STM JOURNALS

Journal of
Control & Instrumentation

https://journals.stmjournals.com/joci

ISSN: 2229-6972 (Online)
ISSN: 2347-7237 (Print)
Volume 17, Issue 2, 2026
May–August
DOI (Journal): 10.37591/JoCI

Review JoCI

Automatic Car Controller Based on Sign Board Using Deep Learning and IoT

T.M. Dudhane Guide[1], Pravin R. Katta[2,*], Omkar V. Jagtap[3], Rushikesh V. Kale[4]

Abstract

The rapid growth of intelligent transportation systems has increased the demand for safer and more efficient driving solutions. Conventional vehicles often rely heavily on human intervention, which can lead to accidents due to negligence, fatigue, or poor visibility of traffic signs. This project proposes an automated car control system that utilizes deep learning and Internet of Things (IoT) technologies to recognize traffic signboards and respond accordingly. The primary objective is to enhance road safety by enabling vehicles to automatically detect traffic signs and adjust their behavior in real time. The system employs a MobileNet-based convolutional neural network (CNN) for traffic sign recognition due to its lightweight architecture and high efficiency in real-time applications. A dataset of traffic sign images is collected and preprocessed, followed by annotation and training of the model using TensorFlow. The trained model can identify multiple traffic signs under varying environmental conditions such as lighting, angle, and partial obstruction. Once a traffic sign is detected, the system generates appropriate control signals. These signals are transmitted to a microcontroller, which regulates the speed of the vehicle using a motor control mechanism. Additionally, an alert system notifies the driver through audio feedback, ensuring awareness even in semi-automated scenarios. This integration of computer vision and embedded systems enables real-time decision-making and reduces dependency on manual driving responses. The proposed system is cost-effective compared to high-end autonomous driving technologies, making it suitable for wider adoption, especially in developing regions. By combining deep learning with IoT-based control, the system not only improves traffic compliance but also reduces the likelihood of accidents caused by missed or ignored traffic signals. Overall, this approach demonstrates a scalable and practical solution for intelligent vehicle automation.

Keyword: Deep learning, traffic signal, mobile-net neural network, ESP 32, Blynk IoT

INTRODUCTION

The advent of artificial intelligence (AI) in recent years has led to an update in the vehicle-aided driving system's driving mode. The system prevents car accidents caused by driver weariness by rapidly reminding drivers to make accurate operations based on real-time road conditions [1]. The development of autonomous vehicles necessitates the quick and precise identification of traffic signs from digital imagery, in addition to auxiliary driving systems. Given a particular categorization, Traffic Sign Recognition (TSR) uses digital images or video frames to identify the location of traffic signs. In essence, TSR techniques leverage visual cues such as traffic sign color and design. However, in real-time tests, traditional TSR algorithms have limitations, such as being readily constrained by driving conditions, such as lighting, camera angle, blockage, and speed. Additionally, multi-target detection is exceedingly challenging, and visual items are easily missed because of the slow recognition.

***Author for Correspondence**
Pravin R. Katta
E-mail: pravinkatta18@gmail.com

[1]Assistant Professor, Department of Electronics and Telecommunication Engineering, Rajgad Dnyanpeeth's Rajgad Technical Campus, Dhangwadi, Bhor, Pune, India
[2–4]Students, Department of Electronics and Telecommunication Engineering, Rajgad Dnyanpeeth's Rajgad Technical Campus, Dhangwadi, Bhor, Pune, India

Receiving Date: April 16, 2026
Accepted Date: June 18, 2026
Published Date: June 25,2026

Citation: T.M. Dudhane Guide, Pravin R. Katta, Omkar V. Jagtap, Rushikesh V. Kale. Automatic Car Controller Based on Sign Board using Deep Learning and IOT. Journal of Control & Instrumentation. 2026; 17(2): 16–24p.

The limitations of artificial neural networks have been significantly reduced by ongoing advancements in computer technology, which have ushered in a golden age of machine learning development. Deep learning is a type of machine learning technique. When processing information, a deep neural network model mimics the neuronal architecture of the human brain. Compared with traditional TSR techniques, our neural network model performs significantly better at extracting useful information from road images, which could increase the applicability and robustness of the algorithms. In addition to preventing traffic accidents and safeguarding drivers, the results of the TSR study also aid in the correct and efficient inspection of traffic signs on roadways, which saves needless manpower and resources. Furthermore, it offers technical assistance for auxiliary and unmanned driving [2].

Consequently, deep learning research is extremely important and indispensable in our day-to-day existence. was founded on deep learning. Our contributions are classified into three categories. First, we created a new dataset of 2,182 photos with eight classes by gathering and enhancing sample photographs of traffic signs. Second, we applied our tests to the most recent version of Mobile-Net and assessed the TSR performance using our dataset. A few crucial references for additional research and exploitation are provided by the main metrics and parameters. Finally, we compared the TSR performances of SSD and Mobile-Net in detail [3]. Additionally, we examined and supported the benefits and drawbacks of these two deep learning methods [4]. In their description of system validation, a number of simulated datasets were used to evaluate the system under conditions of relative velocities and beginning separation distances of 100 m. The sensor reads the location of the vehicle from the object and the velocity of the object directly in front. The validation verified that the car was safely brought to rest by the automated brake control system. Technology continuously applies adequate and suitable braking pressure to stop the car in time in situations where halting the vehicle without colliding is feasible [5]. In other situations where an accident was physically inevitable, the control system still responded immediately and applied a lot of braking pressure, possibly lowering the vehicle's velocity upon impact and lessening the severity of the collision. Future Extent/Restrictions: Future research may design a multivariable system that takes longitudinal and transversal variables into account. Additionally, the weather factor may also be taken into account.

A novel design is created to regulate the cars' speed. The author should anticipate surrounding vehicles interfering and maybe blocking or attenuating radio frequency signals when driving normally. The author employed GPS locations for limited regions in this regard [6]. A project titled "automatic speed control and accident avoidance of vehicle using multi sensors" was presented, successfully constructed, and tested. It was created by combining the capabilities of each piece of gear [7]. The presence of each module has been thoughtfully considered and positioned to maximize the functionality of the unit. The system architecture guarantees driver safety. The automatic speed control system described in this study sets a speed restriction for the vehicle based on the distance to the front obstacle. This strategy is highly beneficial since it will lower the likelihood of traffic accidents and accident-related fatalities. The majority of traffic accidents are caused by careless or reckless driving, driver fatigue, and vehicle smoke [8]. Future Extent/Restrictions: This technique can be used for all types of road transportation. A variety of sensors can be used to measure different activities. By positioning a tiny camera to capture the driver's actions, the author used the open CV technique to make the eye blink sensor function. This makes it possible to determine the person's position [9].

LITERATURE SURVEY

The automated vehicle safety control in this study reduces the number of vehicle accidents by improving vehicle safety issues and making driving easier for the driver [5]. An Arduino microcontroller was used to implement the design. For training and voice recognition, an Easy VR is utilized. The issue based on the vehicle-to-vehicle space measure is the key action for vehicle speed control. Other preset control capabilities, including radio, flasher, and light controls, were incorporated into the design to maintain a proper road and position hub. The Easy VR module provides human voice instructions for operating all these control choices. Future Scope/Limitations: The proposed model was developed, constructed, and tested, with the expected results supporting the likelihood of additional future work with improved robotic operations for vehicle safety.

ISSN: 2229-6972 (Online), ISSN: 2347-7237 (Print)

The automatic beam controller is a straightforward and effective system that needs to be installed in all cars. It helps identify bright lights at night and automatically sends a signal to the car that is causing the glare, controlling the beam and making travel safer and easier for everyone [10]. In addition, we should all be well-versed in road safety, and authors should ensure that they adhere to all safety regulations and make things simple for other drivers and themselves. Life will be much easier and more seamless for everyone if the author upholds the rules of road safety and maintains discipline at all times. Aside from that, one of the most important factors in determining how uncomfortable the glare could be is the shape of the road. Road geometry and, consequently, vehicle location are intimately related to the impacts of the visual angle, lateral and longitudinal separations, and their effects on light intensity. The driver's health is a major concern; therefore, steps must be taken to reduce hazardous glare by using high-quality headlights and adopting a health tracking device. This problem may be readily resolved, and the author will seldom experience any accidents or tragedies if all the elements are integrated into a real-time working module and operated in accordance with the demands of the circumstance. Future Scope/Limitations: Data transfer may be accomplished using Bluetooth or wireless fidelity modules [11]. Accidents can be avoided by implementing an automatic reduction in the vehicle speed unit. Another option is to employ a driver's drowsiness detection equipment. It is necessary to rebuild headlights with filters that do not damage or irritate the eyes.

This system and concluded that it is a very useful automation system that saves operating time and manpower, hence increasing the system's economy [12]. Such automated solutions will be in greater demand in the future. The primary component used in this study to control the washing components of the system was the Mitsubishi (FX 2N-25MR) PLC. It is essential to understand the purpose of each PLC pin. PLCs are more capable than other integrated circuits in performing a wide range of tasks. Numerous programming languages are available for this PLC. The most practical programming language for PLCs is ladder diagram language. In this work, the PLC is programmed using a ladder diagram. The constructed system is a working PLC controller, which needs to be small, quick, and precise [13]. The features and dependability of the original designs may not be present in this system. It is merely being developed to ensure that the idea is workable, not unrealistic, and that it can be more effectively applied on a much greater scale. The coin or token system and the automatic washing system can both be utilized for interior cleaning [14]. Additionally, a counter that counts the number of cars washed can be implemented using this method. To address the issue of drivers failing to apply brakes promptly in an emergency, an automatic braking system model was developed using a fuzzy logic controller based on sensor fusion, which can automatically reduce the speed owing to obstacle detection [15]. The resulting system can accomplish high-accuracy measures, low stopping times, and better short-distance measurements by connecting ultrasonic sensors within the vehicle. The device works well in restricted places, emergency situations, heavy traffic, and parking situations. To prevent wheel locking, the current system uses a PID controller in the vehicle's electronic control unit. In contrast, the modeled system uses fuzzy logic to construct the controller, making it more stable and dependable than the current system and more likely to be accepted by customers. Table 3 indicates that, in comparison with the current PID controller, the proposed system has high values of brake deceleration, short stopping time, and high braking torque. Consequently, compared to the current system, the car might stop quickly. In contrast to the current PID system, the simulation results showed that the curve settled smoothly when decelerating and that the car did not shake under extreme braking circumstances.

The development and improvement of sophisticated computer programming, editing methods, diagnostic software, and algorithms for the dynamic interchange of information at various levels of the hierarchy require a strong multidisciplinary team with a solid engineering foundation [16]. Some of the issues can be resolved with simulation techniques. However, a test setup and robust quantitative model will aid in understanding these systems. However, all technologies, including robotics, are made up of tools and methods. How the designers, builders, and users of the tools and techniques comprehend and use them is crucial. Technology is undoubtedly vital, but success hinges on how it is applied and thought of without adding needless complications.

Auto car washers and dryers, the author can conclude that such automation systems are very advantageous, saving operating time and reducing the need for manpower, thereby boosting the system economy [17, 18]. Such systems will be in increasing demand in the future. Future Scope/Limitations: The system requires electricity for operation.

In this study, an overview of shared control in automated driving is provided, focusing on the theoretical understanding of the term and the examination of steering control system applications. The findings indicate a growing interest in this area and pertinent technological advancements, which encourage additional research and will probably soon result in trial testing in actual automobiles. Future Scope/Limitations: Actual cars and experimental research platforms will be used for testing in subsequent projects. Achieving strong driver acceptability requires minimizing conflicts at the steering wheel during the movements. Research on shared control algorithms that consider the driver state is a definite future direction in terms of application-oriented contributions. Shaheen et al. created a system that solely measures the vehicle speed. However, to avoid mishaps, the author frequently needs to regulate the speed before sending it to the monitoring authorities. Therefore, the author suggested an automatic vehicle speed system to address this issue, which not only recognizes but also regulates the speed of vehicles in accident-prone locations such as hospitals, schools, and temples. This system is made up of encoders that are positioned in various zones and emit various codes. The car's decoder picks up these codes and informs the microcontroller that the vehicle has reached the designated zone, which causes the microcontroller to lower the vehicle's speed. Several types of smart parking systems have been introduced. Its effectiveness in reducing traffic issues, particularly in urban areas where traffic congestion and a lack of parking places are evident, is demonstrated by the numerous instances of smart parking system deployment that are being given [19]. This is accomplished by guiding customers and making the best use of parking spots. The advantages and disadvantages of each sensor technology can be examined by studying all the sensor technologies used in vehicle detection, which is one of the most important components of a smart parking system. As previously mentioned, there are certain drawbacks to using a visual-based system for vehicle detection, but they are greatly outweighed by their benefits. Future Scope/Limitations: Smart Car Recognition: Using the ACP system's image processing capabilities, we can identify cars by their license plates. This type of technology allows users to pay for parking immediately using the balance in their auto-parking account or prepaid balance on their cell phone. Notifying Users of Available Slots and Account Balance By sending a brief SMS to the database, users can receive information regarding available slots for a specific parking space as well as their account balance.

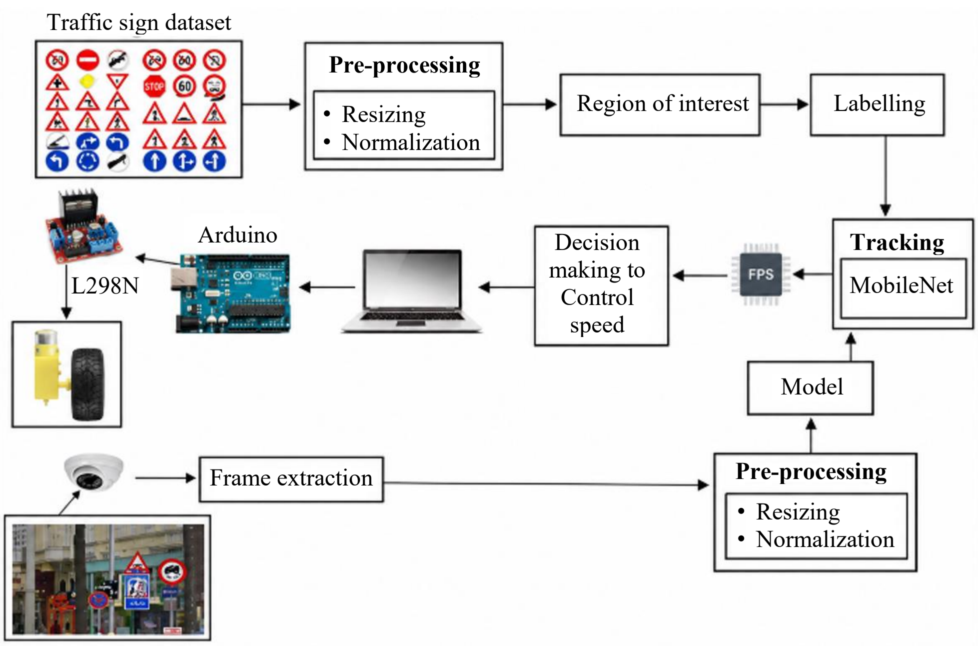

Figure 1. Block diagram of traffic sign detection and speed control system.

Developing a zone-based speed controller using GPS technology may be a practical strategy for reducing traffic accidents caused by excessive speeding [20]. Technology determines the vehicle's position and relevant speed limit based on its GPS coordinates. This data was then used to automatically modify the vehicle's speed. A zone-based speed controller using GPS is an interesting idea that might greatly increase traffic safety. With further study and development, this gadget may ultimately become a standard feature in automobiles, increasing safety and reducing the likelihood of accidents.

METHODOLOGY

Figure 1 shows a block diagram of the traffic sign detection and speed control system. The method that has been suggested to establish an Automatic car controller based on sign board system depicted in the system overview in Figure 1 up top.

The suggested method was based, in part, on the execution of the procedures detailed below.

1. *Step 1:* Dataset Generator: The first step of the process involved using OpenCV to capture images of various traffic signboards on the route. To detect the intended automobile controller during model testing, the CV2 package's Video Capture technique took pictures of it. All signboard photos were stored in the dataset folder. The model works on models such as sign boards and automobiles. The gathered photos were stored in a folder for use in the subsequent stages of the model.
2. *Step 2:* Data Labelling: In this stage, labeling software is used to label the photos that were taken in the previous step. The user must input the image to be labeled into the labeling program to mark the coordinates in the upper right corner and the lower bottom to create the rectangle's x1, y1, and x2, y2 for the lower right coordinates. To train the model using a deep learning network called MobileNet, the acquired coordinates were saved in an XML file.
3. *Step 3:* API installation: The first step was to install the TensorFlow Object Detection API on the Google Colab instance. Claiming the [TensorFlow models repository] (https://github.com/tensorflow/models) and executing a few installation scripts are necessary for this. To execute the subsequent code segments, click the play button. Subsequently, to download and extract the cuDNN files, we must install the Conda environment. Cloning the TensorFlow Models repository from GitHub comes next. The next step was to install the Object Detection API.
4. *Step 4:* Prepare training data and upload image datasets: To prepare the training data for TensorFlow, we uploaded our training images and utilized TF Record generating routines. Before running scripts to create TF Records from our data, we uploaded our photographs and divided them into training, validation, and test folders. First, we created a single folder named" images.zip" on our local PC that contained all of our training images and XML files. The zip folder should include the files directly. We must execute a few commands to unzip it and configure the image directories after it has been uploaded. The /content folder in the file system of this instance is where these folders are formed. The "Files" icon on the left allows us to browse the file system.
5. *Step 5:* The photos were divided into folders for testing, validation, and training. You can view our" images.zip" file in the list of files in this section by clicking the folder icon on the left. After uploading the dataset, we unzipped it and created folders to store the pictures. The /content folder in the file system of this instance is where these folders are formed. By selecting the "Files" icon on the left, we can navigate the file system. The photos were then divided into test, validation, and training sets. Each set is used for the following purposes: Train: The model is trained using these real photos. A batch of photos from the "train" set was fed into the neural network at each stage of training. The network predicts the locations and types of items in the images. Through backpropagation, the training method modifies the network weights and computes the loss. Validation: The training algorithm can use images from the "validation" set to monitor the training progress and modify hyperparameters (such as the learning rate). These images are only used sporadically during training, as opposed to "train" images, which are used once a predetermined number of training steps.

6. *Step 6:* TF Record Generation: The labeled data are converted into TF Record format, which is required by TensorFlow for efficient processing. A label map file was also created to define different classes, such as traffic signs.
7. *Step 7:* Model Configuration and Training: A pre-trained SSD MobileNet model is selected and customized for the project. Configuration parameters, such as the batch size, number of training steps, and file paths, were updated. The model is then trained using the prepared dataset. Checkpoints are saved periodically to monitor progress and improve performance.

IMPLEMENTATION

The proposed system implements deep learning methods with embedded hardware components. The researchers gather traffic sign images from streets by using a camera module and OpenCV image processing tools. Researchers use annotation software to label images because it allows them to define object boundaries and corresponding classes. The labeled dataset was then preprocessed and converted into TF Record format to make it suitable for training with the TensorFlow Object Detection API. Researchers chose a MobileNet-based convolutional neural network because its lightweight design enables efficient processing of real-time tasks. The dataset is divided into training, validation, and testing subsets to ensure proper evaluation and avoid overfitting. The research team established training parameters, including batch size, learning rate, and number of steps, to achieve the best model performance. The model was trained using Google Colab with GPU support to accelerate the training process. The system uses a live camera feed to detect traffic signs after the model is trained. The system processes each frame to identify traffic signs, which are then classified into predefined categories. The system converts the detected output into control signals for further processing. The detection system was connected to the vehicle prototype using an Arduino microcontroller. The microcontroller changes the vehicle motor speed for detected traffic signs using a servo motor system. The system automatically activates vehicle speed reduction when it detects a speed-limit sign. The system includes an audio alert module that notifies the driver in real time. This complete pipeline ensures seamless interaction between the software and hardware components.

Automatic Mode

The system uses a camera fixed on the vehicle to capture instant street images. The system operates through continuous image frame processing to identify traffic signs without human assistance. The system uses a MobileNet deep learning model that has already been trained to automatically detect and classify traffic signs from the collected visual data. The system processes every frame in real time to identify various signboards through its acquired knowledge about their shape and color characteristics. The system creates control commands that correspond to the identified traffic sign after the system successfully identifies the traffic sign. The system maps the identified sign to established response procedures, which include reducing speed, issuing warning alerts, and executing stop commands. The system sends commands to the microcontroller, which operates without requiring manual control. The Arduino microcontroller interprets the incoming signals to operate the motors based on the received commands. The system uses a servo motor system to automatically adjust vehicle speed according to the identified traffic sign. The system uses an audio alert system to automatically notify the driver of the detected sign. The system maintains environmental observation while making ongoing decision updates throughout its operation. The system uses immediate processing methods that enable the vehicle to react without delay to modifications in street traffic patterns. The system uses automatic control methods that decrease the need for drivers to react and reduce human errors. The system functions through a closed-loop system that continuously performs both detection and operational processes. IoT integration creates seamless communication pathways between software systems and their associated hardware. The system maintains operational performance throughout complex traffic situations owing to its automatic traffic control features. The system requires only a small amount of human control to maintain its complete operational capacity. The system achieves better driving safety and operational reliability through its automatic decision-making capabilities.

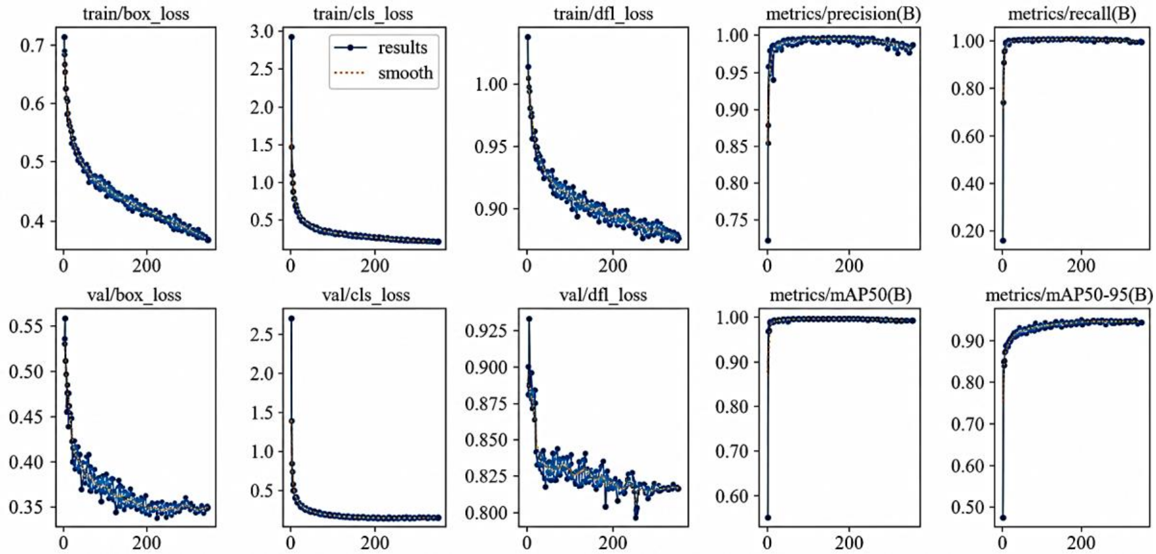

Figure 2. YOLO model performance during training and validation.

RESULTS AND DISCUSSION

Figure 2 shows the performance of the YOLO model during training and validation. The trained object detection model demonstrated effective performance in identifying traffic signs under varying conditions. The output results show that the system successfully detected and classified multiple instances of road signs, especially zebra crossing and road hump signs, which received high confidence scores. Most detections achieved confidence values ranging between 0.9 and 1.0, indicating strong prediction reliability.

The bounding boxes generated by the model accurately localized the objects within the images because the model could track signs that appeared from different angles, distances, and lighting conditions. The model was able to maintain consistency across multiple frames because it correctly identified the same class of objects, which showed only slight variations in orientation and background.

The zebra crossing sign was detected with near-perfect confidence (1.0) in multiple test samples, which showed the model's ability to recognize commonly repeated patterns. The model correctly identified the uneven road and road-hump signs because it could distinguish between visually similar categories.

The results further indicate that the model performs well in controlled indoor environments with minimal background noise. The system performance variance occurs because confidence changes at 0.9 and above display different performance results depending on the angle, occlusion, and image clarity.

The system achieved accuracy in detection and classification, which remained consistent throughout all test cases. The proposed approach demonstrates effectiveness in real-time traffic sign detection, which extends to driver assistance systems and smart transportation solutions.

CONCLUSION AND FUTURE SCOPE

This project developed a system that allows a camera mounted on an automobile to take pictures and recognize different traffic signs. We accomplish this by training a deep learning model that detects different traffic symbols from the designated times using the MobileNet neural network. We employ the deep learning model Mobile Net, which correctly recognizes the object of the traffic symbol and converts it into a label using the camera frames that were taken. A dictionary scrutinizes the received label to establish the name of the traffic symbol. The name of this traffic symbol is translated to text.

The system converts this traffic symbol name in following the audio alarm, and the microcontroller connected to the motor is triggered by delivering a signal to slow down the speed of the wheels. A single tire, Arduino Uno microcontroller, and servo motor were used to represent the entire model, and they functioned well in the scenario. Step 6: TF Record Generation The labeled data were converted into the TF Record format, which is required by TensorFlow for efficient processing. A label map file was also created to define different classes, such as traffic signs. Step 7: Model Configuration and Training A pre-trained SSD MobileNet model is selected and customized for the project. Configuration parameters, such as the batch size, number of training steps, and file paths, were updated. The model is then trained using the prepared dataset. Checkpoints are saved periodically to monitor progress and improve performance.

The proposed system implements deep learning methods with embedded hardware components. The researchers gathered traffic sign images from streets using a camera module and OpenCV image processing tools. Researchers use annotation software to label images because it allows them to define object boundaries and corresponding classes. The labeled dataset was then preprocessed and converted into the TF Record format to make it suitable for training with the TensorFlow Object Detection API. Researchers chose a MobileNet-based convolutional neural network because its lightweight design enables efficient processing of real-time tasks. The dataset is divided into training, validation, and testing subsets to ensure proper evaluation and avoid overfitting. The research team established training parameters, which included batch size, learning rate, and number of steps, to achieve the best model performance. The model was trained on Google Colab using GPU support to accelerate the training process. The system uses a live camera feed to detect traffic signs after the model is trained. The system processes each frame to identify traffic signs which it then classifies into predefined categories. The system converts the detected output into control signals for further processing. The detection system was connected to the vehicle prototype using an Arduino microcontroller. The microcontroller changes vehicle motor speed for detected traffic signs by using a servo motor system. The system automatically activates vehicle speed reduction when it detects a speed-limit sign. The system includes an audio alert module that notifies the driver in real time. This complete pipeline ensures a seamless interaction between the software and hardware components.

REFERENCES

1. Kukreti R, Pal R, Dimri P, Koli S, Joshi K. Automatic speed control of vehicles in speed limit zones using IR sensor. In: International Conference on Emerging Trends in Expert Applications & Security; 2023 Feb 17; Singapore. Springer Nature Singapore; 2023. p. 413-22.
2. Jin Y, Fu Y, Wang W, Guo J, Ren C, Xiang X. Multi-feature fusion and enhancement single shot detector for traffic sign recognition. IEEE Access. 2020;8:38931-40.
3. Ellahyani A, El Ansari M, Lahmyed R, Trémeau A. Traffic sign recognition method for intelligent vehicles. J Opt Soc Am A Opt Image Sci Vis. 2018;35(11):1907-14.
4. Bamigboye OO, Obaje SE. Intelligent automatic car braking control system using neural network classifier. Int J Eng Invent. 2016;5:51-6.
5. Alika R, Mellouli EM, Tissir EH. Adaptive cruise control of the autonomous vehicle based on sliding mode controller using Arduino and ultrasonic sensor. J Robot Control. 2024;5(1):301-11.
6. Prakash SA, Mohan RA, Warrier RM, Krishna RA, Bhaskar AS, Nair AK. Real time automatic speed control unit for vehicles. In: Proceedings of the 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC); 2018 Aug 30-31; Palladam, India. Piscataway (NJ): IEEE; 2018. p. 157-161. doi:10.1109/I-SMAC.2018.8653721.
7. Bhumkar SP, Deotare VV, Babar RV. Accident avoidance and detection on highways. Int J Eng Trends Technol. 2012;3(2):247-52.
8. Bangquan X, Xiong WX. Real-time embedded traffic sign recognition using efficient convolutional neural network. IEEE Access. 2019;7:53330-46.
9. Hao G, Yingkun Y, Yi Q. General target detection method based on improved SSD. In: Proceedings of the 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC); 2019 May 24-26; Chongqing, China. Piscataway (NJ): IEEE; 2019. p. 1787-1791. doi:10.1109/ITAIC.2019.8785733.

10. Roy S, Bagubali A, Agrawal R. Automatic vehicle beam controller. In: Proceedings of the 2019 Innovations in Power and Advanced Computing Technologies (i-PACT); 2019 Mar 22-23; Vellore, India. Piscataway (NJ): IEEE; 2019. Vol. 1. p. 1-6. doi:10.1109/i-PACT44901.2019.8960141.
11. Kuznetsova A, Maleva T, Soloviev V. Detecting apples in orchards using YOLOv3 and YOLOv5 in general and close-up images. In: International Symposium on Neural Networks; 2020 Nov 27; Cham. Springer International Publishing; 2020. p. 233-43.
12. Oo MT, Min HY, Oo NN. Programmable logic controller based automatic car washing system. Iconic Res Eng J. 2019;2(11):1-7.
13. Lian J, Yin Y, Li L, Wang Z, Zhou Y. Small object detection in traffic scenes based on attention feature fusion. Sensors (Basel). 2021;21(9):3031.
14. Zhang J, Hui L, Lu J, Zhu Y. Attention-based neural network for traffic sign detection. In: Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR); 2018 Aug 20-24; Beijing, China. Piscataway (NJ): IEEE; 2018. p. 1839-1844. doi:10.1109/ICPR.2018.8546289.
15. Bassey EF, Udofia KM. Modelling of automatic car braking system using fuzzy logic controller. Niger J Technol. 2019;38(4):1021-9.
16. Naranjo JE, González C, García R, de Pedro T, Haber RE. Power-steering control architecture for automatic driving. IEEE Trans Intell Transp Syst. 2005;6(4):406-15.
17. Nitin SS, Raghunath GP, Vinayak DR. Automatic car washing and drying using microcontroller 8051. Int J Trend Sci Res Dev. 2019;3(3):1544-6.
18. Marcano M, Díaz S, Pérez J, Irigoyen E. A review of shared control for automated vehicles: theory and applications. IEEE Trans Hum Mach Syst. 2020;50(6):475-91.
19. Elfaki AO, Messoudi W, Bushnag A, Abuzneid S, Alhmiedat T. A smart real-time parking control and monitoring system. Sensors (Basel). 2023;23(24):9741.
20. Sathisha G, Subbaraya CK, Ravikumar GK. Automatic speed control in vehicle based on geolocation information. In: Proceedings of the 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS); 2024 Apr 18-19; Bengaluru, India. Piscataway (NJ): IEEE; 2024. Vol. 1. p. 1-7. doi:10.1109/ICKECS61492.2024.10617025.

