested [12-13]. Additionally, real-time train arrival and departure information is often communicated through basic public announcement systems and manual displays, which are prone to human error and delays. This lack of automation and precision can lead to inefficiencies, such as longer wait times for passengers and a higher risk of accidents in the absence of controlled movement between platforms. These manual approaches are often insufficient to cope with the growing demand for safety and operational efficiency in densely populated urban railway stations[8-9]. Moreover, they are challenging for handicapped or elderly passengers to use bridges or stairs.

4. Literature Review

The integration of IoT technology in railway systems has significantly advanced safety, efficiency, and real-time monitoring. IoT-based systems are crucial for preventing railway accidents, as shown by Sagar et al. (2016) in their calamity avoidance system using cloud computing [1]. Similarly, Jo et al. (2018) highlighted IoT's potential in smart railways for predictive maintenance and large-scale monitoring [2]. Rail track health monitoring, as discussed by Chellaswamy et al. (2017), exemplifies how IoT can continuously monitor track conditions, enhancing track safety [3].

Combining IoT with machine learning and computer vision allows for precise, real-time responses, as demonstrated by Wu et al. (2021) with YOLOv5 for object detection [4]. Chandan et al. (2018) showcased deep learning for platform security, which is crucial for detecting incoming trains and anomalies [5]. IoT-based automated level crossing systems, like those by Amjath and Kartheeswaran (2020), underscore the importance of real-time detection to prevent accidents [7].

"Railway as a Thing," proposed by Oransa and Abdel-Azim (2015), suggests centralized IoT control systems for enhanced train scheduling accuracy, aligning with the requirements of automated platform interfaces [12]. Li et al. (2017) expanded on IoT's role in smart railways, emphasizing real-time diagnostics and monitoring [13]. Sustainable IoT applications, as discussed by Singh et al. (2022), focus on energy-efficient, safe railways, while Mansingh et al. (2015) illustrated machine learning's role in unmanned crossings, emphasizing the need for reliable object recognition [14, 17].

5. Block Diagram

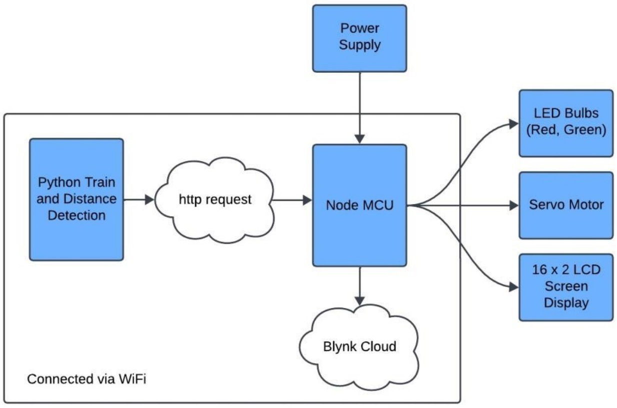

Fig. 2. Block Diagram

Figure 2 explains the working of the different blocks involved in the proposed smart platform interface. It includes:

- *Python Train and Distance Prediction System:* This refers to a toy train which is operated by a program written in Python language. This system involves the measurement of distance with a train and most probably an object or an untoward thing.
- *Node MCU*: This is a microcontroller board that is the main control centre of the entire distance prediction system. It gathers the information from the sensors and analyses the same.
- *Servo Motor*: This motor is also connected to the Node MCU and could be utilized to control the opening and closing of platform interface based on the predicted distance value of the train.
- *LED Bulbs (Red, Green)*: These are used to display the status as to the condition of the train or to measure distance. For instance, red light may mean that the train is near and interface has to be closed while a green light is used when the train is at a safe distance away from the platform.
- *16 x 2 LCD Screen*: This Liquid Crystal Display screen could be employed to display the measured distance or any other information that may be necessary.
- *Blynk Cloud*: This could be a cloud platform used for monitoring or controlling the train remotely through a Smartphone application or a web-based application. However, the Blynk Cloud is not essential for the core functionality of the system.
- *Power Supply*: This offers electrical energy required to energise the components of the system.
- *Wi-Fi*: This we believe signifies the presence of Wi-Fi link, maybe between the Node MCU and a computer or maybe an application on a smart phone for monitoring or controlling the train. However, Wi-Fi connectivity is not necessary for the basic operation of the system itself.

6. Implementation Details

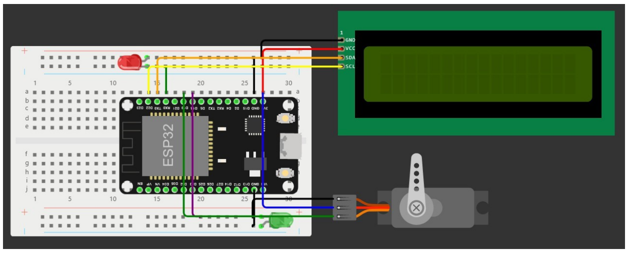

Fig. 3. Circuit Diagram

Figure 3 illustrates the use of NodeMCU, an ESP8266-based microcontroller, in controlling the platform interface system. The Circuit Diagram includes:

- LEDs (Red and Green): Include details of the availability status of the platform interface (Is it open or closed)?
- Servo Motor: Responsible for the movement of the platform from one side to the other to allow passengers transfer from one platform to another.
- LCD Display: Displays such things like time table of trains and other safely procedures to be followed.
- Camera Module: Records real-time footage of the platform, using either a laptop-integrated or camera module.
- Breadboard and Jumper Wires: Facilitate temporary and organized connections between the NodeMCU and components for easy circuit assembly.

6.1. Criteria for Model Selection

- Real-Time Processing for Safety: YOLOv5 and CNN models provide rapid inference times, ensuring the system can detect and react to an approaching train without delay, enhancing passenger safety.
- Compatibility with Low-Power Hardware: YOLOv5 and CNNs are efficient enough to be deployed on lower-power hardware setups while maintaining high accuracy, allowing the system to operate in resource-limited environments typical of public railway platforms.
- Adaptability and Scalability: The combination of YOLOv5 and CNN allows for easy model fine-tuning, letting the system adapt to different lighting conditions and platform environments, while OpenCV's extensive image processing tools provide flexibility in data handling and preprocessing.

6.2. Distance prediction algorithm

To achieve object detection with the use of YOLO, the class called YOLOPredictions was developed. The model was reloaded from ONNX file and labels from a yaml file containing the corresponding labels. The prediction function takes every frame or image and resize it to the input size of the model and then extract the object prediction from the YOLO model output. In order to remove redundant detections, Non-Maximum Suppression (NMS) algorithm was used by bounding the boxes with respect to confidence scores. The detected objects on the video frames had bounding boxes drawn around them, and the class labels gave the actual results together with the confidence scores attached to them.

Distance Estimation Formula

The distance to the detected object was calculated with the help of the reference image to find the focal length proportional to the real size of the face of the train and pixels width. The following formula was employed:

$$Focal\ Length = (Width\ in\ reference\ image \times Measured\ distance) / Real\ width$$

Once the focal length was determined, the distance to the object in each video frame was calculated using:

$$\textit{Distance} = (\textit{Real object width} \times \textit{Focal Length}) / \textit{Width in frame}$$

Automated Control in Accordance to Distance

NodeMCU microcontroller was used to open and close a gate with the help of servo motor according to the distance sensed. The distance was considered in real time from the video feed in the screen and a message was passed to the NodeMCU to power the servo according to the distance measure to the object (train).

6.3. Machine Learning Results

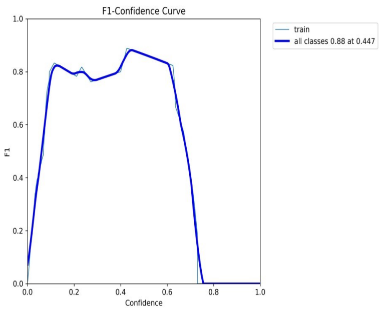

Fig. 4. F1 Confidence Curve

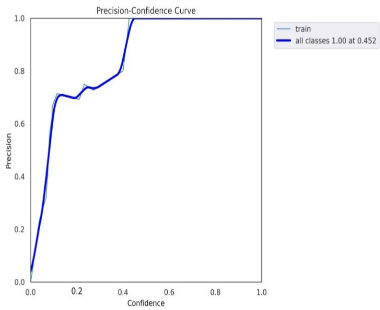

Fig. 5. Precision Curve

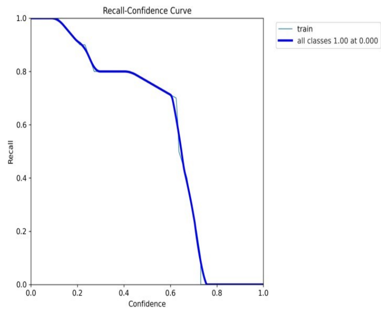

Fig. 6. Recall Curve

Figure 6 Shows how the F1 score (a balance of precision and recall) changes with confidence for the trained model. It peaks in the middle, suggesting optimal performance at moderate confidence levels.

Figure 7 Illustrates how precision (accuracy of positive predictions) improves as confidence increases for the trained model. The steep rise indicates higher precision with higher confidence.

Figure 8 Demonstrates how recall (ability to find all positive cases) decreases as confidence increases. This suggests a trade-off where higher confidence leads to missing more positive cases for the trained model.

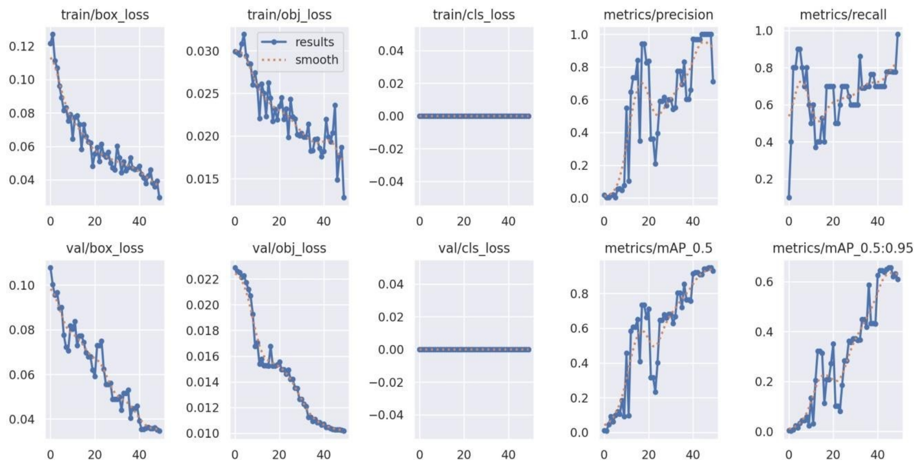

Fig. 7. Results

Figure 9 shows the performance of a machine learning model, for train detection:

- *Loss graphs* (top and bottom left 6) Shows decreasing losses for bounding box prediction and object detection in both training and validation sets, indicating improving performance.
- *Classification loss* (3rd column) is flat at zero, suggesting excellent class prediction.
- *Precision and Recall* (top right) are both generally increasing, showing better accuracy.

- *mAP scores* (bottom right) shows Increasing Mean Average Precision, indicating overall improvement in detection performance.

Overall, the graphs suggest the model is learning effectively across various metrics as training progresses.

Fig. 8. Train Batch

Figure 10 above is a sample of training data for an object detection model with an emphasis on detecting trains. The family has many snapshots of toy, or model trains inside probably on the floor or on a table. They are circumscribed with red rectangles to indicate the labelled objects which the model has to be trained on the trains. The pictures show trains with different orientations and from different perspectives as to their size, angles, depths, and lighting or background settings; this gives the model diverse examples of the object so that it can identify trains.

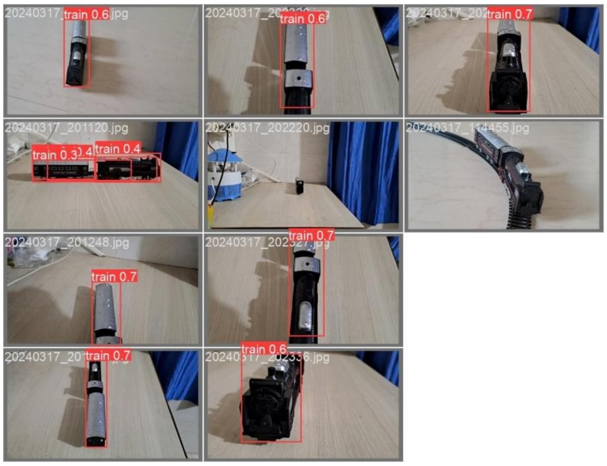

Fig. 9. Predictions/Test (Value Batch)

Figure 11 above is a sample of predictions from given object detection model to detect trains. Every prediction is given a value from 0 to 1, thereby pointing out the model's confidence in how effectively the trains will be identified in the given image. It is an important thing that higher values in such attributes contribute to greater confidence in the correctness of the prediction.

Each figure represents a frame captured and processed by the system, where trains are identified and tracked in real-time with bounding boxes. These visual outputs validate the model's effectiveness in distinguishing train objects from the background and other elements within the environment, while simultaneously estimating the distance of the detected train from the platform.

7. Demonstration

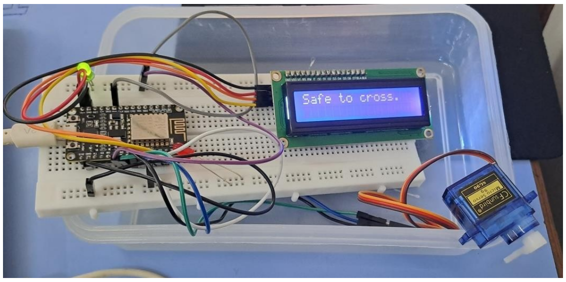

Fig. 10. Initial State

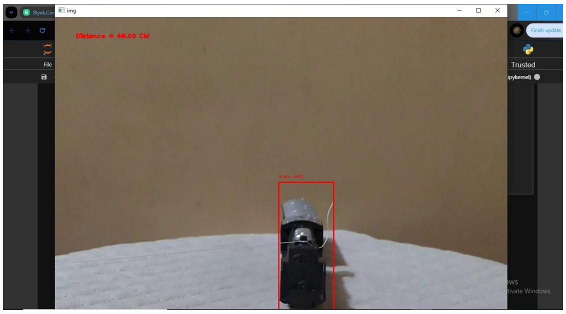

Fig. 11. Train Detection and its Distance Prediction

Figure 11 shows the initial state of the smart interface, with a green signal indicated by an LED bulb and a ”Safe to cross” message displayed on the LCD. The camera module is ready to detect the train and predict its distance.

Figure 12 shows the detection of the train, indicated by a red bounding box, and the prediction of the train’s distance.

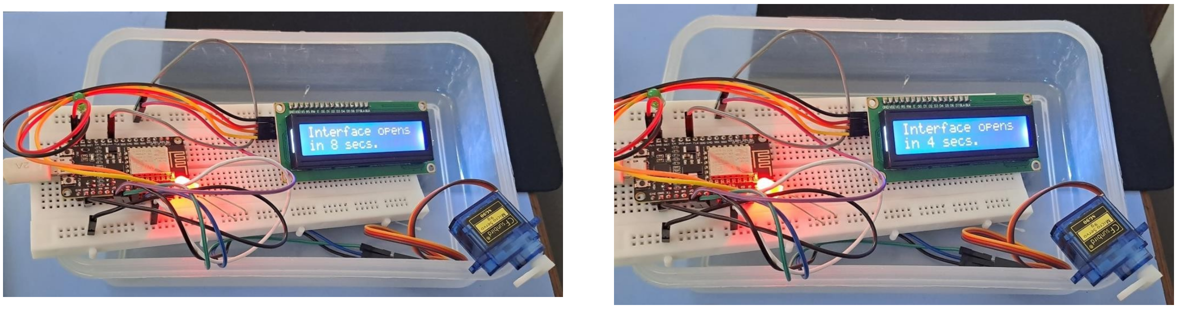

Fig. 12. Interface opening indication for passengers

Figures 14 show that when the detected train’s distance is less than 50 cm, the red signal is activated, indicated by the red LED bulb. During the red signal, passengers are not allowed to use the platform interface as the train is passing. The LED screen displays a 10-second countdown, after which the platform interface will reopen

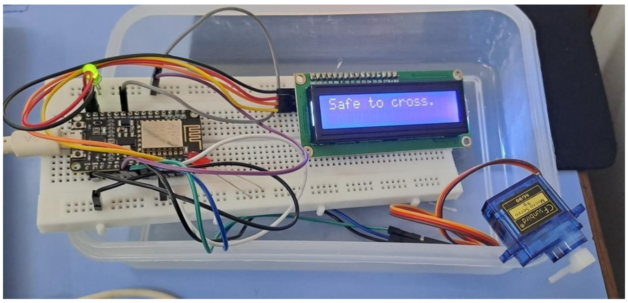

Fig. 13. Fifteen seconds after the train passes, the motor reopens the interface.

Figure 15 shows that 15 seconds after the train has passed, the motors reopen the platform interface, making it safe for passengers to use.

8. Major Factors Considered

- *Detection Accuracy*: Precise train detection is vital for safe platform management. YOLOv5 and OpenCV were chosen due to their strong performance in real-time object detection, which ensures the system can accurately distinguish trains from other objects or movements on the platform.
- *Real-time Processing*: The program requires rapid processing to detect trains as they approach, providing timely responses to open or close crossings. The NodeMCU microcontroller offers efficient connectivity and processing speed, allowing the system to work continuously with low latency.
- *Distance Estimation Precision*: Accurate distance prediction ensures that passengers can safely cross before the train arrives. Using camera-based detection and machine learning algorithms, the system can estimate the train's distance from the platform reliably, accounting for factors like train size and speed.
- *Environmental Adaptability*: The system will be designed to function in diverse lighting and weather conditions often encountered in railway environments. OpenCV's pre-processing features aid in adjusting for variations in lighting, while YOLOv5 can recognize trains across different backgrounds and conditions.

9. Limitations

- *Physical Interface Complexity and Folding Challenges*: The motorized rolling platform interface involves a complex folding and unfolding or rolling mechanism that must align precisely with the platform edges. Ensuring smooth operation of this mechanism under frequent use and varying loads (including passengers with assistive devices like wheelchairs) is a challenge. Furthermore, frequent folding and unfolding can lead to mechanical wear, alignment issues, or malfunctions, requiring regular calibration and maintenance. If the mechanism jams or fails to fold or roll properly, it could delay the platform's retraction, posing safety risks and operational interruptions.
- *Environmental Sensitivity*: The system's object detection model may struggle in extreme weather conditions such as heavy rain, fog, or snow, which could impair the camera's ability to detect approaching trains accurately. This may lead to delays or incorrect activation of the platform interface.
- *Integration with Legacy Systems*: Many railway systems still rely on older, analog technology. Integrating this IoT-based system with legacy railway infrastructure might be challenging, requiring modifications to both new and existing systems to ensure seamless operation.
- *Data Privacy and Security Risks*: As with any IoT system, there are concerns related to data privacy and cybersecurity. Unauthorized access to the system could lead to false activations or deactivations, posing risks to passenger safety and system integrity.

10. Conclusion and Future Work

Thus, the proposed solution is suitable for the analysed task and offers a viable solution to the problem of directing passenger traffic between platforms, thus quitting the need for using footbridges or underpasses. Thus, including NodeMCU for control, Python along with OpenCV for train detection, mechanisms of motor for the platform's movement, LED indicators for signalling, and an LED screen to display messages, the platform interface is both functional and user-friendly. If expanded and developed with cooperation the platform interface can be tailored towards the passenger's needs and the context, while keeping in mind the concept of accessibility for all. In conclusion, this solution can be claimed to be a step forward towards more effective railway station design.

In the future, it would be useful to expand the range of interfaces in the smart platform with additional types of sensors. The focus will be on implementing piezoelectric sensors to harness energy from the footsteps of passengers, thus contributing to energy efficiency. Furthermore, integrating solar sensors could further improve energy efficiency by utilizing solar power, thereby increasing the overall sustainability and performance of the platform interface.

References

[1] K. V. D. Sagar, K. V. D., Kumar, A. P., Ankush, G. S., Harika, T., Saranya, M., Hemanth, D. (2016). Implementation of IoT based Railway Calamity Avoidance System using Cloud Computing Technology. *Indian Journal of Science and Technology*, 9(17). doi:10.17485/ijst/2016/v9i17/93020.

[2] O. Jo, O., Kim, Y.-K., Kim, J. (2018). Internet of Things for Smart Railway: Feasibility and Applications. *IEEE Internet of Things Journal*, 5(2), 482-490. doi:10.1109/JIOT.2017.2749401.

[3] C. Chellaswamy, L. Balaji, A. Vanathi, and L. Saravanan, ”IoT based rail track health monitoring and information system,” in *2017 International Conference on Microelectronic Devices, Circuits and Systems (ICMDCS)*, Vellore, India, 2017, pp. 1-6, doi: 10.1109/ICMDCS.2017.8211548.

[4] T.-H. Wu, T.-W. Wang, and Y.-Q. Liu, ”Real-time vehicle and distance detection based on improved YOLO v5 network,” in *2021 3rd World Symposium on Artificial Intelligence (WSAI)*, Guangzhou, China, 2021, pp. 24-28, doi: 10.1109/WSAI51899.2021.9486316.

[5] G. Chandan, A. Jain, H. Jain, and Mohana, ”Real-time object detection and tracking using deep learning and OpenCV,” in *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2018, pp. 1305-1308, doi: 10.1109/ICIRCA.2018.8597266.

[6] Z. Guo, Z. Zhang, and W. Li, ”Establishment of intelligent identification management platform in railway logistics system by means of the Internet of Things,” *Procedia Engineering*, vol. 29, pp. 726-730, 2012, ISSN: 1877-7058.

[7] M. I. M. Amjath and T. Kartheeswaran, ”An automated railway level crossing system,” in *2020 International Conference on Image Processing and Robotics (ICIP)*, Negombo, Sri Lanka, 2020, pp. 1-7, doi: 10.1109/ICIP48927.2020.9367346.

[8] P. Sunitha, M. E. Krishna, D. Deepika, D. Divya, and K. Bhargav, ”Automated railway crossing system using IoT,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 07, no. 03, pp. 1255-1260, Mar. 2020. e-ISSN: 2395-0056, p-ISSN: 2395-0072.

[9] X. Zhang, X. Wang, and Y. Jia, ”The Visual Internet of Things System Based on Depth Camera,” in *Proceedings of 2013 Chinese Intelligent Automation Conference*, 2013, pp. 447-455.

[10] G. Singh, P. Kumar, R. K. Mishra, S. Sharma, and K. Singh, ”Security system for railway crossings using machine learning,” in *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, Greater Noida, India, 2020, pp. 135-139, doi: 10.1109/ICACCCN51052.2020.9362976.

[11] P. Krishna, ”Internet of Things (IoT) for railways,” *International Journal of Engineering and Advanced Technology*, vol. 8, pp. 738-739, 2019, doi: 10.35940/ijeat.F1136.0986S319.

[12] O. Oransa and M. Abdel-Azim, ”’Railway as a Thing’: New railway control system in Egypt using IoT,” in *2015 Science and Information Conference (SAI)*, London, UK, 2015, pp. 124-133, doi: 10.1109/SAI.2015.7237136.

[13] Q. Y. Li, Z. D. Zhong, M. Liu, and W. W. Fang, ”Smart Railway Based on the Internet of Things,” in *Intelligent Data-Centric Systems: Big Data Analytics for Sensor-Network Collected Intelligence*, H.-H. Hsu, C.-Y. Chang, and C.-H. Hsu, Eds. Academic Press, 2017, pp. 280-297. ISBN: 9780128093931.

[14] P. Singh, Z. Elmi, V. K. Meriga, J. Pasha, and M. A. Dulebenets, ”Internet of Things for sustainable railway transportation: Past, present, and future,” *Cleaner Logistics and Supply Chain*, vol. 4, 100065, 2022, ISSN: 2772-3909, doi: 10.1016/j.clscn.2022.100065.

[15] M. Kamble et al., ”IoT-controlled railway gate system with ML object detection approach: Applied approach for a secured IoT system,” in *Applied Approach to Privacy and Security for the Internet of Things*, P. Chatterjee et al., Eds. IGI Global, 2020, pp. 241-263, doi: 10.4018/978-1-7998-2444-2.ch012.

[16] E. Aboelela, W. Edberg, C. Papakonstantinou, and V. Vokkarane, ”Wireless sensor network based model for secure railway operations,” in *2006 25th IEEE International Performance, Computing, and Communications Conference (IPCCC)*, 2006, pp. 6, doi: 10.1109/PCCC.2006.1629440.

[17] B. B. Mansingh, K. Selvakumar, and S. V. Kumar, ”Automation in unmanned railway level crossing,” in *2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO)*, 2015, pp. 1-4, doi: 10.1109/ISCO.2015.7282293.

[18] M. Adil and M. Khurram Khan, ”Emerging IoT Applications in Sustainable Smart Cities for COVID-19: Network Security and Data Preservation Challenges with Future Directions,” *Sustainable Cities and Society*, vol. 75, 2021, 103311, ISSN: 2210-6707, doi: 10.1016/j.scs.2021.103311.

[19] L. Li, X. F. Wang, and X. Lei, ”The design of railway information system integration based on IoT,” *Advanced Materials Research*, vol. 694, pp. 3336-3339, 2013.

[20] G. Karaduman, M. Karakose, and E. Akin, ”Condition monitoring platform in railways based on IoT,” in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, Malatya, Turkey, 2018, pp. 1-4, doi: 10.1109/IDAP.2018.8620872.

[21] P. Fraga-Lamas, T. M. Fernández-Caramés, and L. Castedo, ”Towards the Internet of Smart Trains: A Review on Industrial IoT-Connected Railways,” *Sensors*, vol. 17, no. 6, 2017, p. 1457, doi: 10.3390/s17061457.

[22] É. Masson and M. Berbineau, *Broadband Wireless Communications for Railway Applications*, Springer International Publishing, 2017.