International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:14, No:10, 2020

Intelligent Transport System: Classification of Traffic Signs Using Deep Neural Networks in Real Time

Anukriti Kumar, Tanmay Singh, Dinesh Kumar Vishwakarma

***Abstract*—**Traffic control has been one of the most common and irritating problems since the time automobiles have hit the roads. Problems like traffic congestion have led to a significant time burden around the world and one significant solution to these problems can be the proper implementation of the Intelligent Transport System (ITS). It involves the integration of various tools like smart sensors, artificial intelligence, position technologies and mobile data services to manage traffic flow, reduce congestion and enhance driver's ability to avoid accidents during adverse weather. Road and traffic signs' recognition is an emerging field of research in ITS. Classification problem of traffic signs needs to be solved as it is a major step in our journey towards building semi-autonomous/autonomous driving systems. The purpose of this work focuses on implementing an approach to solve the problem of traffic sign classification by developing a Convolutional Neural Network (CNN) classifier using the GTSRB (German Traffic Sign Recognition Benchmark) dataset. Rather than using hand-crafted features, our model addresses the concern of exploding huge parameters and data method augmentations. Our model achieved an accuracy of around 97.6% which is comparable to various state-of-the-art architectures.

***Keywords*—**Multiclass classification, convolution neural network, OpenCV, Data Augmentation.

I. Introduction

WITH the rapid increase in population, people are aiming to look for various alternatives to lead a comfortable and easy life. Self-Driving car technology is one such development towards this goal and is one of the newest inventions in the transportation system. Almost every day, new advancements in the field of driverless car technologies are taking place. However, self-driving cars are not yet legal on most of the roads. Although some companies have got permission for testing this technology, running a self-driving car is still illegal in almost all countries. According to DOT [15] which is a US Department of Transportation and NHTSA, around 10,000 people lost their lives in 2019 due to motor vehicle traffic accidents. It also estimated that around 94% of the serious crashes are due to human error only including drunk or distracted driving cases. One of the biggest advantages of autonomous systems as these cars is that they remove such risk factors. There are still various challenges such as mechanical issues that can cause crashes. They must know how to identify traffic signs, other vehicles, branches and other various countless objects in the vehicle's path. Based on this identification, the system must make certain decisions to avoid fatal risks and accidents by taking instantaneous actions like slowing down of the vehicle or control acceleration.

Traffic sign detection and recognition is one of the most important fields in the ITS. Based on the visual impact of traffic signs, self-driving cars can act accordingly and thus, automatic recognition can avoid accidents and dangers. For this problem, our paper presents a CNN based architecture widely for providing high performance in image-based detection tasks. The dataset used for solving this problem is German Traffic Sign Recognition Benchmark (GTSRB) which is a multi-class traffic sign images classification dataset having around 50,000 images of various noise levels. There are several reasons for preferring this model over other state of the art techniques already available. Through dataset analysis, it was observed that it consists of various challenges for which if other techniques or statistical approaches to denoising are applied, it can be computationally very expensive and hence, it is highly unsuitable for real time applications. On the other hand, neural network-based detection and classification of the noise is computationally effective as well as achieves high performance as far as accuracy and efficiency is concerned.

The paper is organized as follows: Section II explains the attempts done for a similar task. Section III presents the methodology used in this paper for the detection as well as classification of traffic signs. Section IV describes the evaluation metrics used for this task and the results obtained by our methodology. Section V refers to the conclusion and discussion of possible extensions of this research.

II. Related Work

A lot of work has already been done in detection and classification of traffic signs for future autonomous vehicle technology. Various Conv Nets based approaches have been used for this task some of them are described here. In [1] You Only Look Once (YOLOv2), Single Shot Detector (SSD) and Faster Region CNN (Faster RCNN) [2] deep learning architectures along with pretrained CNN models were compared for traffic sign detection and classification task. Various pretrained CNN models already trained on ImageNet Dataset were used, YOLO v2 combined with Coco CNN model, SSD combined with Inception V2 and Faster RCNN combined with ResNet pretrained CNN models were analysed on the GTSRB dataset. The evaluation parameter used was Mean Average Precision (mAp) and Frames Per Second (FPS). On comparison, it was found out that YOLO is more accurate and faster than SSD and Faster RCNN.

Anukriti Kumar, Tanmay Singh, and Dinesh Kumar Vishwakarma are with the Department of Information Technology, Delhi Technological University, New Delhi, India (e-mail: anu1999kriti@gmail.com, tanmaysingh60@gmail.com, dinesh@dtu.ac.in).

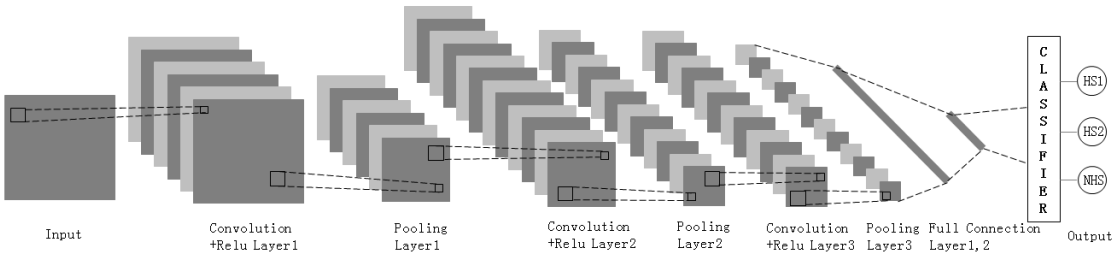

Fig. 1 CNN

Another paper [3] proposed a novel and challenging approach of extending SSD algorithm for the traffic sign detection and identification algorithm. During the preprocessing phase the images were normalized and fed to the VGG-16 front-end framework of the SSD algorithm. The proposed model composed of five stacked convolution layers, three fully connected layers and a softmax layer. Using a learning rate of 0.001 and batch size of 50 and 20 for train and validation set respectively, an accuracy rate of 96% was achieved after 20,000 iterations.

Changzhen et al. [4] proposed deep CNNs that were based on Chinese Traffic Sign Detection Algorithm using Faster-RCNN's region proposal network. There are seven categories of traffic signs in China and the dataset consisting of images from the internet and road-side scenes from China. The data were augmented by motion blur and applied several levels of brightness on those images. Three different models were trained namely VGG16, VGG_CNN_M_1024 and ZF. The ZF model had the highest detection efficiency with 60 ms as the average detection time. The model was tested on 33 video sequences captured using a mobile phone and on-board camera. The detection rate of the proposed algorithm was in real-time with an efficiency of around 99%.

In [5], Mao proposed a clustering algorithm based on CNN that was used to separate categories into k different subsets or families. After this, hierarchical CNN was used to train k+1 classification CNNs, out of which one was for family classification and other k CNNs, corresponding to each family that although achieved 99.67% accuracy, this model was still computationally very expensive. Another research [7] held by Qian proposed an effective feature for the classification task by using max pooling positions. They showed how MPP is a better feature through various experiments which indicated that MPPs demonstrate the desirable characteristics of large intraclass variance and small inter-class variance in general but did not improve accuracy further.

Another research team [7] proposed a CNN-ELM model, which integrated the feature learning capacity of CNNs with extreme learning machine (ELM) [8] because of their amazing generalization performance. In this model, firstly CNN was used for learning features and these features were then fed into the fully connected layers that were replaced by ELM for classification. The proposed model trained on GTSRB dataset achieved an accuracy of 99.4% but could not surpass the results of state-of-the-art algorithms.

In [9], Cireşan developed a model by combining 25 different CNNs having three convolutional layers and two fully connected layers that could learn more than 88 million parameters. Although it achieved an accuracy of 99.46%, one of the biggest disadvantages of this model was that it used image augmentation due to which a reliable classification accuracy cannot be ensured for unknown data in general.

Dan Cireşan et al. [10] proposed a nine-layer CNN along with seven hidden layers consisting of an input layer, three convolutional layers, three maxpooling layers followed by two fully connected layers. In the pre-processing of the data, they cropped the images to equal size. Three different contrast normalization techniques were used in order to reduce high contrast variation in pictures. A greyscale representation of the original images was also produced and the model was trained on 8 different datasets comprising of original as well as sets resulting from three different contrast normalizations of color and grayscale images. Before every epoch in the training phase, images were translated, rotated and scaled based on a uniform distribution over a specified range. A recognition rate of 98.73% was achieved using CNN and a combination of MLP and CNN achieved a 99.15% recognition rate. Both the models misclassified the 'no vehicle' traffic sign. In another paper [11] a CNN based approach for aforementioned task under adverse environmental conditions is proposed. Various types of challenges are faced during real-time image capture on the road such as blur, decolorization, shadow, haze, exposure, rain, dirty lens and noise. A CNN model similar to VGG-16 architecture was used to detect the type of challenge present in the image. Contrast Limited Adaptive Histogram Equalization (CLAHE) method is used to prevent enhancement of noise and contrast normalization. A CNN with ResNet type architecture is used to remove rain from the images. A deep CNN with U-Net architecture is used to localize traffic signs from the preprocessed image. Finally, for the classification task, two convolution blocks, each with two convolution layers, a ReLU activation layer followed by a Maxpool layer, with a dropout layer added after each block was used. The proposed model was trained and evaluated on the IEEE VIP Cup 2017 video dataset with an overall precision and recall of 62% and 42% respectively.

III. Research Methodology

In this section, we aim to discuss in detail our approach to the proposed CNN model. Section III *A* refers to the dataset description of GTSRB and its associated statistics. Section III *B* highlights the challenges faced with analyzing the dataset.

Section III *C* deals with the pre-processing phase overcoming the challenges mentioned previously. Section III *D* discusses the architecture used and the hyperparameters involved in it.

A. Dataset Description

The dataset used for training is generated at the International Joint Conference on Neural Networks (IJCNN) in 2011 for the German Traffic Sign Benchmark challenge inviting researchers to participate even without some specific domain challenge. This image dataset consists of around 43 classes representing unique traffic sign images. Training set has around 34,799 images (around 67.12 %), test set has 12,630 images (around 24.38 %) and validation set has 4410 images (8.5 %).

TABLE I
DATA STATISTICS

	No. of images	Percentage of the dataset
Training Set	34799	67.12
Validation Set	12630	24.38
Test Set	4410	8.5

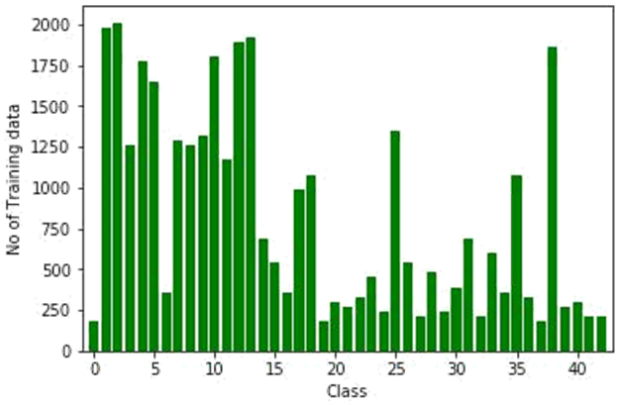

Fig. 2 Class distribution of the training set

B. Data Preprocessing

1. Challenges Faced

i. Low Image Contrast

It can occur due to several factors such as a limited range of sensor sensitivity, bad sensor transmission function and so on. This can be detected by plotting brightness histograms with the values varying from black to white on horizontal axis and number of pixels (absolute or normalized) on the vertical axis. Low image contrast will be observed if either this brightness range given is not fully used or the brightness values are concentrated around certain areas only.

ii. Imbalanced Data

As observed from Fig. 2, the data are highly imbalanced as there exists a disproportionate ratio of images in each unique class of traffic sign. Some classes seem to have a lesser number of images than the other which causes class bias as some classes then remain underrepresented. There are several approaches to resolve this issue including resampling techniques such as oversampling the minority class or undersampling the majority class, generating synthetic samples or changing the performance metric of the algorithm.

2. Preprocessing Phase

It aims at solving the mentioned challenges obtained from the dataset by applying various techniques.

i. Data Augmentation

Data Augmentation refers to accepting some training images in the form of batches, applying various random transformations to each image present in the batch including random rotation, changes in scale, translations, shearing, horizontal or vertical flips, replacing the original batch with the newly transformed one and finally training the CNN on this new dataset. This is done in order to recognize the target object more effectively as it increases the generalizability of our classifier. Although their appearances might change a bit, still their class labels will remain the same.

OpenCV is used for this task which is a library developed by Intel aimed at real-time computer vision [12]. It is used for performing various image processing operations such as rotation, transformation, translation and so on.

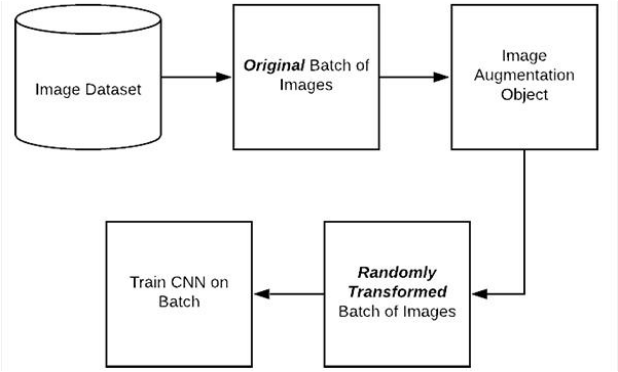

Fig. 3 Data Augmentation

1. Rotation: Images are rotated slightly at around 10 degrees to improve performance [13]. More rotation might cause incorrect recognition.

Fig. 4 Rotation of images

2. Translation: It will move every point in an image by some constant distance in a particular direction. It can also be considered as shifting the origin of the entire coordinate system. Here, this translation shifted the image slightly in downward direction.

Fig. 5 Translation of image downwards

3. Bilateral Filtering: It is similar to blurring but the key difference between them is that blurring smoothens edges whereas a bilateral filter can keep the image's edges sharp while working on noise reduction. Hence, it is preferred here.
4. Gray Scaling: It is performed so that less information is provided for each pixel that reduces complexity in comparison to a colored image.
5. Local Histogram Equalization: It is applied to increase image contrast.

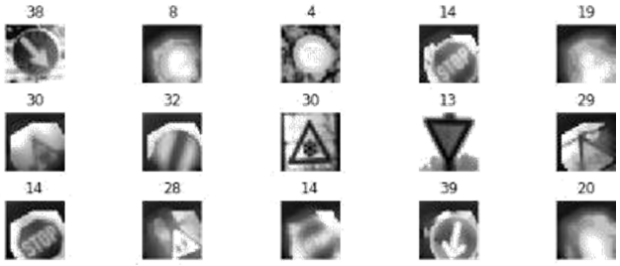

Fig. 6 Images after data augmentation

ii. Class Bias Fixing

To remove the class bias problem, all the classes or unique traffic signs are made to have the same number of image samples which is an arbitrary number that can be obtained on analyzing the dataset from Fig. 2. On observation, a maximum number of records belong to class 2, which are around 2010 records. So, the arbitrary number can be taken as around 4000 which is around twice of 2010.

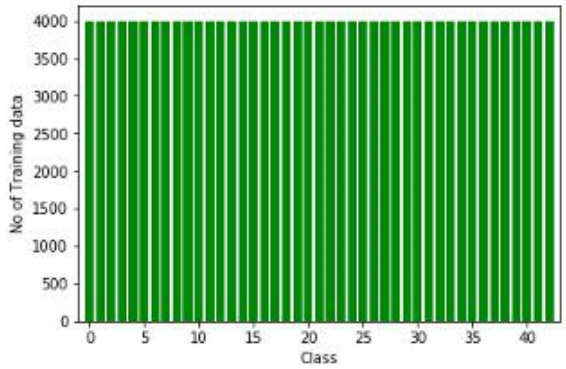

Fig. 7 Class distribution after fixing class bias issue

C. Activation Function

Activation function is an important part of neural networks as they determine whether information received by a neuron is relevant or should be ignored. It is the non-linear transformation which is done over the input signal and its output is then sent to the next layer as input. They are crucial as without them, backpropagation process is not even possible.

1. ReLU

One of the most commonly used activation functions is ReLU that is Rectified linear unit. It is defined as:

$$ReLU = \max(0, x)$$

One of the biggest advantages of ReLU is that it is non-linear which makes backpropagation of errors possible and we can have multiple neuron layers activated by ReLU. Also, at a time it activates only a few neurons making the network more efficient as well as easy for computation.

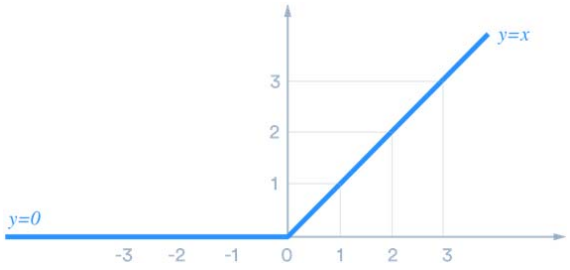

Fig. 8 ReLU Activation Function

2. Softmax

Softmax function is another activation function that we mainly use in handling classification problems. It is applied over the final layer of the network which tells how much it is confident in its prediction. This is mainly done by performing two calculations, first exponentiating values received at each node and then normalizing this value by summing up these exponentiated values. The vector returned by the softmax function yields probability scores for each class label since they are easier for interpretation. It is represented by:

$$Softmax(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

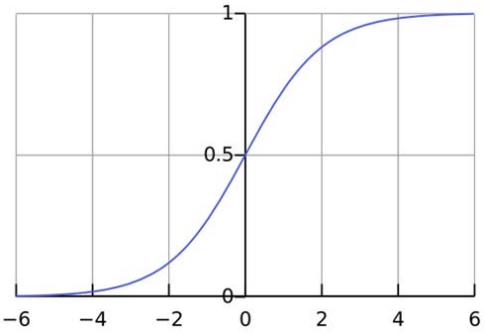

Fig. 9 Softmax Activation Function

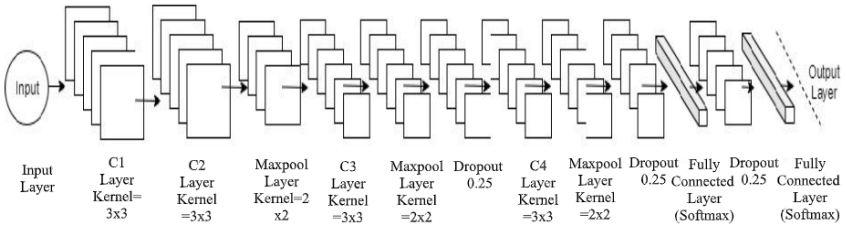

Fig. 10 Model Architecture

D. Model Architecture

The CNN architectures are used mostly in image processing applications as it involves processing just like our human brain. They are preferred over feed-forward neural nets as they are capable of capturing the temporal as well as special dependencies as well. In our model, we have built the deep learning model for classifying unlabelled traffic signs using CNN model architecture comprising of 4 convolution layers and max-pooling layers. The kernel size is chosen as (3, 3) for these convolutional layers. First, the convolution layer takes an image as input for processing with its shape as (32, 32, 1) as the channels have been pre-processed into grayscale images. In order to reduce the training time and overfitting, Max pooling layers are then added. Then, two fully connected layers are added which require a one-dimensional vector as input for which flattening is done. In the output layer, we have used the softmax activation function as it is a multi-class classification problem [14]. The model architecture is shown in Fig. 10. This model is run with 700 epochs with GPU for faster processing.

TABLE II
CNN Parameters

Layer No.	Layer Type	Hyperparameters	Kernel Size
0	Input	32 x 32 x 1	
1	Convolution (ReLU)	32 neurons	3 x 3
2	Convolution (ReLU)	64 neurons	3 x 3
3	Max Pooling		2 x 2
4	Convolution (ReLU)	64 neurons	3 x 3
5	Max Pooling		2 x 2
6	Dropout	Dropout rate= 0.25	
7	Convolution (ReLU)	128 neurons	3 x 3
8	Max Pooling		2 x 2
9	Dropout	Dropout rate= 0.5	
10	Flatten	128 neurons	
11	Fully Connected (Softmax)		
12	Dropout	Dropout rate= 0.5	
13	Fully Connected (Softmax)		

IV. Evaluation and Results

Accuracy was chosen as the evaluation metric in the German Traffic Sign Benchmark challenge. Our model was tested on the validation data and performance results were analyzed with the help of confusion matrix which in simpler terms can be described as a table depicting its confusion while making predictions, also summarizing the performance of any model.

With the help of this confusion matrix obtained, Accuracy can be obtained as:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

where TP: True Positive that is the observation is positive and prediction is also positive, FN: False Negative that is the observation is positive but the prediction is negative, TN: True Negative that is the observation is negative and prediction is also negative, FP: False Positive that is the observation is negative but prediction is positive.

Using the proposed model, we have been able to reach a very high accuracy rate of around 97.6 %. We also observed that our model starts saturating after 10 epochs. The number of epochs can also be reduced to 10 for decreasing the computation cost.

V. Conclusion and Future Work

In this paper, we developed a CNN architecture for the classification of unique traffic signs for self-driving car technology. We used OpenCV for image augmentation techniques for improving the model performance and it is also suitable for real-time applications since it involves low computation at every point. For future work, we aim to identify the best architecture along with the best hyperparameters and train our proposed model on a larger dataset. We can try some other pre-processing techniques to improve the model's accuracy. We can make it a more generalist system by first using a CNN to localize the traffic signs in realistic scenes and another one to classify them. We can also try some different architectures such as AlexNet or VGGNet and compare their performances.

References

[1] Priya Garg, Debapriyo Roy Chowdhury and Vidya N. More, "Traffic Sign Recognition and Classification Using YOLOv2, Faster RCNN and SSD," In: 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT) 2019, December 2019.

[2] Evan Peng, Feng Chen, and Xinkai Song, "Traffic Sign Detection with Convolutional Neural Networks," In International Conference on Cognitive Systems and Signal Processing, July 2017.

[3] Wang Canyong, "Research and Application of Traffic Sign Detection and Recognition based on Deep Learning," In: International Conference on Robots & Intelligent System (ICRIS) 2018, May 2018.

[4] Xiong Changzhen, Wang Cong, Ma Weixin, Shan Yanmei, "A Traffic Sign Detection Algorithm Based on Deep Convolutional Neural Network," In: IEEE International Conference on Signal and Image Processing (ICSIP) 2016, March 2017.

[5] Mao, X., Hijazi, S., Casas, R., Kaul, P., Kumar, R., Rowen, C.: Hierarchical CNN for traffic sign recognition. In: Intelligent Vehicles Symposium (IV), 2016 IEEE, pp. 130–135. IEEE (2016).

[6] Qian, R., Yue, Y., Coenen, F., Zhang, B.: Traffic sign recognition with convolutional neural network based on max pooling positions. In: 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), pp. 578–582. IEEE (2016).
[7] Zeng, Y., Xu, X., Fang, Y., Zhao, K.: Traffic sign recognition using extreme learning classifier with deep convolutional features. In: The 2015 International Conference on Intelligence Science and Big Data Engineering (IScIDE 2015), Suzhou, China, vol. 9242, pp. 272–280 (2015).
[8] Zeng, Y., Xu, X., Fang, Y., & Zhao, K. (2015). Traffic sign recognition using deep convolutional networks and extreme learning machine. In Intelligence science and big data engineering. image and video data engineering (IScIDE) (pp. 272–280). Springer. doi:10.1007/ 978-3-319-23989-7_28.
[9] Dan Cireşan, Ueli Meier, Jonathan Masci, and Jürgen Schmidhuber, "Multi-column deep neural network for traffic sign classification," In Neural Networks, Elsevier, August 2012.
[10] Ciresan, D., Meier, U., Masci, J., Schmidhuber, J., "A committee of neural networks for traffic sign classification," In: Dalle Molle Institute for Artificial Intelligence (2011).
[11] Uday Kamal, Sowmitra Das, Abid Abrar, and Md. Kamrul Hasan, "Traffic-Sign Detection and Classification Under Challenging Conditions: A Deep Neural Network Based Approach," In: IEEE Video and Image Processing Cup 2017, September 2017.
[12] Hamed Habibi Aghdam, Elnaz Jahani Heravi, Domenec Puig, "A Practical and Highly Optimized Convolutional Neural Network for Classifying Traffic Signs in Real-Time," In International Journal of Computer Vision, September 2016.
[13] Fleyeh, H., & Davami, E. (2011). Eigen-based traffic sign recognition. IET Intelligent Transport Systems, 5(3), 190. doi:10.1049/iet-its. 2010.0159.
[14] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. (2011). The GermanTraffic Sign Recognition Benchmark: A multi-class classification com-petition. In: International Joint Conference on Neural Networks.
[15] National Highway Traffic Safety Administration, Traffic Safety Facts: Alcohol-Impaired Driving, 2019 Data, DOT HS 812 360, 2019