BioGecko Vol 12 Issue 03 2023
ISSN NO: 2230-5807

Traffic Sign Recognition using Deep Learning

C. B.Selva Lakshmi [1], Assistant Professor
Department of CSE,Velammal College of Engineering and Technology,Madurai
cbselak08@gmail.com
Nishanthi V[2]. Mathumitha M.[3] Keerthika R.[4] Isha Parasu B[5]
[2,3,4,5]UG Student,Department of CSE, Velammal College of Engineering and Technology, Madurai, Tamil Nadu

Abstract

Traffic sign recognition (TSR) has been an essential part of driver-assistance systems, which can assist drivers in avoiding a vast number of potential hazards and improve the experience of driving. However, the TSR is a realistic task that is full of constraints, such as visual environment, physical damages, partial occasions, etc. To deal with such constraints, convolutional neural networks (CNN) are widely used to extract the features of traffic signs and classify them into corresponding classes. The system consists of three stages: traffic sign detection, refinement, and classification. The detection and refinement are performed using CNN, while the classification is achieved with a proposed Convolutional Neural Network (CNN) architecture. We introduced the extended version of the German Traffic Sign Detection Benchmark (GTSDB) or GTSRB, labeled in a pixel manner (masks), with 11 classes grouped into 8 categories.

***Keywords* -** Deep learning, Traffic sign recognition - CNN, dataset, traffic signs, traffic sign classification.

I. INTRODUCTION

Nowadays, intelligent autonomous vehicles, together with advanced driver assistance systems (ADAS), deal with the problem of traffic sign recognition. Traffic sign recognition using deep learning is a computer vision project that involves training a model to recognize different traffic signs in real time. In this project, we will use deep learning techniques to train a neural network that can identify traffic signs from images. To start, we need to download the dataset containing images of traffic signs. One of the most popular datasets for traffic sign recognition is the German Traffic Sign Recognition Benchmark (GTSRB) dataset. It contains more than 50,000 images of traffic signs from different angles and lighting conditions. Once we have the dataset, we can start building our neural network. We will be using Python programming language and the TensorFlow library to create our model. The TensorFlow library provides a high-level interface for building deep learning models, making it easier to implement complex neural networks [1]

our model, we will use a convolutional neural network (CNN). CNNs are widely used in computer vision tasks because they can extract features from images and learn to recognize patterns. Our CNN will have several layers, including convolutional layers, pooling layers, and fully connected layers.The first layer of our CNN is the input layer, where we feed the images from the dataset. The next few layers are convolutional layers that apply filters to the input images, extracting important features. The pooling layers are then used to reduce the size of the feature maps and make the model more efficient. Finally, the fully connected layers are used to classify the images based on the extracted features. Once we have built our model, we can start training it on the dataset. We will use the backpropagation algorithm to update the weights of the neural network and minimize the loss function. The loss function measures the difference between the predicted output of the model and the true label. After training, we can test our model on a separate test dataset and evaluate its performance[3] We can use metrics such as accuracy, precision, recall, and F1 score to evaluate the model's performance. Once we are satisfied with the performance of our model, we can use it to classify new traffic sign images in real time. We can use the OpenCV library to capture images from a camera or a video feed and use our model to recognize the traffic signs. In summary, traffic sign recognition using deep learning is a challenging but rewarding project that involves building a convolutional neural network to recognize different traffic signs from images. With the help of Python and TensorFlow, we can create a model that can identify traffic signs in real time, making roads safer for everyone.

II.PROBLEM DEFINITION

Classifying traffic signs is a very important task for autonomous driving systems, as the safety of everyone as well as the passenger depends on it [4-8]. Depending on the country, traffic signs possess a variety in their visual appearance, making it harder for classification systems to succeed. Nowadays, intelligent autonomous vehicles, together with advanced driver assistance systems (ADAS), deal with the problem of traffic sign recognition. It is a challenging real-world computer vision problem due to the different and complex scenarios they are placed into. The proposed system will help understand the problem and provide a systematic way of approaching it.

III. OBJECTIVES

Overall, the ultimate objective of this report is to complete the customized traffic-sign recognition and figure out which state-of-art networks better fit into this project. Firstly, to recognize traffic sign, the first process is where dataset is generated as the quality of the actual data will have an impact on the result of network and models. Next is the tuning process of detectors and classifiers will be introduced, and we will adjust the neural networks from the following : (1) Experiments performed on different algorithms, (2) Validation and evaluation based on the chosen models, (3) Parameter tuning process. Finally, the report will summarize the pros and cons across the applied algorithm systems and find the best system for traffic-sign recognition.

IV.LITERATURE SURVEY

Traffic sign recognition (TSR) has benefited a large number of realistic applications, such as driver assistance systems, autonomous vehicles, and intelligent mobile robots since they have delivered the current state of traffic signs into various systems.

A. Real-Time Traffic Sign Recognition Based on Efficient CNNs in the Wild [9]
Year: Mar. 2018.Author: J. Li and Z. Wang,

Methodology

A complete pipeline of traffic sign recognition is proposed, which is superior to the majority of previous work, considering generality, reliability and run time.

B. A computer vision assisted geoinformation inventory for traffic infrastructure [2]
Year: 2010.Author: S. Segvi ˇ c, K. Brki ´ c, Z. Kalafati ´ c, V. Stanisavljevi ´ c,M. ´ Sevrovi ˇ c,
D. Budimir and I. Dadi ´ c

Methodology

The developed techniques were tested on triangular warning signs from the superclass A. The superclass A is chosen as a model subset due to its prevalence in our videos. All algorithms have been trained on the training set T2009 containing about 2000 triangular signs collected in 2009 using an interlaced camera.

C. Simultaneous Traffic Sign Detection and Boundary Estimation Using Convolutional Neural Network[10]
Year: 2018.Author: Hee Seok Lee and Kang Kim

Methodology

The efficient traffic sign detection method is proposed where locations of traffic signs are estimated together with their precise boundaries. To this end, the object bounding box detection problem is generalized and formulated an object pose estimation problem, and the problem is effectively modeled using CNN based on the recent advances in object detection networks.

D. Towards real-time traffic sign detection and classification
Year: .2016. Author: Yi Yang, Hengliang Luo, Huarong Xu, and Fuchao Wu

Methodology

Aim of this project is to address the problem of real-time traffic sign recognition. To this end, two fast algorithms for traffic sign detection and classification is proposed, respectively.

E. Thai Traffic Sign Detection and Recognition Using Convolutional Neural Networks
Year: .2018.
Author: Mohamed Shahud, Jigyasa Bajracharya

Methodology:

After careful consideration and analysis of cost and benefit, use Yolov3 dark net developed by Joseph Redmon . There are 2 main reasons. First, Yolov3 is really fast, which makes it ideal for use in a real time application such as detecting traffic signs.

V.PROPOSED SYSTEM

We introduced the extended version of the German Traffic Sign Detection Benchmark, labeled in a pixel manner with 42 classes grouped into 8 categories. which is used to design the task of semantic segmentation of traffic signs. It is an extension of the original GTSDB dataset, which contains only bounding box annotations for traffic signs. The extended version of GTSDB with pixel-level annotations is a valuable resource for researchers and practitioners working on the development of computer vision algorithms for traffic sign detection and recognition, as well as for the evaluation and comparison of existing methods. Figure 1 shows the system architecture of the proposed model.

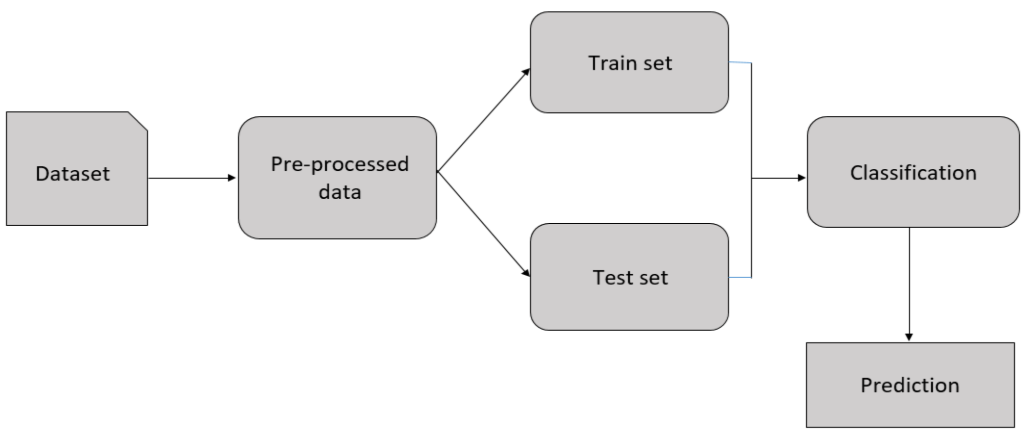


Figure 1. System Architecture

A).DATASETS

The images of our proposed European dataset are composed of publicly available datasets and of sequences recorded in Belfort, France, and surroundings during Spring and Summer from 2014, 2015, and 2018. The sequences are composed of urban and rural environments and cover daytime and sunset conditions. The public datasets(refer figure 2) are composed of different scenarios (urban, rural, highway) mostly captured during the daytime.

Figure 2. Sample Dataset

The flow diagram of the proposed work is shown below in figure 3 through which the entire flow of the proposed model is implemented on the signal dataset.

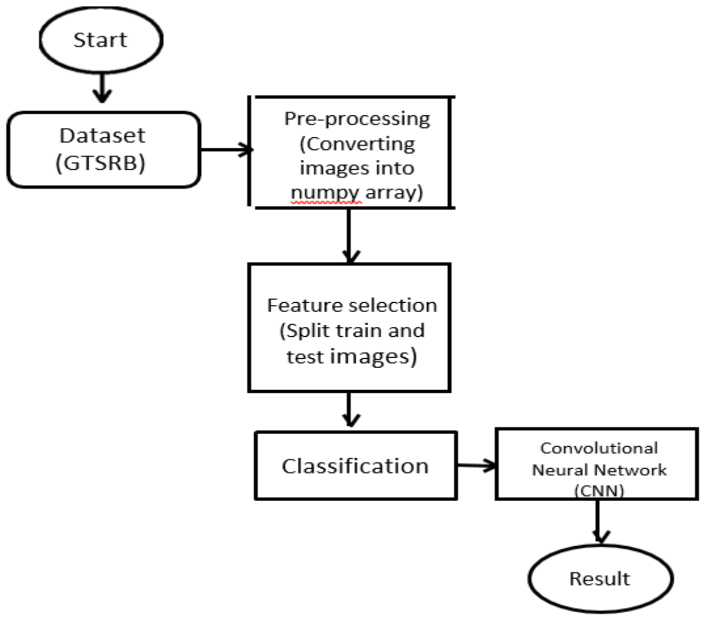


Figure 3. Flow Diagram

B. DATA SELECTION AND LOADING

The data selection is the process used for selecting the German Traffic Sign Recognition Dataset. In this paper, German Traffic Sign images from the GTSRB dataset (refer figure 4) are used to find traffic signs.

Figure 4 GTSRB dataset

C. DATA PREPROCESSING

Image Data pre-processing is the process of getting rescaled data from the dataset.

- Resize image from the dataset
- Getting data

Resize image dataset: Rescale the German Traffic Sign image (from the GTSRB dataset) size into 200.
Getting data: The categorical data are defined as variables with a finite set of rescaled values. Most deep learning algorithms require array input and output variables.

D. FEATURE SELECTION

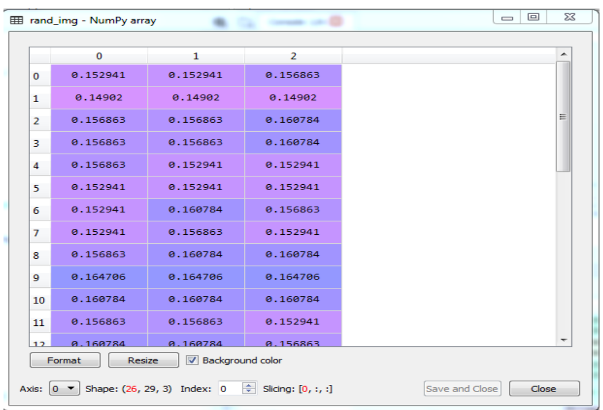
Data splitting is the act of partitioning data from the dataset (GTSRB) into two portions. One Portion of the data is used to develop a predictive model. Another is to evaluate the model's performance. Separating data into training and testing sets is an important part of evaluating data mining models. When you separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is used for testing.

E. CLASSIFICATION

CNN In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "full connectedness" of these networks makes them prone to overfitting data. Typical ways of regularization include adding some form of magnitude measurement of weights to the loss function. CNNs take a different approach towards regularization: they take advantage of the hierarchical pattern in data and assemble more complex patterns using smaller and simpler patterns. Therefore, on the scale of connectedness and complexity, CNNs are on the lower extremity.

F. PREDICTION

It is a process of predicting the GTSRB from the dataset. This project will predict the data by enhancing the performance of the overall prediction results (refer figure 5 (a) & (b)). The Final Result will get generated based on the overall classification and prediction.



	0	1	2
0	0.152941	0.152941	0.156863
1	0.14902	0.14902	0.14902
2	0.156863	0.156863	0.160784
3	0.156863	0.156863	0.160784
4	0.156863	0.152941	0.152941
5	0.152941	0.152941	0.152941
6	0.152941	0.160784	0.156863
7	0.152941	0.156863	0.152941
8	0.156863	0.160784	0.160784
9	0.164706	0.164706	0.164706
10	0.160784	0.160784	0.160784
11	0.156863	0.156863	0.152941
12	0.160784	0.160784	0.156863

Figure 5 (a). Predicted Results

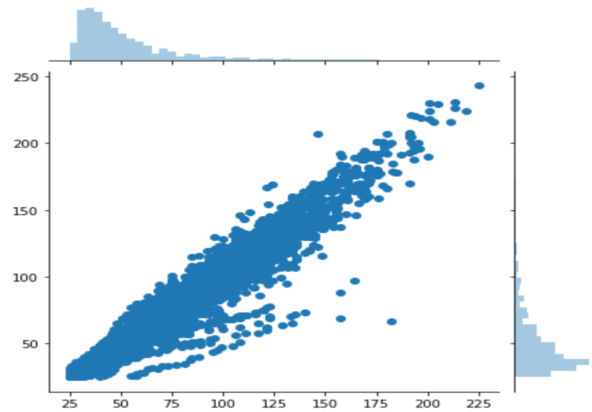


Figure 5 (b). Predicted Results for different values

VI.CONCLUSION

In this work, Traffic light detection in a European urban environment. The traffic sign digital images are used as the input of the data. The sign images are fed into the pre-processing method. In the pre-processing method, all the input digital images are resized. Then it is processed into the feature selection method. In the feature selection method, the digital image data set is split into training and testing datasets. The classification deep learning algorithm is used to predict the traffic sign images. Convolutional Neural Network is used to increase the overall performance of the detection of traffic sign detection also to improve the accuracy of the system.

In the future, the traffic sign detection method will be applied to vehicles. And also it will detect the traffic light signs and signal in distance measurement. It will also help to detect the signs and control the vehicle using a sensor device. Our process is detection the signs in the form of images and their pixels. It will help to avoid accidents and make them aware of traffic rules.

REFERENCES

1. C. Grigorescu and N. Petkov, "Distance sets for shape filters and shape recognition," IEEE Trans. Image Process., vol. 12, no. 10, pp. 1274–1286, Oct. 2003
2. S. Šegvic et al., "A computer vision assisted geoinformation inventory for traffic infrastructure," in Proc. 13th Int. IEEE Conference Intell. Transport System (ITSC), Sep. 2010,
3. F. Larsson and M. Fels berg, "Using Fourier descriptors and spatial models for traffic sign recognition," at Proc. Scand. Conf. Image Anal. Berlin, Germany: Springer, 2011,
4. R. Timofte and L. Van Gool, "Sparse representation-based projections," in Proc. 22nd Brit. Mach. Vis. Conf. (BMVC), 2011.
5. Mogelmose, M. M. Trivedi, and T. B. Moeslund, "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," IEEE Trans. Intell. Transp. Syst., vol. 13, no. 4, Dec. 2012.
6. S. Hou ben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German traffic sign detection benchmark," in Proc. Int. Joint Conf. Neural Netw. (IJCNN), Aug. 2013, pp. 1–8.
7. G. Serna and Y. Rui chek, "Classification of traffic signs: The European dataset," IEEE Access in 2018.
8. H. Luo, Y. Yang, B. Tong, F. Wu, and B. Fan, "Traffic sign recognition using a multi-task convolutional neural network," IEEE Trans. Intell. Transp. Syst., vol. 19, no. 4, , Apr. 2018.
9. J. Li and Z. Wang, "Real-time traffic sign recognition based on efficient CNNs in the wild," IEEE Trans. Intell. Transp. Syst., vol. 20, no. 3, Mar. 2018.

10. H. S. Lee and K. Kim, “Simultaneous traffic sign detection & boundary estimation using convolutional neural network,” IEEE Trans. Intell. Transp. Syst., vol. 19, no. 5, May 2018.