International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 12, ISSN No. 2455-2143, Pages 103-107
Published Online April 2023 in IJEAST (http://www.ijeast.com)

REAL-TIME TRAFFIC SIGN DETECTION AND MAPPING FOR DRIVER ASSISTANCE USING CNN AND OPEN-SOURCE LIBRARIES

Utkarsh Kumar Singh, Vaibhav Mishra, Radhey Shyam, and Atul Kumar
Department of Information Technology,
Shri Ramswaroop Memorial College of Engineering and Management,
Lucknow, India

Abstract: **The rising number of accidents caused by failure to obey traffic laws and road signs can be addressed through a traffic sign detection system and its mapping for road sign navigation. This paper addresses a solution that utilizes generated training data made from photos and location of road traffic signs. Our system detects and notifies drivers of upcoming traffic signs during navigation, even if they are obscured by weather or other factors. The system assures fewer accidents and allows drivers to focus on driving rather than reading every single traffic sign on road and achieves the accuracy up to 99.27%.**

Keywords: **Traffic Sign Detection, CNN, Deep Learning, Navigation, Image Processing, Object Detection, Color-based Segmentation, Shape-based features**

I. INTRODUCTION

Traffic sign recognition is a vital area of study in computer vision and machine learning. With an increasing number of vehicles on the road, it is critical to have a dependable system capable of accurately detecting and recognizing traffic signs. This project proposes a solution that employs a Convolutional Neural Network (CNN) model to detect traffic signs via image upload or live camera feed [12][13]. Road signs are not expected to be noticeable to drivers; there may be little confusion between signs, or some signs may be un-recognizable to a small number of drivers or beginners. Traffic sign recognition systems detect, classify, and interpret traffic signs using computer vision and machine learning techniques. Deep learning algorithms have yielded good results and performance on several benchmark datasets in recent years, indicating significant progress in these systems. However, challenges remain, particularly when dealing with complex scenarios such as geometric distortion, changing light levels, and weather conditions. This may cause confusion and increase the likelihood of an occurrence of a hazardous event. It is dangerous if there is confusion in identifying warning signs.

We are developing a system that not only detects signs but also marks their locations on a map. The marked signs serve as way points for the user to follow, allowing the user to navigate more efficiently. Furthermore, the system warns the user about upcoming signs on the user's route, removing the need for camera detection in the future. The integration of traffic sign recognition technology with navigation systems provides users with a number of advantages. The system is intended to provide a warning for upcoming signs on the user's route, removing the need for users to rely solely on camera detection to stay informed of traffic signs. This not only improves driving safety but also reduces time spent looking for signs, allowing drivers to focus on the road ahead.

In this experiment, the authors have used a training data set that contains around 35,000 images that are very similar to real-world datasets. The intension behind choosing this data set is to how we can improve our detection precision that should be greater than 90 % to 95% of the true observation.

II. RELATED WORKS

From the recent past decades, plenty of research articles have been published in various reputed journals in this domain. In [1][11], the researchers have proposed a traffic sign recognition system that improves recognition accuracy by combining transfer learning. These systems achieved better recognition accuracy when tested on a common dataset i.e., German traffic signs.

In [2], the authors have proposed a deep learning approach for the detection and recognition of traffic sign in real-time. This approach combines two popular methods: (i) Faster R-CNN, and (ii) SSD algorithms. The system achieved good recognition accuracy when tested on a large dataset which contains real-world traffic sign images.

In [3], the authors have proposed a real-time traffic sign recognition system that combines two different approaches: the first one is the color-based segmentation technique, and the second one is the shape-based recognition technique. The system performed well when tested on a large dataset of real-world traffic sign images.

III. LITERATURE REVIEW

Numerous strategies for traffic sign detection and recognition have been proposed over the years, including traditional image processing techniques and deep learning-based approaches. Rule-based algorithms are frequently used in traditional traffic sign detection and recognition techniques. In [4], authors have proposed a method for traffic sign detection that used color and shape features. In [5], the authors have used a combination of shape and texture features. But this method does not perform well variable lighting conditions.

1.1. **Color-based image segmentation**:
Different colors, such as red, yellow, and blue, are frequently used on traffic signs. Based on color information, color-based segmentation can be used to separate the sign region from the background. While color-based segmentation can be useful for traffic sign detection. Although it is good but it suffers from lighting conditions, color variation, and color ambiguity.

1.2. **Shape-based characteristics**:
Traffic signs frequently have distinguishing shapes such as circles, triangles, and rectangles. Shape-based features can be used to detect and distinguish the sign region from other objects in the scene. While color-based segmentation can be useful for traffic sign detection, it has some limitations, including sensitivity to scale and rotation, limitation to predefined shapes, and inability to handle obstruction.

1.3. **Edge Detection**:
Edge detection algorithms can be used to detect the boundaries of traffic signs, which can assist in sign localization and recognition. While color-based segmentation can be useful for traffic sign detection, it has some drawbacks, including noise sensitivity, difficulty in detecting curved edges, reliance on threshold values, and inability to handle occlusion.

1.4. **Object Detection**:
Object detection is a computer vision task that involves detecting and localizing objects of interest within an input image or video. Object detection algorithms have gained popularity in recent years due to their wide range of applications in various fields, including autonomous driving, surveillance, and robotics. Object detection algorithms, such as the popular YOLO can be used to detect traffic signs from the input image as an object.

1.5. **Deep learning based neural networks approach**:
Deep learning approaches such as convolutional neural networks (CNNs), have demonstrated promising results in traffic sign detection and recognition [13][14]. These methods can learn complex features from raw images with high accuracy. Other techniques, in addition to CNNs, have been used for traffic sign detection, such as color-based segmentation, shape-based features, edge detection, and object detection, often in combination with CNNs or other approaches to improve accuracy and robustness.
We propose a CNN-based approach for real-time traffic sign detection and mapping using Python, OpenCV, and TensorFlow in this paper. We intend to evaluate our approach's performance and compare it to existing approaches in terms of accuracy, speed, and robustness. We hope to improve road safety by combining a traffic sign detection and mapping system with a navigation system.

IV. PROPOSED SYSTEM

The proposed system includes a stage for adding the detected road sign onto a map for road sign navigation, in addition to the three stages of **detection**, **feature extraction**, and **recognition**. After detecting and recognizing a traffic sign, its location is determined and added to the map, which is displayed to the driver in navigation mode. This feature alerts the driver to upcoming road signs, allowing them to navigate the road more safely and efficiently in the future without using a camera for sign detection. The block diagram of the proposed system is illustrated in Figure 1.

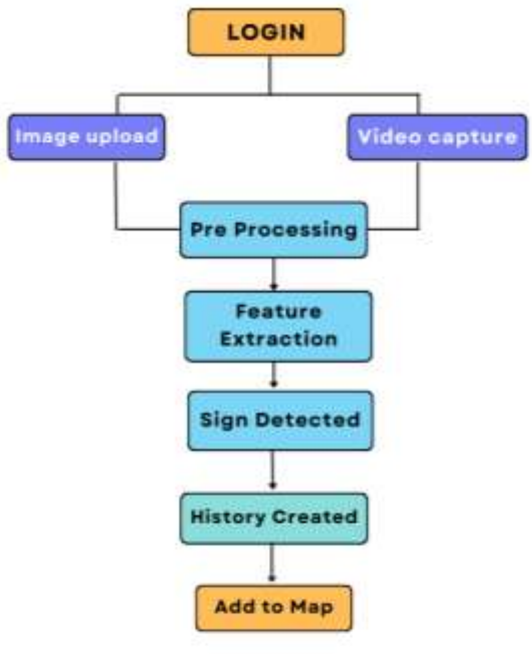

Figure 1: Diagrammatic representation of proposed system.

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 7, Issue 12, ISSN No. 2455-2143, Pages 103-107
Published Online April 2023 in IJEAST (http://www.ijeast.com)

V. METHODOLOGY

1.6. **Data Collection**:
To train the model, we have used German Traffic Sign Recognition datasets. This dataset contains around 43 traffic sign classes that are divided into 39,209 as a training image and 12,630 as a test images. A camera installed inside the vehicle or a smartphone records a video that is converted to a series of images. The user can also upload an image file. These images are processed in order to detect traffic signs or any other signs that are installed on side of the road.

1.7. **Data Pre-processing**:
Eliminate any existing noise and irrelevant details from the recorded data, and standardize the image size and format. High resolution images are scaled down to a lower resolution, and RGB images are converted to greyscale format so that it can be easily processed through CNN.

1.8. **Model Training**:
Using the pre-processed and augmented data, training of the selected CNN model is carried out. This entails splitting the data into training and validation sets, configuring hyper parameters like learning rate, and running the model for multiple epochs to optimize the weights for achieving the better accuracy.

1.9. **Recognition**:
After extracting the features from the input image, the CNN model uses these features to classify the traffic sign into one of the several predefined classes [11]. This classification is based on the output of the CNN models i.e., final layer. which represents the probability distribution across the various traffic sign classes. The most likely traffic sign is then identified as the recognized traffic sign. The recognition phase is an important part of the traffic sign recognition because accurate and dependable detection of traffic signs is required for safe and efficient driving.

1.10. **Developing the navigation system**:
This can be accomplished by using one of the numerous Open Street Map [6][7] data download tools available on the internet. After downloading and pre-processing the Open Street Map data, it must be integrated into the navigation system. This can be accomplished by utilizing appropriate libraries and APIs for the programming language.
In the history, the location where the sign is detected. This information can be obtained from the device's GPS module, which can provide the location's latitude and longitude coordinates. The coordinates, as well as the detected sign and other relevant information, such as the time and date of detection, can be saved. The system can provide the user with a record of the signs detected at various locations by saving this information, which can be useful for future reference or analysis.
Once a detected signs location is saved in its history, it can be used to mark that sign on a specific location on the map. The marked signs will serve as way points for the user to follow, making navigation more efficient. The system can provide a warning for upcoming signs on the user's route by using the location history of a sign, eliminating the need for camera detection.

VI. RESULTS

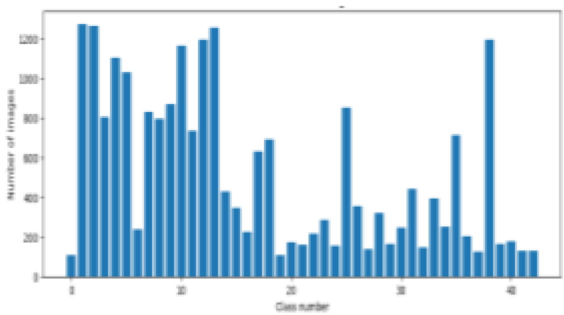

Figure 2: Distribution of Training Data

The plt.bar function from Matplotlib library in Python is used to plot the data and shown in Figure 2. Which displays a bar graph representing the distribution of the training dataset. The ‘num_of_samples’ list contains the number of images for each class in the dataset, where the x-axis represents the class number and the y-axis represents the number of images.

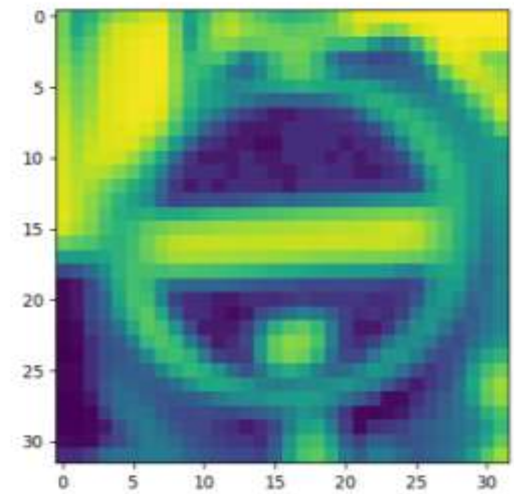

Figure 3.1: Augmented image samples.

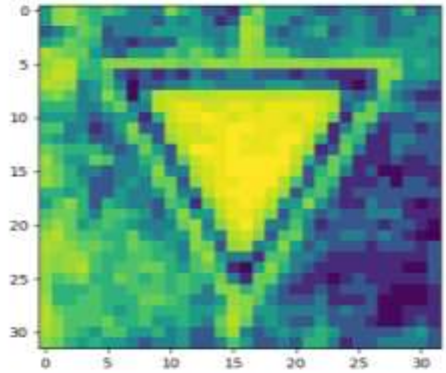

Figure 3.2: Augmented image samples.

Performing pre-processing on the training, validation, and test images. It first defines three functions: grayscale, equalize, and pre-processing. The grayscale function converts the image to grayscale, the equalize function standardizes the lighting in the image using histogram equalization, and the pre-processing function applies these two operations in sequence and normalizes the pixel values between the range 0 and 1. Which is shown in Figure 3.1 and Figure 3.2, respectively. These images show the generated images by applying various transformations and manipulations to the original images in a dataset.

Augmenting the training dataset in this way can help increase the variability of the images seen by the model during training, which can improve its ability to generalize to new, unseen images.

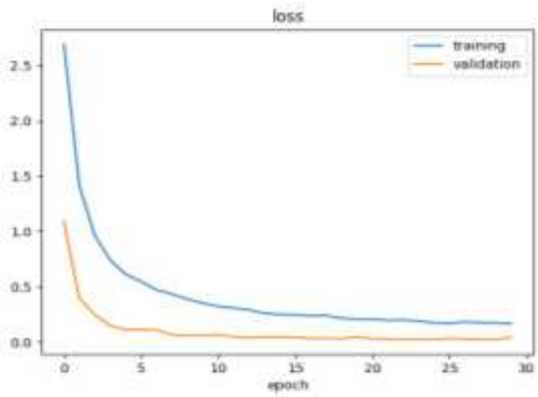

Figure 4: loss of the model on the training and validation data over each epoch.

The loss is computed for each batch of data and averaged across all batches to obtain the loss for one epoch of training that is shown in Figure 4. The same is done for the validation data, which is a separate set of data that the model is not trained on, but is used to evaluate its performance on unseen data.

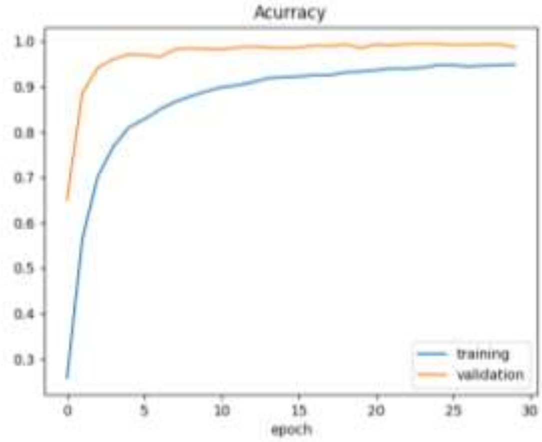

Figure 5: the accuracy of the model on the training and validation data over each epoch.

The accuracy graph shows how well the model is able to learn the patterns in the data over time. The goal is to achieve high accuracy on both the training and validation datasets, which indicates that the model is generalizing well to new examples and not over fitting to the training data. The training accuracy measures the proportion of the training examples that are correctly classified by the model. The validation accuracy measures the proportion of the validation examples that are correctly classified by the model. Which is shown in Figure 5.

Figure 6: Processed image

Figure 7: Output

The resultant image of the system shows that the sign has been successfully detected as "Stop" with a high probability of 96.44%. This indicates that the system has accurately recognized the sign and has a high level of confidence in its prediction. That area are shown in Figure 6 and Figure 7.

Image	Class Name	Date-Time	Location
00035.png	No entry	April 17, 2023, 4:10 a.m.	Dugawan, Naka Hindola, Lucknow, Uttar Pradesh, 226019, India
00009.png	Ahead only	April 16, 2023, 3:10 p.m.	Viram Khand 3, Gwari, Lucknow, Uttar Pradesh, 226010, India
00001_xZ314DG.png	Speed Limit (30 Km/h)	April 16, 2023, 2:32 p.m.	Viram Khand 3, Gwari, Lucknow, Uttar Pradesh, 226010, India

Table 1: Prediction history

One of our system's key advantages is the ability to save the prediction history of detected traffic signs that is shown in Table 1. Which allowing the system to mark their locations on a map. This feature has the potential to improve driver safety by allowing users to navigate roads and avoid potential hazards more confidently.

We aimed to develop a system for traffic sign recognition and navigation using Open Street Maps. The high accuracy achieved on both the training and test datasets suggests that the use of Convolutional Neural Networks (CNNs) for traffic sign recognition is highly effective, as evidenced by our findings. Furthermore, the addition of navigation features based on Open Street Maps has allowed the system to guide users more efficiently and accurately.

However, our study has limitations that must be addressed in future research. One limitation is the relatively small dataset size. Furthermore, our system currently only supports a limited number of traffic signs, and expanding it to include more signs could improve its effectiveness.

Our research shows the effectiveness of CNNs for traffic sign recognition as well as the potential benefits of integrating navigation features using Open Street Maps. These findings have implications for the development of intelligent transportation systems as well as driver safety.

VII. CONCLUSION

The paper has presented a novel system for traffic sign recognition and navigation based on computer vision and open-source mapping tools. The system recognised traffic signs with high accuracy of 99.27% and obtain GPS location of the detected sign (location accuracy within 5 to 15 meters under ideal conditions) to efficiently integrate them into a navigation system. The low-test score of 0.0274 also indicates that the model has very low error on the test data, which is a good sign.

Overall, these results indicate that the model has been successful in learning the patterns in the training data and generalizing well to new, unseen data. This show that the proposed system can significantly improve driver safety and convenience, particularly in unfamiliar areas. The successful implementation of this system opens new avenues for research into intelligent transportation systems. Future work could concentrate on optimizing the system for real-time performance and expanding its applicability to other aspects of traffic safety and management. Overall, this study is a significant step forward in the development of more intelligent and efficient transportation systems.

VIII. REFERENCES

[1]. Han S., Kim D., & Ko H. (2018). Traffic Sign Recognition Based on Convolutional Neural Networks and Transfer Learning Sensors. 18(7).

[2]. Luo J., Wu Z., &Fu Y. (2018). Real-time traffic sign detection and recognition using deep learning, IEEE Transactions on Intelligent Transportation Systems. 19(1). 277-286.

[3]. Yang H., Guo X., and Liu J. (2016). Real-time traffic sign recognition and analysis. IEEE Transactions on Intelligent Transportation Systems. 17(7).

[4]. Zhang X., Zhu Y., and Liu J. (2016). Traffic sign detection using color and shape features. Multimedia Tools and Applications, 75(13).

[5]. Zhao H., Guo C., Yang S., and Huang D. (2019) Traffic sign detection based on a combination of shape and texture features.

[6]. Haklay M. and Weber P. (2008). OpenStreetMap: User-Generated Street Maps, IEEE Pervasive Computing, vol. 7, no. 4, pp. 12-18.

[7]. Vargas-Munoz J. E., Srivastava S., TuiaD. and Falcão A. X., (2021), OpenStreetMap: Challenges and Opportunities in Machine Learning and Remote Sensing, in IEEE Geoscience and Remote Sensing Magazine, vol. 9, no. 1, pp. 184-199.

[8]. Shyam R and Chakraborty R. (2021). Machine Learning and its Dominant Paradigms. Journal of Advancements in Robotics; 8(2): 1–10p.

[9]. Megalingam R. K. (2022). Indian traffic sign detection and recognition using deep learning, IJTST.

[10]. Kumar G. B., Rani N. A. (2020). Traffic Sign Detection using Convolutional Neural Network. IJCRT | Volume 8, Issue 5.

[11]. Simran S. T., Khanna S., Shyam R. (2022). Detection of Traffic Sign Using CNN. Recent Trends in Parallel Computing.; 9(1): 14–23p.

[12]. Verma S., Jaiswal V., Shyam R. (2022). Intensifying Security with Smart Video Surveillance. Recent Trends in Programming Languages.; 9(1): 24–30p.

[13]. Shyam R. (2021). Convolutional Neural Network and its Architectures. Journal of Computer Technology & Applications; 12(2): 6–14p.

[14]. Srivastava V., Shyam R. (2021). Enhanced Object Detection with Deep Convolutional Neural Networks, International Journal of All Research Education and Scientific Methods (IJARESM), Volume 9, Issue 7.