www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

Deep Neural Networks for RobustTraffic Light Recognition inAutonomous Navigation

Dr. M.Swapna[1*],Gangishetti Snigdha[2],Korutla Lahari[2], Anantharam Murali Mohan[2], Bussari Rakesh[2]

[1]Assistant Professor, [2]UG Student, [1,2]Department of Computer Sciences and Engineering,

[1,2]Vaagdevi College of Engineering (UGC - Autonomous), Bollikunta, Warangal, Telangana.

[*]Corresponding author: Dr. M.Swapna (swapna_m@vaagdevi.edu.in)

ABSTRACT

Traffic Sign Detection and Recognition (TSDR) plays a critical role in the development of driver assistance systems and autonomous vehicle technologies. Accurate and reliable recognition of traffic signs is essential for ensuring road safety and efficient navigation. However, real-world deployment of TSDR systems faces numerous challenges due to the wide variety of traffic signs, as well as adverse conditions in which images are captured. Traffic signs in the wild are often obscured by environmental factors such as rain, fog, poor lighting, and motion blur, all of which degrade the quality of captured images and significantly impact recognition accuracy. Existing machine learning approaches for TSDR have shown promising results, but they are often trained and evaluated on clean, ideal datasets. These methods typically fail to generalize effectively under real-world challenging conditions (CCs), resulting in performance degradation and misclassification in critical situations.To address these limitations, this study proposes a robust Convolutional Neural Network (CNN)-based TSDR framework enhanced with pre-processing techniques to improve image quality before classification. The proposed system integrates an image enhancement module that preprocesses traffic sign images affected by CCs to restore clarity and contrast. The refined images are then passed through a CNN model trained on both clean and augmented data, enabling better adaptability to varying environmental factors. This hybrid approach significantly improves recognition accuracy and robustness under challenging conditions, making it highly suitable for deployment in real-time autonomous driving systems. Our framework ensures more reliable TSDR performance, contributing to safer autonomous navigation in diverse environments.

Keywords: Traffic sign recognition, Convolutional Neural Network (CNN), Autonomous vehicles, Adverse weather conditions, Image enhancement.

1. INTRODUCTION

The ability of autonomous driverless vehicles to navigate safely and make informed driving decisions is greatly aided by traffic sign recognition (TSR), a crucial part of the technology stack. TSR systems are made to recognize and decipher traffic signs and signals, giving the vehicle's control system vital information. In order to process data from onboard cameras and sensors, these systems usually combine computer vision, machine learning, and deep neural networks. Cameras start the process by taking real-time pictures of the area around the car, including traffic lights, road signs, and other pertinent traffic-related data. Advanced image recognition algorithms that can recognize and categorize a variety of traffic sign types, including yield signs, stop signs, speed limits, and more, are then used to process these images. In order to improve the visibility of signs in a variety of lighting and weather conditions, this recognition procedure frequently uses sophisticated image preprocessing techniques. Following the detection and classification of a traffic sign, the TSR system decodes its meaning and transmits the information to the vehicle's control system. For example, the vehicle can adjust its speed in accordance with traffic laws and safety regulations if it recognizes a speed limit sign. TSR is also essential for recognizing traffic signals and stop signs, which allows the car to safely navigate intersections and stop completely when necessary. TSR systems use machine learning techniques to continuously improve in order to achieve high accuracy and robustness. Large datasets of different traffic sign images taken in a variety of settings and situations are used to train them, enabling the algorithms to learn and adjust to a wide range of real-world situations. Additionally, TSR systems can benefit from hardware advancements that increase their speed and accuracy, such as powerful GPUs, high-resolution cameras, and specialized AI chips. Therefore, traffic sign recognition is a key technology for autonomous driverless cars since it improves their comprehension and response to traffic laws, protecting pedestrians, passengers, and other road users. TSR systems will continue to develop and play a vital role in bringing self-driving transportation to reality as autonomous vehicles become more common on our roads.

2. LITERATURE SURVEY

A new flexible gaussian mixture model-based method with automatic split and merge strategy was introduced by Mannan et al. [4]. In a real-world scenario, this adaptive scheme functions as a preprocessing step that makes it easier to locate traffic signs that are degraded to a certain extent. To identify the contents of the sign, a multiscale convolutional neural network enhanced with a dimensionality reduction layer is suggested. Since no benchmark dataset is available for this purpose, we gathered a number of images from the well-known German traffic sign detection benchmark that contained partially degraded signs. They supplemented these images with manually and naturally degraded traffic sign images from the longest highway in the authors' home country. According to experimental results, our suggested method for detecting and recognizing deteriorated traffic signs performs better than many recent and state-of-the-art approaches. MF-SSD, a multi-feature fusion and enhancement algorithm, was proposed by Jin et al. [5] as an improved (single shot detector) SSD algorithm for traffic sign recognition. Initially, low-level features are combined with high-level features to enhance the SSD's ability to detect small targets. In order to detect the target, we then improve the features in various channels by suppressing invalid channel features and enhancing effective channel features. When it comes to domestic real-time traffic signs, our algorithm performs well. Using the German Traffic Sign Recognition Benchmark Dataset, the suggested MF-SSD algorithm is assessed. The experimental findings demonstrate the benefits of the MF-SSD algorithm for identifying small traffic signs. It achieves better robustness, efficiency, and detection accuracy in complex traffic environments as compared to existing methods.

Gámezserna et. al [6] introduced a real-world European dataset for traffic sign classification. the dataset is composed of traffic sings from six European countries: Belgium, Croatia, France, Germany, the netherlands, and Sweden. it gathers publically available datasets and complements French traffic signs with images acquired in Belfort with the equipped university autonomous vehicle. It is composed of more than 80000 images divided in 164 classes that at the same time belong to four main categories following the Vienna convention of road signs. we analyzed the intra variability of classes and compared the classification performance of five convolutional neural network architectures.Varytimidis et. al [7] investigates a number of feature extraction methods in combination with several machine learning algorithms to build knowledge on how to automatically detect the action and intention of pedestrians in urban traffic. They focused on the motion and head orientation to predict whether the pedestrian is about to cross the street or not. This work is based on the joint attention for autonomous driving (JAAD) dataset, which contains 346 videoclips of various traffic scenarios captured with cameras mounted in the windshield of a car. an accuracy of 72% for head orientation estimation and 85% for motion detection is obtained in our experiments.Bangquan et. al [8] introduced a new efficient TSC network called E-net (efficient network) and a TSD network called EMBnet (efficient network using multiscale operation and depth wise separable convolution). They used data mining and multiscale operation to improve the accuracy and generalization ability and used depthwise separable convolution (DSC) to improve the speed. Theresulting Enet possesses 0.9 m parameters (1/15 the parameters of the start-of-the-art method) while still achieving an accuracy of 98.6 % on the German traffic sign recognition benchmark. in addition, we design EMDnet' s backbone network according to the principles of ENet. the EMDNet with the SDD framework possesses only 6.3 m parameters, which is similar toMobileNet's scale.Tabernik et. al [9] address the issue of detecting and recognizing a large number of traffic-sign categories suitable for automating traffic-sign inventory management. we adopt a convolutional neural network (CNN) approach, the mask r-CNN, to address the full pipeline of detection and recognition with automatic end-to-end learning. we propose several improvements that are evaluated on the detection of traffic signs and result in an improved overall performance. This approach is applied to detection of 200 traffic-sign categories represented in our novel dataset. Theresults are reported on highly challenging traffic-sign categories that have not yet been considered in previous works. They provide comprehensive analysis of the deep learning method for the detection of traffic signs with a large intra-category appearance variation and show below 3% error rates with the proposed approach, which is sufficient for deployment in practical applications of the traffic-sign inventory management.

Cao et. Al [10] improved traffic sign detection and recognition algorithm is proposed for intelligent vehicles. Firstly, the hsv color space is used for spatial threshold segmentation, and traffic signs are effectively detected based on the shape features. Secondly, this model is considerably improved on the basis of the classical lenet-5 CNN model by using gabor kernel as the initial convolutional kernel, adding the bn processing after the pooling layer, selecting adam method as the optimizer algorithm. Finally, the traffic sign classification and recognition experiments are conducted based on the gtsrb. The favorable prediction and accurate recognition of traffic signs are achieved through the continuous training and testing of the network model. The experimental results show that the accurate recognition rate of traffic signs reaches 99.75%, and the average processing time per frame is 5.4 ms. The proposed algorithm has more admirable accuracy, better real-time performance, stronger generalization ability and higher training efficiency than other algorithms. The accurate recognition rate and average processing time are significantly improved.Wali et. Al [11] provides a comprehensive survey on traffic sign detection, tracking and classification. The details of algorithms, methods and their specifications on detection, tracking and classification are investigated and summarized in the tables along with the corresponding key references. A comparative study on each section has been provided to evaluate the tsdr data, performance metrics and their availability. Current issues and challenges of the existing technologies are illustrated with brief suggestions and a discussion on the progress of driver assistance system research in the future. This review will hopefully lead to increasing efforts towards the development of future vision-based tsdr system.

Liu et. Al [12] proposed a new transfer learning structure based on two novel methods of supplemental boosting and cascaded convnet to address this shortcoming. The supplemental boosting method is proposed to supplementally retrain an adaboost-based detector for the purpose of transferring a detector to adapt to unknown application scenes. The cascaded convnet is designed and attached to the end of the adaboost-based detector for improving the detection rate and collecting supplemental training samples. With the added supplemental training samples provided by the cascaded convnet, the adaboost-based detector can be retrained with the supplemental boosting method. The detector combined with the retrained boosted detector and cascaded convnet detector can achieve high accuracy and a short detection time. As a representative object detection problem in intelligent transportation systems, the traffic sign detection problem is chosen to show our method. Through experiments with the public datasets from different countries, we show that the proposed framework can quickly detect objects in unknown application scenes.

Zhang et. Al [13] presented a new way to evaluate traffic sign visual recognizability in each lane. The proposed model not only quantitatively expresses the visibility and recognizability of a traffic sign from a viewpoint, but also continuously expresses visibility and recognizability, within sight distance, over the entire road surface. Unlike the existing methods for studying visibility and recognizability limited by position of viewpoint in 2d space or cannot be applied in the real road environment, we proposed a new way to evaluate visibility and recognizability in 3d space conquered those problem. Based on traffic sign detection method [31], our algorithm can automatically process more than (92.61% in [31]) traffic signs. The rest traffic signs can be manually detected and processed by our algorithm. Our methods also can be used to detect occlusion and inspect spatial installation information of traffic signs for inventory purposes. Moreover, our model, because it has a process similar to traffic signs, can be easily expanded to other traffic devices, such as traffic lights.

3. Proposed System

This research is aimed at improving traffic sign recognition for autonomous driverless vehicles operating under adverse weather conditions. It involves a sequence of steps, starting with the acquisition of hazy traffic sign images and progressing through haze removal, DLCNN -based sign detection, and performance evaluation through loss and accuracy calculations. Ultimately, the goal is to develop a robust system that can reliably recognize and interpret traffic signs in challenging environmental conditions, contributing to the safety and efficiency of autonomous driving systems. Figure 4.1 shows the proposed system model. The detailed operation illustrated as follows:

step 1. Hazy traffic sign image: this is the initial phase of the project where you start with hazy or degraded traffic sign images. These images are likely to be affected by adverse weather conditions, such as fog, rain, or haze, which can obscure the visibility of traffic signs.

step 2. Deep learning haze removal: in the second step, you employ deep learning techniques to perform haze removal from the hazy traffic sign images. This involves using deep learning convolutional neural networks (DLCNNS) or similar deep learning architectures to enhance the clarity and visibility of the traffic signs in the images by mitigating the effects of haze.

step 3. DlCNN traffic sign detection: after removing the haze, you proceed to the core task of traffic sign recognition. In this step, a dlCNN -based model is employed to detect and recognize traffic signs within the processed images. The dlCNN is trained to identify various traffic sign types, including speed limits, stop signs, yield signs, etc.

step 4. Accuracy and loss estimation: to train and evaluate the dlCNN model's performance, calculate the loss and accuracy during the training process. The loss function measures the difference between the predicted traffic sign labels and the ground truth labels.

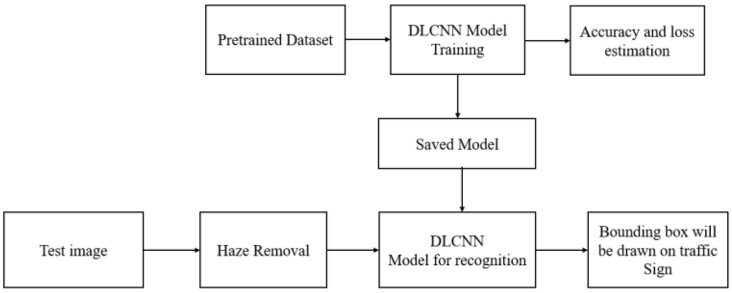

Fig. 1: Block diagram of proposed system.

3.2 Haze Removal

This project revolves around the development of a deep learning-based system for image dehazing, particularly with a focus on enhancing the clarity of hazy or obscured images, which is a common problem in computer vision and image processing.

Step 1. Global variable initialization: global variables like dehaze_model, saver, rgb, and max are declared. These variables play essential roles in storing and managing components used in the deep learning model.

Step 2. Data reading and preprocessing: the project begins by gathering a dataset consisting of two types of images: "clear" and "haze." these images likely represent pairs of original (clear) and hazy versions of the same scenes, which are essential for training and evaluating the model.

Step 3. Tensorflow graph reset: tensorflow, a popular deep learning framework, is used. Resetting the tensorflow graph ensures a clean slate for defining and running subsequent computations without interference from previous operations.

Step 4. Data generation: the collected data is preprocessed and organized for model training and evaluation. This step involves tasks like resizing, normalizing, and splitting the dataset into training and testing subsets.

Step 5. Tensorflow placeholders: tensorflow placeholders, rgb and max, are introduced. These placeholders serve as symbolic inputs to the deep learning model. Rgb likely represents input hazy images, while max represents the target clear images.

Step 6. Model initialization: the core of the project involves building and initializing a deep learning model, referred to as dehaze_model. This model is likely based on an encoder-decoder architecture, specialized in the task of image dehazing.

Step 7. Loss calculation: to quantify how well the model is performing, a loss function is defined. The project utilizes the mean squared error (trainingloss) as the loss metric. It measures the disparity between the dehazed images predicted by the model and the actual clear images.

Step 8. Optimization and gradients: the model's parameters are optimized using an adam optimizer, a common optimization algorithm in deep learning. Gradients of the loss with respect to the trainable model parameters are computed.

Step 9. Gradient clipping: gradient clipping is applied to ensure stable training. This technique limits the magnitude of gradients during optimization, preventing them from becoming too large and causing instability.

www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

Step 10.Gradient descent: the computed and clipped gradients are employed to update the model's parameters, iteratively minimizing the loss. This process is essential for training the model to perform the dehazing task effectively.

Step 11.Model saving: finally, a tensorflow saver object (saver) is instantiated. This object allows the trained model's parameters to be saved to disk. This step is crucial for preserving the trained model for future use or deployment in real-world applications.

4.RESULTS AND DISCUSSION

Figure 2 provides a visual demonstration of the performance of a dehazing model. It consists of two images side by side. The first image is a "weather-affected" image, which could be hazy, cloudy, rainy, or captured in poor lighting conditions. This image is typically challenging to interpret due to the weather-related issues. The second image is the "clean" image, which serves as a reference or ground truth. It represents what the image should ideally look like without any weather-induced degradation.

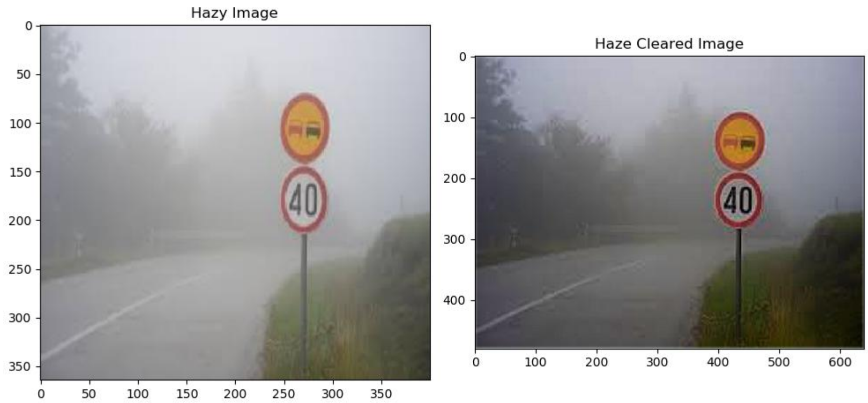

Figure 2. Dehazed performance.

In figure 3, the focus is on traffic sign detection using a CNN model. It likely shows an image with detected traffic signs, and bounding boxes may be drawn around them to indicate their locations. This figure demonstrates the successful application of the CNN model for traffic sign recognition.

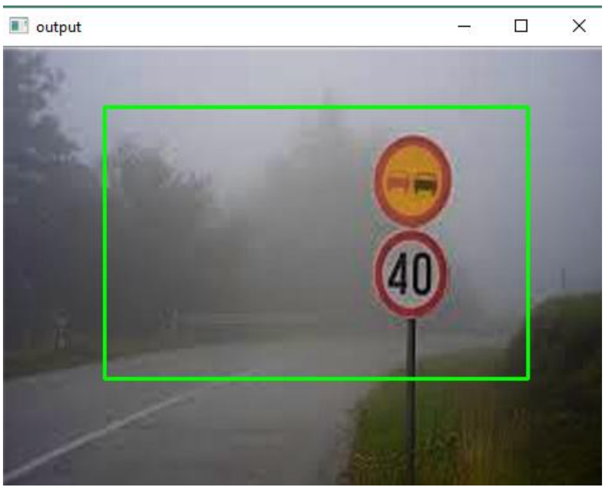

Figure 3. Traffic sign detected image.

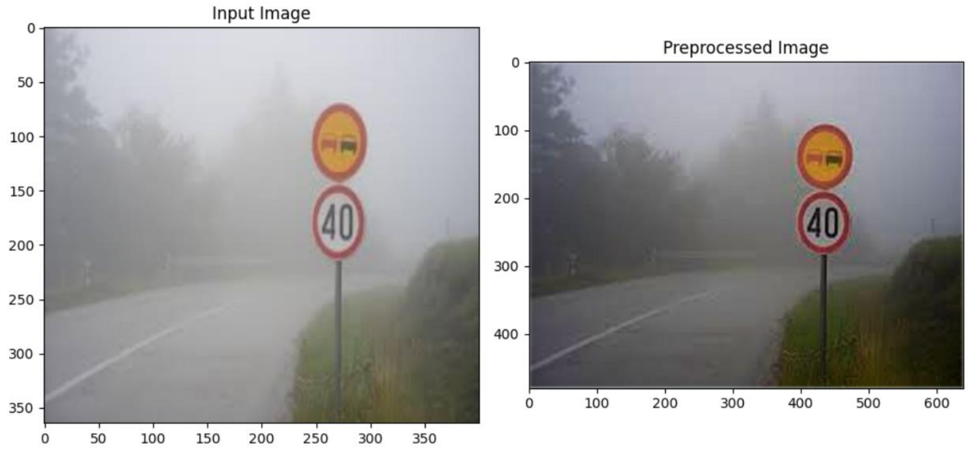

Figure 4. Dehazed performance on sample 2.

Figure 5 shows the results of traffic sign detection and speaks out the sound based on the predicted class on the second sample image mentioned in figure 10.4. It displays the detected traffic signs and any bounding boxes or annotations associated with them. This is a continuation of the traffic sign detection process.

Figure 6 represents a line plot or a graph that visualizes the performance of the proposed CNN model over the course of training. It typically consists of two subplots:

- One subplot showing the model's accuracy on the training and validation datasets over the course of 10 training epochs. This helps assess how well the model is learning to recognize traffic signs.
- Another subplot showing the model's loss (e.g., mean squared error or cross-entropy loss) on the training and validation datasets across the 10 epochs. This helps monitor how well the model is converging during training and whether it's overfitting or underfitting.

Figure 5. Traffic sign detected from sample 2.

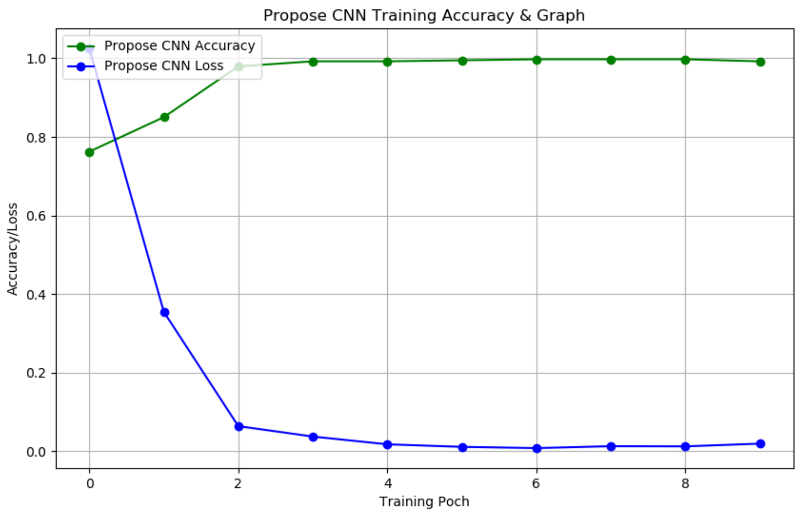

Figure 6. Proposed CNNaccuracy and loss for 10 epochs.

Table 1, performance metrics are compared across two different models: the “existing RFC” (random forest classifier) model, and the “proposed CNN” (convolutional neural network) model. The table provides values for two key performance metrics, namely “accuracy (%)” and “loss,” for each of these models. Let’s explain the table:

- **Metric:** this column specifies the performance metric being measured, which is categorized into two models: “existing RFC,” and “proposed CNN.”
- **Existing RFC (random forest classifier):**
    - **Accuracy (%):** the “existing RFC” model got the accuracy of 50.5. It correctly classifies approximately 50.5% of the test data samples.
    - **Loss:** the loss for the “existing RFC” model is 49.5, which is lower than that of the model. This indicates that the RFC model’s predictions are closer to the true labels, resulting in a lower loss.
- **Proposed CNN (convolutional neural network):**
    - **Accuracy (%):** the “proposed CNN” model attains a perfect accuracy of 100%. This implies that the CNN model correctly classifies all of the test data samples, indicating a flawless performance in terms of accuracy.
    - **Loss:** the loss for the “proposed CNN” model is 0.0584, which is the lowest among the two models. This suggests that the CNN model’s predictions are very close to the true labels, resulting in the smallest loss.

Table 1. Performance Comparison

Metric	Existing RFC	Proposed CNN
Accuracy (%)	50.5	100
Loss	49.5	0.0584

5. CONCLUSION

In conclusion, traffic sign detection using CNN represents a transformative advancement in the realm of computer vision and autonomous transportation systems. The utilization of CNNs offers a multitude of advantages, including unparalleled accuracy in recognizing and classifying traffic signs, robustness in the face of diverse environmental conditions and sign variations, and real-time processing capabilities crucial for ensuring road safety. These systems reduce human intervention, minimize the risk of errors, and enhance the adaptability of autonomous vehicles to different regions and signage styles. Moreover, they contribute to safer roadways by ensuring that vehicles accurately interpret and respond to traffic signs, ultimately leading to improved road safety and traffic management. As technology continues to evolve, CNN-based traffic sign detection systems are poised to play an increasingly pivotal role in advancing the capabilities of autonomous vehicles, making our roads safer and more efficient.

REFERENCES

[1] H. Li, f. Sun, l. Liu and l. Wang, "a novel traffic sign detection method via color segmentation and robust shape matching", neurocomputing, vol. 169, pp. 77-88, dec. 2015.

[2] A. Gudigar, s. Chokkadi, u. Raghavendra and u. R. Acharya, "multiple thresholding and subspace-based approach for detection and recognition of traffic sign", multimedia tools appl., vol. 76, no. 5, pp. 6973-6991, feb. 2016.

[3] S. K. Berkaya, h. Gunduz, o. Ozsen, c. Akinlar and s. Gunal, "on circular traffic sign detection and recognition", expert syst. Appl., vol. 48, pp. 67-75, apr. 2016.

[4] A. Mannan, k. Javed, a. Ur rehman, h. A. Babri and s. K. Noon, "classification of degraded traffic signs using flexible mixture model and transfer learning," in ieee access, vol. 7, pp. 148800-148813, 2019, doi: 10.1109/access.2019.2947069.

[5] Y. Jin, y. Fu, w. Wang, j. Guo, c. Ren and x. Xiang, "multi-feature fusion and enhancement single shot detector for traffic sign recognition," in ieee access, vol. 8, pp. 38931-38940, 2020, doi: 10.1109/access.2020.2975828.

[6] C. Gámezserna and y. Ruichek, "classification of traffic signs: the european dataset," in ieee access, vol. 6, pp. 78136-78148, 2018, doi: 10.1109/access.2018.2884826.

[7] D. Varytimidis, f. Alonso-fernandez, b. Duran and c. Englund, "action and intention recognition of pedestrians in urban traffic," 2018 14th international conference on signal-image technology & internet-based systems (sitis), 2018, pp. 676-682, doi: 10.1109/sitis.2018.00109.

[8] X. Bangquan and w. Xiao xiong, "real-time embedded traffic sign recognition using efficient convolutional neural network," in ieee access, vol. 7, pp. 53330-53346, 2019, doi: 10.1109/access.2019.2912311.

[9] D. Tabernik and d. Skočaj, "deep learning for large-scale traffic-sign detection and recognition," in ieee transactions on intelligent transportation systems, vol. 21, no. 4, pp. 1427-1440, april 2020, doi: 10.1109/tits.2019.2913588.

[10] Cao, j., song, c., peng, s., xiao, f. And song, s., 2019. Improved traffic sign detection and recognition algorithm for intelligent vehicles. Sensors, 19(18), p.4021.

[11] Wali, s.b., abdullah, m.a., hannan, m.a., hussain, a., samad, s.a., ker, p.j. And mansor, m.b., 2019. Vision-based traffic sign detection and recognition systems: current trends and challenges. Sensors, 19(9), p.2093.

[12] Liu, c., li, s., chang, f. And dong, w., 2018. Supplemental boosting and cascaded convnet based transfer learning structure for fast traffic sign detection in unknown application scenes. Sensors, 18(7), p.2386.

[13] Zhang, s., wang, c., lin, l., wen, c., yang, c., zhang, z. And li, j., 2019. Automated visual recognizability evaluation of traffic sign based on 3d lidar point clouds. Remote sensing, 11(12), p.1453.