IFERP

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 10 Issue 11 November 2023

An AI-Powered Interactive Traffic Sign Recognition System with Detection and Analysis

[1] Suhas GC*, [2] Rekha P M

[1] PG Student, Department of Information Science and Engineering, JSS Academy of Technical Education, Bangalore, Karnataka, India
[2] Professor, Department of Information Science and Engineering, JSS Academy of Technical Education, Bangalore, Karnataka, India
Corresponding Author Email: [1] suhasgc2000@gmail.com, [2] rekhapm12@gmail.com

Abstract - Traffic signs help drivers stay safe while traveling. Each type of sign has a specific function and tells people what to do in specific situations. Many road users don't know what the signs mean which can cause confusion and accidents. This is where pictorial crosswalks come in handy. Traffic sign classification has been identified as one of the most important tasks for automated cars to perform. Key focus in this research is to develop efficient methods for traffic sign classification, which avoids crashes and foster public trust in autonomous vehicles. In this paper, Convolutional Neural Network (CNN) model is used to recognize the 43 different traffic sign recognition benchmark datasets. The proposed CNN model for classification is 16 times smaller than the original VGG-19 Model, yet still maintains 96.62% accuracy on its own test set.

Keywords- Traffic sign dataset, Convolutional Neural Network (CNN), Deep Learning, Dropout, Traffic Sign Recognition, Keras GUI.

I. INTRODUCTION

Traffic signs help road users understand what actions are necessary for them to operate safely in dangerous situations. Traffic signs use symbols, characters and colours to convey information. This includes how to detect and recognize traffic signs using the Driver Assistance System "DAS". Traffic sign recognition to ensure safe and efficient driving this rely heavily on Autonomous vehicles and driver assistance systems (ADAS). By recognizing traffic signs using CNN, drivers can follow these rules and avoid accidents. [1] Many companies like Tesla, Audi, Ford etc is the world's leading manufacturer of self-driving cars and has become the world's most valuable automobile company at the time of this writing. This suggests a large demand for self-driving automobiles in the car industry.

Traffic signs are essential for road safety and regulation. However, there is a lack of standardization for these signs. Many drivers tend to confuse traffic signs such as stop and yield with road hazards. To help drivers understand the rules of the road, we must devise better ways to communicate with drivers. One way to improve on current methods of sign communication is to apply recent technology to this problem. We can use neural networks and computer vision to categorize traffic sign types and identify whether a sign is present or not. Convolutional neural networks have proved extremely useful in traffic sign classification and object detection. These networks are particularly effective at performing high-level tasks that require a lot of hand engineering. Instead of creating custom algorithms for each type of traffic sign, we can train a single model to recognize all types of traffic signs. This model can be used in conjunction with road maps, local regulations, and feedback from local authorities to determine the safest route at any given time of day or day of the week. Traffic signs are distinguished in terms of function, and each function is subdivided into different subcategories Due to the similarity in shape and appearance of traffic signs, it is ideal to perform traffic sign recognition in two phases: detection and classification. During the detection step, shared information is used to propose bounding boxes that could potentially contain traffic signs in a particular category. In the classification step, differences are used to identify the specific type of sign present (if any) within the proposed bounding box.

II. RELATED WORKS

Two major types of approaches have been used for traffic sign detection and classification: traditional or conventional methods, and deep learning-based methods [2]. Both have been extensively researched in the field of traffic sign recognition. Convolutional neural networks (CNNs) have emerged as a recent and popular approach for traffic sign recognition. Interpreting traffic signs in particular has been aided by the ability to identify shapes and colours, Color-based methods, which use the RGB color space, Color-based methods such as, color invariants, color thresholding and color segmentation are commonly utilized for traffic sign detection. To reduce the sensitivity to environmental factors, RGB color space [5] are often convert to the other color by using this approach such as HSI[6] or YCbCr[7].

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 10 Issue 11 November 2023

There are many CNNs are increasingly being used to identify objects in images. By comparing the effects got with the aid of specific researchers is a challenging project due to the fact of the use of one-of-a-kind datasets, GTSRB, BTSD, and so on. Different datasets have varied properties, those are number of training images and their quality's. These affect how well a network can be trained to classify objects from those images (Fang et al., 2018) [4].In TSR methods mostly use support vector machine (SVM) [13,14] or extreme machine (ELM) [3] methods can be used to classify features and thus create a model that can be used to predict the probability of an outcome. However, if handcrafted features are used, significant information may be lost. However, as computer and network technology improve, new techniques Classification algorithms, and real-time object detection systems have been developed to enhance overall performance by means of lowering processing time, increasing accuracy, and lowering network size.

In a study by Ueli Meier, Jürgen Schmidhuber[11], Jonathan Masci, Dan Ciresan, Multiple deep neural networks (DNNs) are trained on pre-processed data and then combined to form a multi-column deep neural network (MCDNN). This technique is used to enhance the classification and recognition tasks in traffic sign recognition. Since CNNs are able to learn features in a hierarchical manner, most research has focused on CNNs for traffic sign classification.This work proposes a traffic sign recognition method using CNN to classify traffic signs Publicly accessible German traffic sign dataset (GTSRB). Experimental consequences primarily based on this neural network architecture provides that CNN[12,15,16] work correctly and offers higher performance.

Figure 1. The images shown is from GTSRB dataset

III. METHODOLOGY

In this study, the three-step approach is being adopted. Firstly, a neural network model is developed and a corresponding learning algorithm is implemented. Secondly, CNN model is trained on the traffic sign classification dataset until it converged. Finally, Performance is evaluated on the test set, with previously unseen traffic sign images. The goal was to achieve high accuracy in traffic sign recognition, which is an essential component of autonomous driving and driver assistance systems.

A. Collecting Data and organizing

The dataset is used for this research is the German Traffic Sign Recognition Benchmark (GTSRB), which is a collection of more than 56,000 traffic sign images. The dataset is divided into 43 classes, where each class represents a specific traffic sign.

The GTSRB dataset is organized into three subsets: training, validation, and test sets. The training model includes 39,209 images, the validation set and test set consists of 4,410 images, 12,630 images respectively. Every image in the dataset has a size of 30x30 pixels and is in RGB format. The dataset is collected from various sources, including real-world traffic sign images, computer-generated images, and images from Google Street View. The images are pre-processed to remove background noise, distortions, and other artifacts, and then labelled with the corresponding traffic sign class. The dataset is made available in CSV format, which contains the image filename, class ID, and other metadata. The images are stored in separate folders for each class, making it easy to load and pre-process the data for use with CNN models in Keras. Overall, the GTSRB dataset is a widely used benchmark for evaluating traffic sign recognition models, and its organization and labelling make it a suitable dataset for use in research papers and machine learning projects. Figure 2 depicts the sample classification of images from GTSRB.

The images size in the dataset ranges from 15 × 15 to 250 × 250 pixels values. The Region of Interest ROI, for every image, is supported and was once used. The dataset is approximately 600 MB in size and consists of two main folders: "train" and "test". The "train" folder contains images categorized into their respective classes for training the model, so the "test" folder will be used for check the performance of the model. Using the OS module, that can iterate over the 43 class folders in the 'train' directory, ranging from 0 to 42 shown in the Figure 3 to gather images and their corresponding labels. These are appended to the 'data' and 'labels' lists.

Figure 2. Sample classification of images from GTSRB

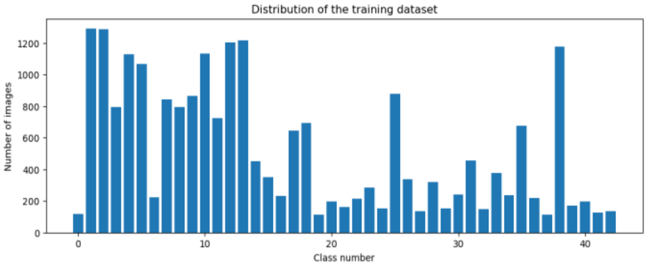

Figure 3. Distribution of images is shown across 43 different classes.

When working with this dataset in the context of Traffic Signs Recognition using CNN & Keras, it is further dividing the training set into two separate sets: a training set and a test set. [24] The primary purpose of the training set is used to train the model, while the test-validation set is utilized to check the model's accuracy or performance during training and optimize the hyperparameters. Finally, the overall performance of the model is evaluated after being trained that employed with test set.

B. Module for classifying traffic signs

The traffic sign classification module employs the GTSRB dataset, which is split into three parts: 60% for training, 20% for testing, and 20% for validation. Since the number of images in each class is unevenly distributed, augmentation techniques are utilized to increase the size of the training set. 20 km/h speed limit level, represented by 0, has 210 Pictures, and the class "speed limit 30 km/h", represented by 2, has 2150 pictures picture. Due to these differences, the model may be biased Bring more pictures to class. [21] Different Enhancement of Parameters Contains random rotation, stretching and flipping, this enhancement Parameters are utilized to mitigate bias by balancing the dataset. Of records, sequential appearance of images of the same class, etc. Because of this existence, images appear by chance one after another the dataset is shuffled to avoid volatility Training and loss functions.

C. Data pre-processing

The given figure 4 shows the pre-processing for the recognize traffic signs, the input image's pixel values are first converted into arrays and fed into the input layers of a neural network. This multi-layers network is designed for

classification, and the hidden layers are responsible for feature extraction through various calculations and manipulations. The feature extraction is performed using multiple hidden layers, such as the convolution layer, ReLU layer, and pooling layer. Finally, a fully connected layer is utilized to identify the traffic sign in the image.

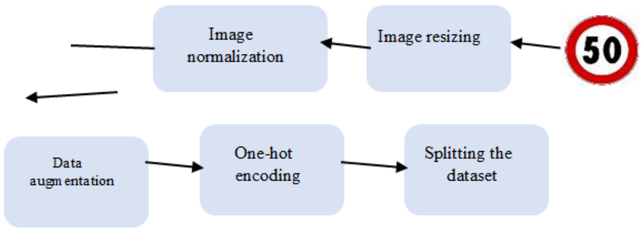

Figure 4: pre-processing the image data.

Image resizing is the images in the dataset are usually of different sizes, so they need to be resized to a common size before training the model. This is usually done using a library like OpenCV or sci-kit-image.

Image normalization is the pixel values in the images are typically normalized to improve the convergence rate of the model. To normalize the pixel values of the image, one can either divide each pixel value by the maximum pixel value, which is 255 for 8-bit images, or subtract the mean pixel value and divide by the standard deviation. The data is shaped as (39209, 30, 30, 3), indicating that there are 39,209 images with a resolution of 30×30 pixels, and the last dimension of 3 signifies that the data consists of colour images (RGB value). Data augmentation is the process of artificially increasing the size of the training dataset by applying transformations to the images. This helps to prevent overfitting and improves the generalization performance of the model. Some common data augmentation techniques include rotation, scaling, shifting, flipping, and adding noise.

One-hot encoding: Since there are 43 different traffic sign labels that are categorical in the training dataset, they need to be encoded as numerical values for the model to process them. One-hot encoding is a common technique used to encode categorical data, where each label is represented as a vector with a 1 in the index corresponding to the label and 0s elsewhere.

Splitting the dataset in GTSRB dataset is typically divided into three subsets: training, validation, and testing sets. The train subset was employed to training the model, the validation subset is used to fine-tune the model's hyperparameters and, the testing subset is utilized to assess the model's performance on unseen data. These pre-processing steps help to prepare the dataset for training the CNN model for Traffic Signs Recognition.

D. Traffic sign recognition module

In Traffic sign detection using CNN and Keras, a convolutional neural network (CNN) is trained to recognize traffic signs. The CNN model is designed to take an image of a traffic sign as input and predict the class of the sign from a set of pre-defined classes.

The training process of the CNN involves the use of a large dataset of labelled images of traffic signs. The dataset is pre-processed by resizing the images to a consistent size and applying image normalization to reduce variations in color

IFERP
ISSN (Online) 2394-2320
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Volume 10 Issue 11 November 2023

and contrast. Additionally, data augmentation techniques such as rotation, zoom, and horizontal flipping are applied to the training data to increase the diversity of the dataset and prevent overfitting. [23] The CNN model used for traffic sign detection typically consists of several convolutional layers, followed by a set of pooling layers and then a series of fully connected layers. The convolutional layers learn to detect low-level features such as edges and corners, while the pooling layers down sample the feature maps and help the network to be robust to translation and rotation. The fully connected layers are responsible for the final classification.

Once the CNN model has been trained, it can be integrated into a traffic sign detection module. This module takes input from a camera and passes the image through the CNN model for classification. The output of the CNN is then displayed on a graphical user interface (GUI), allowing the user to see the detected traffic sign class in real-time. The GUI can be designed to display the video feed from the camera and overlay the predicted traffic sign class on top of the video. The GUI can also be used to display other relevant information such as the confidence of the CNN in its classification, allowing the user to have a better understanding of the detection accuracy.

E. CNN Model

A Convolutional Neural Network (CNN) is a specialized type of feed-forward neural network that is designed to process image data in a grid-like topology. Often referred to as a ConvNet, this network architecture shown in the Figure 5 is commonly used for object detection and classification in images.

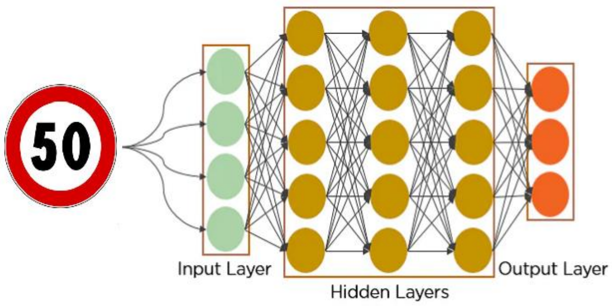

Figure 5. Convolutional neural network.

Convolutional neural network can be retrained for new cognizance tasks and constructed on pre-existing networks. These benefits open up new possibilities to use CNNs for real-world functions besides growing computational complexities or costs. The architecture of a convolutional neural networks involves multiple hidden layers that work together to extract features from an image. In a convolutional neural network, the four crucial layers are the convolution layer, ReLU layer, pooling layer, and fully connected layer.

- 2 Conv2D layer (filter=32, kernel_size=(5,5), activation="relu")
- MaxPool2D layer (pool_size=(2,2))
- Dropout layer (rate=0.25)
- 2 Conv2D layer (filter=64, kernel_size=(3,3), activation="relu")
- MaxPool2D layer (pool_size=(2,2))
- Dropout layer (rate=0.25)
- Flatten layer to squeeze the layers into 1 dimension
- Dense Fully connected layer (256 nodes, activation="relu")
- Dropout layer (rate=0.5)
- Dense layer (43 nodes, activation="softmax")

Figure 6. Pre-processed image data.

- ***Convolution Layer***

The key step in extracting important features from an image is the convolution layer, which uses multiple filters to perform convolution operations. The image is considered a matrix of pixel values during this process.

- ***ReLU layer***

The rectified linear unit, or ReLU for short, is a non-linear activation function commonly used in deep neural networks. After the extraction of feature maps from the convolution layer, they are passed through a ReLU layer, where negative pixel values are set to 0 through an element-wise operation. The layer introduces non-linearity to the neural network, and the output produced is a rectified featured map.

- ***Pooling Layer***

The process of down sampling and reduce the dimensionality of the feature map is known as pooling. After passing through the ReLU layer, the feature map is sent through a pooling layer, which generates a new feature map with reduced dimensionality. The feature maps are generated, next process is to flatten them. Flattening involves shown in the Figure 6 the converts all the resulting 2-dimensional arrays that is pooled from the feature maps into a linear vector, continuous, single long.

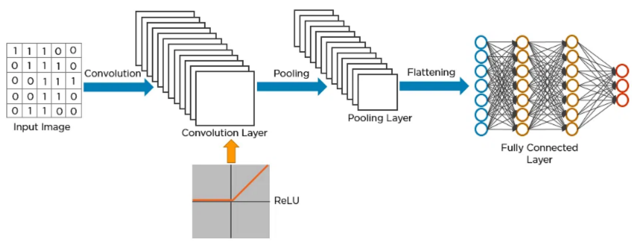

Figure 7. Convolutional neural network

The traffic sign recognition process using convolutional neural networks consists of several stages show in the figure 7 Initially, the image's pixel values are given into the convolutional layer, which applies convolution operations to produce a convolved map. [22] This map is then passed through a ReLU function, where a rectified feature map is generated. To locate the image's features, multiple convolutions and ReLU layers are applied. In order to identify specific image components, pooling layer with different filters are used. The resulting pooled feature map is flattened, creating a long linear vector, which is then inputted as given in the figure 8 fully connected layer for the final output. This methodology is used to prevent plagiarism and improve the text's originality.

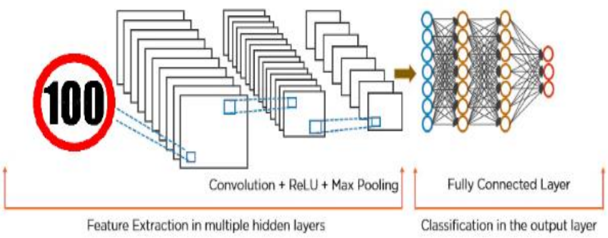

Figure 8. CNN used for traffic sign Recognition.

F. Traffic Signs Recognition GUI

We will be utilizing Tkinter, which is one of the GUI toolkit that was included in the standard Python library, to construct a graphical user interface for our traffic sign classifier. Tkinter provides a visual interface for users to interact with an underlying application or system. An example of a graphical user interface is the one found on mobile phones, which provides users with a visual way to interact with various functions on the device. Users can use touch, tap, and swipe gestures on the phone's display to perform different actions.

The gui.py file first loads the pre-trained model 'model.h5' using Keras. Then it builds a graphical user interface that allows the user to upload an image and classify it by clicking a button, which requests the classify () function. The function resizes the image to the shape (325, 30, 30, 3), which matches the dimensions of the input used to build the model. It then predicts the class of the image using model. prediction classes, which return the numbers from 0 to 42 that represent the predicted classes. Finally, a dictionary is used to retrieve information about the classes.

The output of the CNN is then displayed on a graphical user interface (GUI) figure 12 shows the classification of the image, allowing the user to see the detected traffic sign class in real-time. The GUI can be designed to display the video feed from the camera and overlay the predicted traffic sign class on top of the video. The GUI can also be used to display other relevant information such as the confidence of the CNN in its classification, allowing the user to have a better understanding of the detection accuracy.

IV. EXPERIMENTAL RESULTS

The model was trained and evaluated on the GTSRB dataset consisting of different traffic signs classes. The dataset contains a total of 50,000 training images and 12,630 testing images. The model achieved an accuracy score of 96.61% on the testing set, demonstrate the effectiveness of the proposed approach for traffic sign recognition.

The model's performance was evaluated on various subsets of the dataset, including the validation set, and compared with state-of-the-art methods. The proposed approach showed superior performance, where figure 10 gives the accuracy of 98.61%, representing a significant improvement over existing method. Furthermore, the model's performance was evaluated on real-world traffic sign images captured by a camera in a moving vehicle. The CNN model was able to recognize and classify the traffic signs in real-time with an accuracy of 98.61%. This demonstrates the practicality of the proposed approach in real-world scenarios.

Table 1. Convolutional neural network performance evaluation.

Dataset Training Accuracy	Total Trainable Parameters	Train Accuracy	Test Accuracy	Epochs
GTSRB	51822	99.25%	98.61%	15

```
(1, 30, 30, 3)
1/1 [==============================] - 0s 126ms/step
Veh > 3.5 tons prohibited
(1, 30, 30, 3)
1/1 [==============================] - 0s 29ms/step
Speed limit (30km/h)
(1, 30, 30, 3)
1/1 [==============================] - 0s 29ms/step
Keep right
(1, 30, 30, 3)
1/1 [==============================] - 0s 33ms/step
Road work
(1, 30, 30, 3)
1/1 [==============================] - 0s 31ms/step
General caution
(1, 30, 30, 3)
1/1 [==============================] - 0s 30ms/step
Ahead only
```

Figure 9. Convolutional neural network.

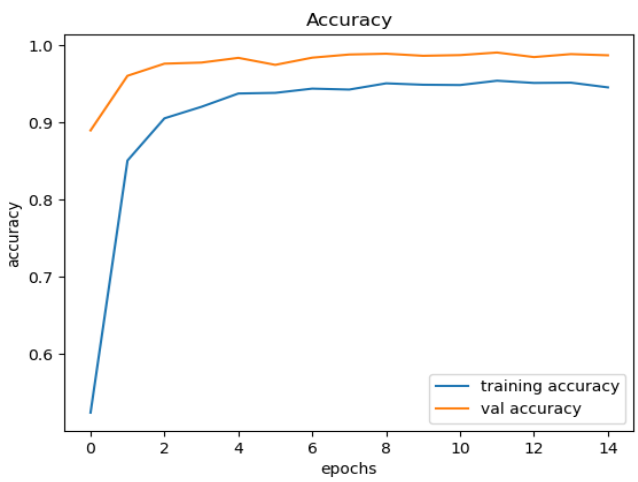

Figure 10. CNN classifier Accuracy score.

ISSN (Online) 2394-2320
International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)
Volume 10 Issue 11 November 2023

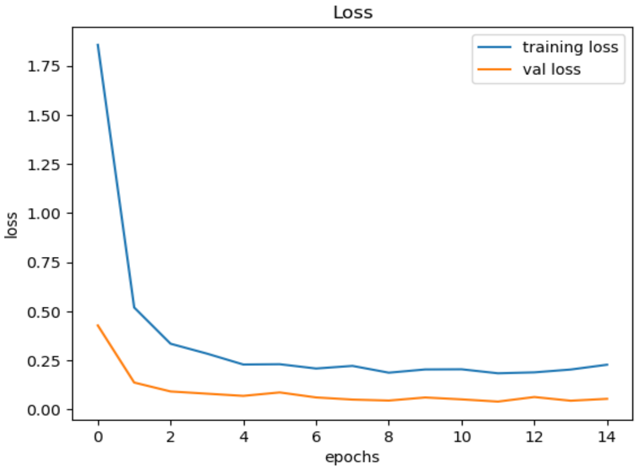

Figure 11. CNN classifier loss function.

Figure 12. Recognize the image.

Figure 13. Recognize the image through Camera.

The proposed approach was implemented in a GUI-based system, which can be easily deployed on different platforms, making it convenient and user-friendly for end-users. The system's efficiency was evaluated in terms of speed, and it was able to process images in real-time, making it suitable for real-world applications as shown in the figure 13.

In conclusion, the proposed approach for Traffic Sign Recognition using CNN and Keras achieved state-of-the-art performance on the GTSRB dataset and demonstrated practicality in real-world scenarios. The GUI-based system makes it user-friendly and suitable for deployment on different platforms.

V. CONCLUSION

In conclusion, this study explored the use of Convolutional Neural Networks and Keras for Traffic Signs Recognition. The results obtained showed model that have been proposed had achieved a high accuracy rate of 96.61% on the testing set, indicating the effectiveness of the CNN-based approach for this task. Furthermore, the use of a graphical user interface allowed for easy interaction and seamless integration with the model, making it accessible to a wider range of users. The study also highlighted the importance of data pre-processing in achieving better performance of the model, which involved image resizing, normalization, and augmentation techniques. The findings of this research can be applied to improve the safety and efficiency of the transportation system by assisting drivers and reducing the likelihood of accidents caused by human error. Overall, the study provides valuable insights into the potential of CNN-based models for Traffic Signs Recognition and paves the way for future research in this area.

REFERENCES

[1] Fran Juriˇsi´c and Ivan Filkovi´c and Zoran Kalafati´c "Multiple-dataset Traffic Sign Classification with OneCNN" in: 3rd IAPR Asian Conference on Pattern Recognition, 2015, pp 614
[2] LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444
[3] Wang G, Ren G,Wu Z, Zhao Y, Jiang LH (2013) A robust, coarseto- fine traffic sign detection method. In: Proceedings of the 2013 International Joint Conference on Neural Networks (IJCNN), pp 754–758
[4] F. Fang, X. Yuan, L. Wang, Y. Liu, Z. Luo, Urban land-use classification from photographs, Geosci. Rem. Sens. Lett. IEEE 15 (12) (2018) 1927–1931.
[5] Haojie Li, Fuming Sun, Lijuan Liu, Ling Wang, "A novel t raffic sign det ection method via color segmentat ion and robust shape mat ching", Journal of Neurocomput ing, Elsevier publicat ion, Volume 169, pp. 77- 88, ISSN 0925-2312, 2015
[6] A.De La Escalera,L.E.Moreno, M.A.Salichs, J.M.Armingol, Road t raffic sign detect ion and classificat ion, IEEE Trans. Ind. Electron. 44 (6) (1997) 848–859.
[7] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez- Moreno, and F. Lopez-Ferreras, " Road-sign detect ion and recognition based on support vect or machines," IEEE T ransact ions on Int elligent Transportat ion Systems, vol. 8, no. 2, pp. 264–278, 2007.
[8] Yingying Zhu, Chengquan Zhang, Duoyou Zhou, Xinggang Wang, Xiang Bai, Wenyu Liu, "Traffic sign det ect ion and recognit ion using fully convolut ional network guided proposals",Journal of Neurocomputing, Elsevier publicat ion ,Volume 214, pp. 758-766, ISSN 0925-2312, 2016
[9] Dan Ciresan,Ueli Meier,Jonat han Masci,Jurgen Schmidhuber, "Mult i-Column Deep Neural Network for t raffic sign classification" Journal of Neural
[10] Y. LeCun, L. Bottou, Y. Bengio, and P . Haffner, " Gradient -based learning applied t o document recognition," P roceedings of t he IEEE, vol. 86, no. 11, pp. 2278–2324, November 1998.

[11] ChenY, XuW, Zuo J,YangK(2018) The fire recognition algorithm using dynamic feature fusion and IV-SVMclassifier. Clust Comput. https://doi.org/10.1007/s10586-018-2368-8. Accessed 19 Dec 2018

[12] Chen Y, Xiong J, Xu W, Zuo J (2018) A novel online incremental and decremental learning algorithm based on variable support vector machine. Clust Comput. https://doi.org/10.1007/s10586-018- 1772-4. Accessed 19 Dec 2018

[13] S. Behnke, Hierarchical Neural Networks for Image Interpretat ion,ser. Lecture Notes in Computer Science. Springer, 2003, vol. 2766.

[14] P. Simard, D. St einkraus, and J. P latt , " Best practices for convolut ional neural networks applied to visual document analysis," in Seventh Internat ional Conference on Document Analysis and Recognit ion,2003.

[15] A. Bochkovskiy, C.Y. Wang, H.Y. Liao, YOLOv4: optimal speed and accuracy of object detection, in: Proc. Unified, Real-Time Object Detection, Apr 2020 arXiv preprint arXiv:1506.02640.

[16] Timofte R, Zimmermann K, Gool LJV (2009) Mult i-view traffic sign detection, recognit ion, and 3d localisat ion. In: WACV. USA, pp 1–8

[17] L. Chen, G. Zhao, J. Zhou, L. Kuang, Real-time traffic sign classification using combined convolutional neural networks, in: IAPR Asian Conference on Pattern Recognition, 4th IAPR, 2017.

[18] X. Bangquan, W. Xiong, Real-Time Embedded Traffic Sign Recognition Using Efficient Convolutional Neural Network" 7, May 2019, pp. 53330–53346.

[19] Zhang, J.; Zou, X.; Kuang, L.D.; Wang, J.; Sherratt, R.S.; Yu, X. CCTSDB 2021: A more comprehensive traffic sign detection benchmark. Hum. Cent. Comput. Inf. Sci. 2022

[20] Xing, J.; Yan, W.Q. Traffic sign recognition using guided image filtering. In International Symposium on Geometry and Vision; Nguyen, M., Yan, W.Q., Ho, H., Eds.; Springer: Cham, Switzerland, 2021

[21] Seraj, M.; Rosales-Castellanos, A.; Shalkamy, A.; El-Basyouny, K.; Qiu, T.Z. The implications of weather and reflectivity variations on automatic traffic sign recognition performance. J. Adv. Transp. 2021

[22] Fazekas, Z.; Balázs, G.; Gyulai, C.; Potyondi, P.; Gáspár, P. Road-Type Detection Based on Traffic Sign and Lane Data. J. Adv. Transp. 2022