IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

TRAFFIC SIGN BOARD DETECTION BY SELF DRIVING CARS USING TRANSFER LEARNING

Yashwanth Telukuntla[1#], Varun Totakura[2#], V. Raghavendra Goud[3#], V. Sai Charan Goud[4#], Vedha Sai Telukuntla[5]

[#]*Guru Nanak Institutions Technical Campus, Hyderabad, India,*

[5]*Amritha School of Engineering, Bangalore, India,*

***Abstract*—** You must have heard about the self-driving cars in which the passenger can fully depend on the car for traveling. But to achieve level 5 autonomous, it is necessary for vehicles to understand and follow all traffic rules. In the world of Artificial Intelligence and advancement in technologies, many researchers and big companies like Tesla, Uber, Google, Mercedes-Benz, Toyota, Ford, Audi, etc are working on autonomous vehicles and selfdriving cars. So, for achieving accuracy in this technology, the vehicles should be able to interpret traffic signs and make decisions accordingly. There are several different types of traffic signs like speed limits, no entry, traffic signals, turn left or right, children crossing, no passing of heavy vehicles, etc. Traffic signs classification is the process of identifying which class a traffic sign belongs to. In our project we used ResNet-50 architecture with German Traffic Sign Database is chosen and augmented by data pre-processing technique. Subsequently the layer-wise features extracted using different convolution and pooling operations are compared and analysed. Finally transfer learning-based model is repetitively retrained several times with fine-tuning parameters at different learning rate, and excellent reliability and repeatability are observed based on statistical analysis.

***Keywords*—** Self Driving Car, Traffic, Board, Vehicle, CNN, ResNet.

I. INTRODUCTION

Nowadays, Intelligent Autonomous Vehicles together with Advanced Driver Assistance Systems (ADAS) deal with the problem of traffic sign recognition. It is a challenging realworld computer vision problem due to the different and com-plex scenarios they are placed into. Some of the hard conditions include: illumination changes, occlusions, perspectives, weather conditions, aging and human artifacts to name a few. Therefore and because of the high industrial demand for autonomous vehicles, many studies have been published together with datasets from all over the world [1] [2]. How-ever, the systems are limited to the country and/or certain types of signs (shape, category). Traffic signs provide crucial visual information in order to understand the proper driving conditions [3]. For example, they inform about speed limits, drivable lanes, obstacles, temporary situations, roadway access, restrictive areas, etc. Reasons why they are designed to be easily detectable, recognizable and interpretable by humans [4]. Standard shapes, colours, pictographs and text are used to de ne a meaning. Nevertheless and besides the efforts to standardize traffic signs [5], there exists inter and intra variability between countries and between classes for specific traffic signs. For example, the inter variability is mostly seen between countries that do not follow a common convention [6] while intra variability is perceptible among places which agreed to follow one. In Europe, the Convention on Road Signs and Signals [7] established the common sizes, shapes and colour to be used but allows each country to choose its own symbols and inscriptions. Fig. 2 illustrates some examples of intra class variability where it can be seen that symbols do not only vary between countries but also inside each of them. Regarding the last issue, Croatia and France (Fig. 2 second row) use 2 symbols for pedestrian crossing sign in Danger category while Belgium has speed limit signs with and without adding the Km inscription. Germany also uses 2 different symbols in the pass-right class which belongs to Mandatory category (Fig. 2 fourth row). At the same time, the background colour in some categories can vary as de ned in [8]. For example, Croatia uses the two possible colour (yellow and white) for danger and prohibitory signs (Fig. 2, first and third rows) while the other countries stick to only one. As mentioned earlier and due to the importance of traffic sign recognition, the research in this field has been popular and several methods that use selected hand-coded features as well as the ones which extract the features automatically have been proposed [9]. Among them, the most effective ones relying on CNNs architectures [10]. However, being able to recognize the same traffic sign in different countries is still a problem that in our knowledge, not many studies have intra variability is perceptible among places which agreed to follow one. In german, the Convention on Road Signs and Signals [11] established the common sizes, 2 shapes and colours to be used but allows each country to choose its own symbols and inscriptions. Fig. 1.1 illustrates some examples of intra class variability where it can be seen that symbols do not only vary between countries but also inside each of them. Regarding the last issue, Croatia and France (Fig. 2 second row) use 2 symbols for pedestrian crossing sign in Danger category while Belgium has speed limit signs with and without adding the Km inscription. Germany also uses 2 different symbols in the pass-right class which belongs to Mandatory category (Fig. 2 fourth row). At the same time, the background colour in some categories can vary as de ned in [12]. For example, Croatia uses the two possible colours (yellow and white) for danger and prohibitory signs (Fig. 1.1, first and third rows) while the other countries stick to only

one. As mentioned earlier and due to the importance of traffic sign recognition, the research in this field has been popular and several methods that use selected hand-coded features as well as the ones which extract the features automatically have been proposed [13]. Among them, the most effective ones relying on CNNs architectures [14]. However, being able to recognize the same traffic sign in different countries is still a problem that in our knowledge, not many studies have addressed, especially in a continent (Europe) where countries are a few hours apart. In this paper we summarize our contributions to the following.

Fig. 1 Intraclass variability examples of German traffic signs.

II. CONVOLUTIONAL NEURAL NETWORKS (CNNs) FOR TRAFFIC SIGN CLASSIFICATION

It has been proven that CNNs are capable to solve problems with really high accuracy compared to human performance [28], [34] Since the German Traffic Sign Recognition Benchmark [6], a lot of works were proposed to deal with traffic sign classification through different machine learning methods [10], [15], [28], from which CNNs outperform the others. As traffic sign classification is in high demand for the automotive industry, a lot of efforts have been made to achieve real time classification [10], [28], We will describe 5 CNNs that achieve the best performances in the state of the art regarding Traffic Sign Classification.

A. LENET-5

LeCun et al. [27] proposed the well-known LeNet-5 convolutional neural network that is mostly used for handwritten recognition. Besides it was introduced in 1998, it became popular to solve other problems due to its simple and efficient architecture. It is composed of 7 layers, 3 Convolutional layers followed by Sub-sampling layers (except in the last), 1 Fully connected layer and the nal output layer composed of Euclidean RBF units. The input size for this network is 32 32 pixels.

Jung et al. [10] used LeNet-5 to classify 6 types of Korean traffic signs obtaining an accuracy of 100% recognizing correctly 16 signs while driving on the KAIST campus road. As the results were promising in their study, we also trained the network with our proposed dataset for comparison.

Cire‡an et al. [28] based their work on combining several Deep convolutional Neural Networks (DNN) columns to form a Multi-column DNN (MCDNN). Their DNN network is composed of 2 Convolutional layers followed by Max-pooling layers. At the end, 2 fully connected hidden layers are used to pass the output to a fully connected layer with 6 neurons to perform classification. They used a scaled hyperbolic tangent activation function for convolutional and fully connected layers. Their net takes as input 2 images of 48 48 pixels and performs some distortions in each column to average at the end the nil predictions of each DNN.

Aghdam et al. [15] trained this model with the GTSRB dataset and obtained an accuracy of 98.52% performing data-augmentation for the training set

B. URV MODEL

Aghdam et al. [15] made a comparative study between methods using the GTSRB[6]. Their CNN based on Cire‡an et al.'s [28] work demonstrated in their results, that their network is able to reduce complexity and computational time, improving accuracy compared to the one of Ciseran et al. Their Network is based on 3 convolution-pooling layers and 2 fully connected layers with a dropout layer in between to avoid overfitting. ReLU activations [35] after each convolutional layer and after the rest fully-connected layer is applied. Their network takes as input a 48 48 RGB image and classifies it into one of the 43 traffic sign 24 classes of the GTSRB dataset. They claim to achieve 98.94% accuracy performing dataaugmentation for the dataset.

C. CNN WITH ASYMMETRIC KERNELS

Li and Wang [31] based their CNN design on convolutional layers using asymmetric kernel sizes to replace the usual symmetric n n kernel (e.g. 3 3,5 5,7 7), with asymmetric ones de ned by n 1 and 1 n in some convolutional layers. This replacement decreases the number of convolutional operations making the network more efficient. Their CNN architecture is composed of 3 convolutions with symmetric kernels, 6 convolutional layers with asymmetric ones (7 1, 1 7, 1 3, 3 1, 1 7 and 7 1), and 2 fully connected layers. Each of these layers except for the last one (SoftMax classifier) are followed by Batch Normalization [36] and ReLu activations [35]. They used an inception module with asymmetric kernels after the third convolution to learn different spatial information. The last two convolutional layers use symmetric kernels. Dropout technique [37] is used by the authors to avoid overfitting. As they trained the network with the GTSRB, the output is set to 43 and input size to 48 48 3 (RGB image). The performance of this CNN achieved 99.66% accuracy on the GTSRB test set trained with data-augmentation for 200 epochs. Despite the use of asymmetric kernels to decrease the complexity of the network, the number of parameters for this architecture is still high compared to the others studied here. A comparison of this metric will be provided in the next Section.

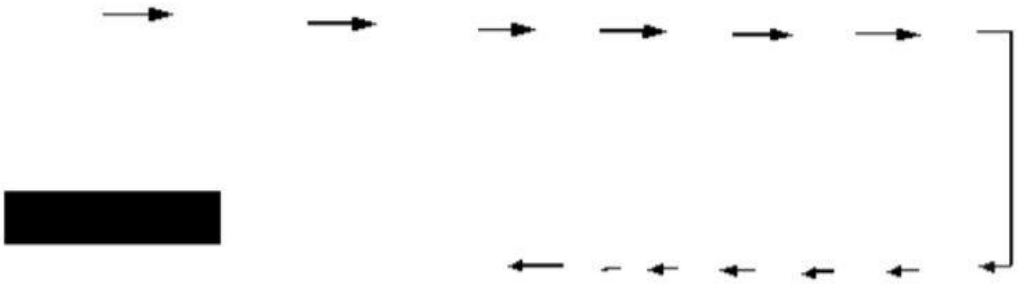

Fig. 2 Architecture drawn from Chilamkurthy proposal.

D. CNN 8-LAYERS

Besides the previously mentioned architectures, we decided to train a classifier that does not represent a really deep network but competes with the high accuracy in the state of the art. Networks composed with deep architectures (several hidden layers) have proven to provide the best results (Inception [38], VGG16 [39], ResNet [40]), but their complexity kills the computational time. For this reason, simple networks are used considering 25 information pre-processing and data-augmentation [41] if the dataset does not contain enough examples for the learning phase. Chilamkurthy [42] worked on the traffic sign image classification problem. Even though he did not mention in which Network he based his work, we could see that his proposal is like a VGG architecture [39]. There are blocks of Convolutional layers, activated by ReLu function [35], followed by Max Pooling and additionally Dropout layers. His architecture can be seen in Fig. 9. Different from VGG architecture, he added dropout layers with a range of 0.2 after each block of convolutional layers and, in the same way, after the fully connected layer with a range of 0.5. Dropout layers are used to prevent overt ting and to make the network learn robustly its parameters [37]. Chilamkurthy reported 97.92% and 98.29% accuracies without and with dataaugmentation respectively using the GTSRB dataset. His network takes RGB images of size 48 48 pixels as input, while transforming them to HSV colour space and performing histogram equalization in the V channel. A comparison of all the previously mentioned architectures will be performed in the next Section using the GTSRB and our German dataset.

III. EXPERIMENTAL EVALUATION

In order to perform dataset comparison between different networks, we trained the models described in Section IV with the GTSDB and our proposed german dataset. All models are trained in GPU mode using a NVIDIA GeForce GTX1080Ti with 11GB of memory, an Intel Core i7K-8700K (6 cores 12 threads, 12 Mb cache memory) processor and RAM of 32GB. The learning process varied according to the complexity of each model. The URV model proposed by Aghdam et al. [15] and the IDSIA model proposed by Cire‡an et al. [28] are implemented in the Caffe framework.1 The input image of both models is an RGB image of 48 48 pixels. We trained both models with the original parameters as stated by Aghdam et al. [15] changing only the batch size from 100 to 128 and the number of iterations to make it learn for the equivalent of 40 epochs (11500 iterations for the GTSRB and 17500 for the European dataset). In the same way, the test iterations are modified according to the validation dataset: 31 for the GTSDB and 48 for the European one.

Model	Input size	GTSRB		European		Time
		Parameters	Accuracy	Parameters	Accuracy	
LeNet-5	32x32x1	0.13 M	89.1%	0.35 M	89.8%	0.0067 ms
IDSIA	48x48x3	1.54 M	94.62%	1.58 M	95.82%	0.6 ms
URV	48x48x3	1.12 M	96.1%	1.16 M	95.53%	0.61 ms
CNN asymmetricK	48x48x3	2.92 M	97.88%	2.95 M	98.48%	0.39 ms
CNN 8-layers	48x48x3	1.48 M	98.52%	1.51 M	97.38%	0.15 ms

Table. 1 Accuracy percentage results obtained on the GTSRB and European test sets. The input size refers to image ''width height channels'', while the number of parameters is presented in millions (M) and time in milliseconds (Ms)

Towards a fair comparison, we normalize the European dataset subtracting the mean image like it is done for the GTSDB dataset. The results presented in [15], based their accuracy on augmented-data carried out with 12 transformations (see paper [15] for more details) UVR and IDSIA models were trained without performing any data-augmentation or pre-processing.

This with the aim to evaluate the performance of the pure models. We use 10% of the training sets for validation (3921 images for GTSDB and 6055 for European dataset). The results obtained can be seen in Table. 2, where the classification accuracies presented come from evaluating the models on the test sets. In the same manner, we trained the models: LeNet-5 [27], the CNN with asymmetric kernels proposed by Li and Wang [31] (CNN asymmetric), and the model proposed by Chilamkurthy [42] (CNN 8-layers) implementing some changes to increase the model accuracy. All these architectures are implemented in the Tensor ow framework.

The processing time for each model depends on its number of parameters and the framework used. For instance, the processing times presented in Table. 2 are computed to predict the traffic sign class of a single image in GPU mode. Essentially, we can see that the models implemented in the Caffe framework (IDISA and URV) are relatively slower than the ones implemented in Tensor ow. For example, the IDSIA model (Caffe) has 1.54 Million parameters and takes 0.6 milliseconds to make a prediction, while CNN asymmetric model (Tensor flow) has 2.92 Million parameters and takes 0.39 milliseconds (around 40% faster).

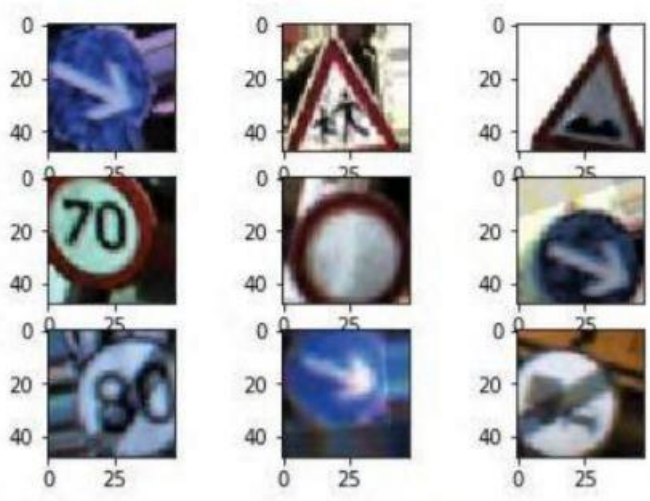

Fig. 3 An example of some augmented traffic signs from the german dataset

As mentioned before, techniques like data-augmentation also help to improve the accuracy of a classier without acquiring and labelling more data. We applied this technique with the 2 models which obtained the best accuracies on the test datasets: the CNN asymmetric model [31] and the modified CNN-8 layers model. Luckily, as these models are implemented in Keras using Tensor ow as back-end. Keras provides an option to perform real-time data-augmentation with its Image Data Generator class. We considered the following 5 transformations: 1) Width shift D C4 pixels 2) Height shit D C4 pixels 3) Scaling D [0.8,1.2] 4) Shear D [0,0.1] radians 5) Rotation D C10 degrees Besides that, transformations, histogram equalization is also considered as data preprocessing. For this, the exact same procedure is applied as stated by Chilamkurthy [42]. Some examples of augmented images can be seen in Fig. 10. 34 Considering that the models were previously trained with-out data-augmentation, we used their pre-trained weights as initializers for the new training procedures. This technique is also called transfer learning and avoids learning everything from scratch. Normally, it is more common to use it with deep architectures which were trained on huge amount of data to adapt the model to a new output with less training examples [44]. In our case, we used it as initialization for the architectures to continue learning with the new generated data. The training parameters were left unchanged as de ned previously and only the number of epochs was set to 50 for both CNN models. Table. 3 shows the accuracies obtained on the test sets on each dataset. These testing accuracies with the CNN asymmetric model [31] are improved by 1.49% in the GTSDB dataset and 0.41% in the European dataset, while with the modified CNN 8-layers model, they are improved by 0.85% and 1.11% respectively.

Model	GTSRB		European	
	Original	Augmentation	Original	Augmentation
CNN asymmetricK	97.88%	99.37%	98.48%	98.89%
CNN 8-layers	98.52%	99.37%	97.88%	98.99%

Table. 2 Accuracy percentage results obtained on the GTSRB and European test sets without and with performing data-augmentation.

The average human performance for detecting traffic signs on the GTSRB dataset is 98.84% as reported in [6]. Both CNNs trained in this study with data-augmentation surpassed the human performance with both datasets. For the URV model proposed by Aghdam et al., [15] we obtained 96.1% accuracy while they reported 98.94% applying 12 transformations as data-augmentation. With this in mind, we can affirm that a classifier learns more robustly if the dataset comprises a wide variety of data situations.

Due to the fact that the proposed European dataset com-prises a wide range of situations and possesses a larger number of training data; most of the accuracy results for each model are improved comparing them to the accuracies obtained on the GTSRB dataset (see Table. 2). Nevertheless, even with data-augmentation, the models could not achieve more than 99% accuracy in the European dataset . We will analyse the predictions in both datasets with the 2 CNN models trained with data-augmentation to find out the reasons that made the classifiers failed.

A deeper analysis for the incorrect predictions on the European dataset is performed to find out the characteristics of the traffic signs that make the classifiers failed. The rst intuition for bad predictions was image size and aspect ratio. We counted the number of misclassified signs that were 1) rectangular or squared and 2) big or small. A squared sign is considered if its aspect ratio falls between 0.9 and 1.1, while small signs are considered if the image has less than 255 pixels.

The latest parameter is set taking into account that the smallest image size on the GTSRB is 15 15 pixels (255 pixels). As a reference, the total number of incorrect predictions with the CNN asymmetric model is 244, while for the CNN 8-layers model is 222. Fig. 11 shows the relation of the incorrect predictions according to the parameters previously mentioned. There, we can see that only a few signs are small in each of the classifiers (8.2% for CNN asymmetric model and 13.06% for the CNN 8-layers model), while almost half of the incorrect predicted signs are rectangular with both classifiers. We considered these metrics because: 1) when the image size is small, even for humans, it is hard to distinguish the correct class; and 2) when the image is rectangular, the classifier resizes it to a squared size suffering information loss.

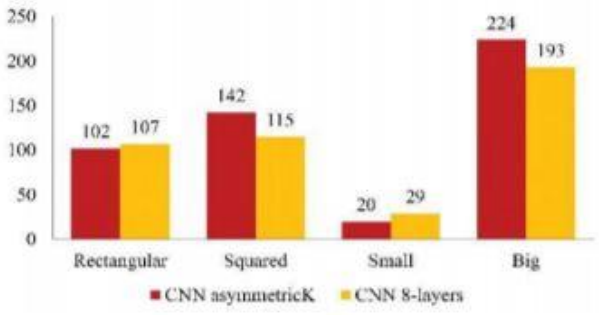

Fig. 4 Image size analysis for the incorrect predictions on the german test set. Results are obtained with the CNN models trained with data-augmentation.

In the same way, we analysed the predicted probabilities for the misclassified signs. We consider as uncertain predictions the ones which are incorrect and predicted with a probability equal or bigger than 0.9. The most uncertain predictions obtained with the CNN asymmetric model were 107/244 D 43.85% while for CNN 8-layers model were 65/222 D 29.28%. This kind of analysis can help a classifier refuse the prediction if the confidence probability is less than a certain threshold, however in this approach is not applicable. As we are interested in the visual characteristics that make the signs difficult to classify correctly, we inspected all the incorrect predictions. Fig. 12 and Fig. 13 illustrate some of the incorrect predictions for the CNN asymmetric model [31] and CNN 8-layers model [42] respectively. After the visual inspection, we found that most of the misclassified signs possess the following characteristics: 36 Strong motion blur. 1. Incomplete signs (cropped). Occlusions. 2. Strong shadows or highlights. Strong perspectives. 3. Human added artifacts. Poor image quality. 4. Aging. 5. Very different aspect ratios (rectangular signs). Most of the errors in Danger and Regulatory categories are due to the characteristics listed above. For the Informative category, the misinterpreted signs are mostly due to their visual complexity and to the very different aspect ratios. Informative signs contain text, which by nature, makes them the hardest ones to recognize. At the same time, their very different aspect ratios conduce them to information loss once the classifier resizes them to a common input shape, normally, a squared shape.

Category	GTSRB			European		
	Signs in GT	CNN asymmetricK	CNN 8-layers	Signs in GT	CNN asymmetricK	CNN 8-layers
Danger	2790	1.25%	0.39%	4626	1.75%	1.36%
Priority	1680	0.00%	0.83%	2946	0.17%	0.20%
Prohibitory	6390	0.66%	0.83%	8625	1.18%	0.94%
Mandatory	1770	0.17%	0.11%	2818	0.43%	0.64%
Special Regulation	-	-	-	1550	1.35%	1.35%
Information	-	-	-	59	0.00%	3.39%
Direction	-	-	-	706	2.55%	3.12%
Additional panels	-	-	-	392	0.77%	1.79%
Others	-	-	-	208	0.96%	0.96%
Total	12630	0.63%	0.63%	21930	1.11%	1.01%

Table. 3 Error percentage predictions by category on the GTSRB and European test sets. Results are computed from the CNN models trained with data-augmentation.

For example, in the Direction sub-category, most of the errors reside on confusion of class 139 (Direction to place) with class 138 (Advance directional signs) and vice-versa (see Fig. 13) due to the fact that both contain text and their appearances vary a lot.Interestingly, no matter how many conditions our proposed European dataset considers (Fig. 8), there will always be hard situations for the classifiers to learn. In order to overcome this issue, image processing techniques can be used to enhance the visibility of an image and data- augmentation can be applied to improve the learning process generating more samples.

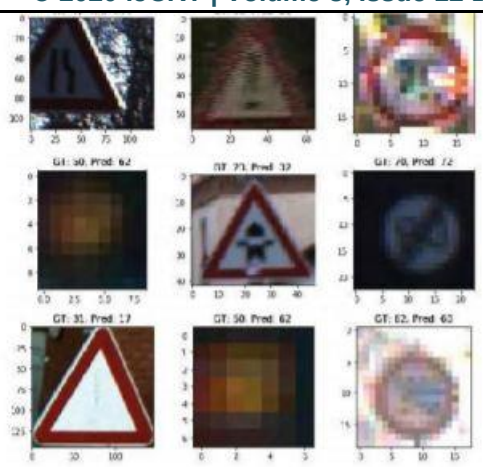

Fig. 5 Random sample of incorrect predictions of the german dataset with the CNN asymmetric model trained with data-augmentation.

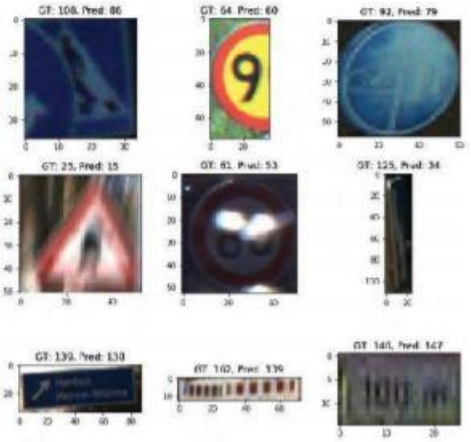

Fig. 6 Random sample of incorrect predictions of the German dataset with the CNN 8- layers model trained with data-augmentation

In summary, our proposed European traffic sign dataset proved to be more robust than the GTSRB dataset with the 5 CNN architectures trained on (Table. 2) making it reliable and more complete for traffic sign recognition. After applying transfer learning to the pre-trained VGG16 network and trying out different hyperparameter sets the results showed that the set with best performance was the hyperparameter values: learning rate=1e-6, batch size=15 and dropout rate=0.3. This hyperparameter set got the best performance with 74.50% accuracy and a loss of 1.0746. Another hyperparameter set that also performed well was the one with a batch size of 30 and the learning rate and dropout rate at standard values (1e-6 and 0.5). This 38 hyperparameter set got 73.5% accuracy and a loss of 1.0064. Table 2 shows the classification rates and losses for different values of the dropout rate. In table 3 the classification rates and losses for different batch size values are shown. Table 4 shows the classification rates and losses for different learning rate values.

Learning rate	Batch size	Dropout rate	Classification rate	Loss
1e-6	15	0.2	71%	1.1096
1e-6	15	0.3	74.5%	1.0746
1e-6	15	0.4	70.5%	1.1359
1e-6	15	0.5	70%	1.2288
1e-6	15	0.6	68.5%	1.1174
1e-6	15	0.7	68%	1.3894
1e-6	15	0.8	72%	1.2119
1e-6	15	0.9	70.5%	1.0356

Table. 4 Classification rates and losses for the VGG16 network applied on the GTSRB dataset for different dropout rates

Learning rate	Batch size	Dropout rate	Classification rate	Loss
1e-6	5	0.5	45%	1.6920
1e-6	10	0.5	65.5%	1.1437
1e-6	15	0.5	70%	1.2288
1e-6	20	0.5	70%	1.0331
1e-6	30	0.5	73.5%	1.0064
1e-6	60	0.5	72.5%	0.9804
1e-6	80	0.5	72.5%	0.9755
1e-6	100	0.5	70.5%	0.9690

Table. 5 Classification rates and losses for the VGG16 network applied on the GTSRB dataset for different batch sizes.

Learning rate	Batch size	Dropout rate	Classification rate	Loss
1e-1	15	0.5	71%	1.0759
1e-2	15	0.5	72%	1.0026
1e-3	15	0.5	72%	1.2662
1e-4	15	0.5	70.5%	1.1618
1e-5	15	0.5	68.5%	1.0823
1e-6	15	0.5	70%	1.2288
1e-7	15	0.5	73.5%	1.1362
1e-8	15	0.5	71.5%	1.0548

Table. 6 Classification rates and losses for the VGG16 network applied on the GTSRB dataset for different learning rates.

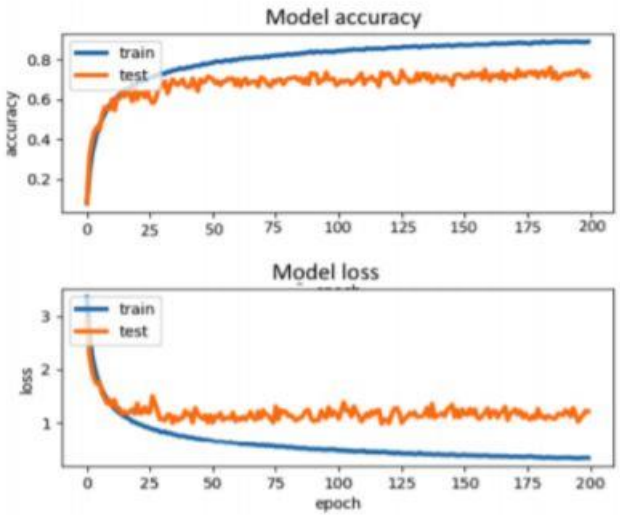

Fig. 8 Results from the training and testing of the VGG16 network on the GTSRB after applying transfer learning using bottleneck features. The graphs show the accuracy and loss curves for the training and test data for dropout rate=0.3, learning rate=1e-6 och batch size=15.

Two graphs are shown, one for the model accuracy and one for the model loss. The blue lines are for the training data and the orange lines are for the test data. The graphs are for the case when the learning rate and batch size had standard values of 1e-6 and 15 respectively and the dropout rate was 0.3, slightly lower than the standard value. As one can see in the test loss curve flattens out at around 1.0. The test accuracy reaches 74.5% shows the confusion matrix for the same hyperparameter values (learning rate=1e-6, batch size=15 and dropout rate=0.3). The more aligned the colored squares are along the diagonal the better is the model at classifying the traffic signs correctly. The x-axis represents the predicted labels, that is, the traffic sign that the network think is the correct one and the y-axis represents the truely correct traffic sign.

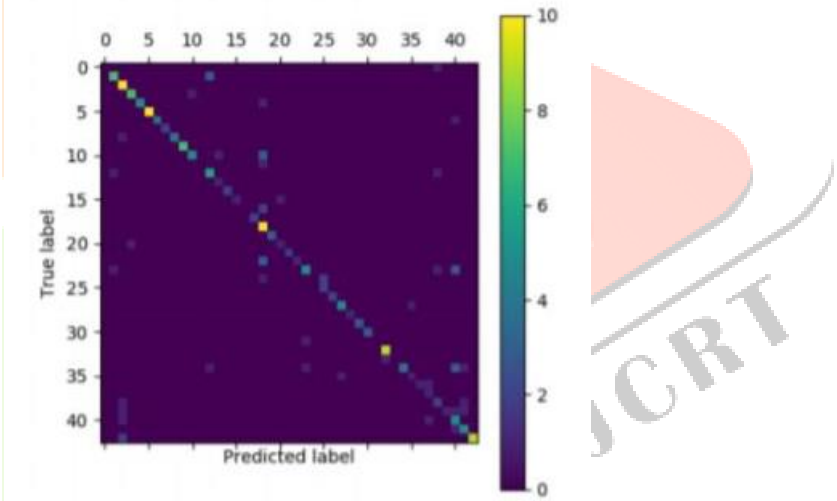

Fig. 9 Confusion matrix for dropout rate=0.3, learning rate=1e-6 och batch size=15. The confusion matrix shows what labels the model have predicted. The more aligned the colored squares are along the diagonal the better is the model at classifying the traffic signs correctly.

IV. CONCLUSIONS

We proposed a traffic sign European dataset which deals with intra-class variability from 6 countries (Belgium, Croatia, France, Germany, Netherlands and Sweden). Such characteristic is a crucial aspect for autonomous vehicles when driving from one country to another, since a classifier does not perform properly when traffic signs (pictographs or text) are slightly different from each other [15]. In Europe, this is a vital issue considering that countries are relatively close to each other. For this reason, denning a traffic sign dataset that contemplates the aforementioned problem, conducts our work to make an important contribution for intelligent vehicles.

Simultaneously, by training several state of the art CNN models, we showed that Deep CNNs are not required to solve traffic sign classification. Instead, techniques like image preprocessing and data-augmentation are used to improve classification accuracy. Accuracies of 99.37% and 98.99% were the best results obtained for training the GTSRB and our European dataset respectively with the modified CNN 8-layers model.

However, the classes based on text and with very different aspect ratios (most belonging to Informative category) were the most challenging ones to learn. This problem comes from the input definition of the CNNs since they require axed size (most of the time squared). In consequence, information might be discarded when downscaling the image, or distorted from the original input.

REFERENCES

[1]C. Grigorescu and N. Petkov, ‘‘Distance sets for shape lters and shape recognition,’’ IEEE Trans. Image Process., vol. 12, no. 10, pp. 1274 1286, Oct. 2003.

[2]S. egvic et al., ‘‘A computer vision assisted geoinformation inventory for traffic infrastructure,’’ in Proc. 13th Int. IEEE Conf. Intell. Transp. Syst. (ITSC), Sep. 2010, pp. 66 73.

[3]F. Larsson and M. Felsberg, ‘‘Using Fourier descriptors and spatial models for traffic sign recognition,’’ in Scandinavian Conference on Image Analysis. Berlin, Germany: Springer, 2011, pp. 238 249.

[4]R. Timofte and L. Van Gool, ‘‘Sparse representation-based projections,’’ in Proc. 22nd Brit. Mach. Vis. Conf.-BMVC, 2011, pp. 1 61.

[5]N. Paparoditis et al., ‘‘Stereo polis II: A multi-purpose and multi-sensor 3D mobile mapping system for street visualisation and 3D metrology,’’ Revue Française Photogramm. TØlØdØtection, vol. 200, no. 1, pp. 69 79, 2012.

[6]J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, ‘‘Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition,’’ Neural Netw., vol. 32, pp. 323 332, Aug. 2012.

[7]R. Timofte, K. Zimmermann, and L. Van Gool, ‘‘Multi-view traffic sign detection, recognition, and 3D localisation,’’ Mach. Vis. Appl., vol. 25, no. 3, pp. 633 647, Apr. 2014.

[8]M. M. Lau, K. H. Lim, and A. A. Gopalai, ‘‘Malaysia traffic sign recognition with convolutional neural network,’’ in Proc. IEEE Int. Conf. Digit. Signal Process. (DSP), Jul. 2015, pp. 1006 1010.

[9]A. 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, pp. 1484 1497, Dec. 2012.

[10] S. Jung, U. Lee, J. Jung, and D. H. Shim, ‘‘Real-time traffic sign recognition system with deep convolutional neural network,’’ in Proc. 13th Int. Conf. Ubiquitous Robots Ambient Intell. (URAI), 2016, pp. 1 4.

[11] S. B. Wali, M. A. Hannan, A. Hussain, and S. A. Samad, ‘‘Comparative survey on traffic sign detection and recognition: A review,’’ in Przegl¡d Elektrotechniczny. 2015.

[12] Economic Commission for Europe-Inland Transport Committee, ‘‘Convention on road signs and signals,’’ United Nations Treaty Ser., vol. 1091, p. 3, Nov. 1968.

[13] Y. Saadna and A. Behloul, ‘‘An overview of traf c sign detection and classification methods,’’ Int. J. Multimedia Inf. Retr., vol. 6, no. 3, pp. 193 210, 2017.

[14] H. Fleyeh and M. Dougherty, ‘‘Road and traf c sign detection and recognition,’’ in Proc. 16th Mini-EURO Conf. 10th Meeting EWGT, 2005, pp. 644 653.

[15] H. H. Aghdam, E. J. Heravi, and D. Puig, ‘‘A practical and highly optimized convolutional neural network for classifying traffic signs in real-time,’’ Int. J. Comput. Vis., vol. 122, no. 2, pp. 246 269, 2017.

[16] Y. Yang, H. Luo, H. Xu, and F. Wu, ‘‘Towards real-time traffic sign detection and classification,’’ IEEE Trans. Intell. Transp. Syst., vol. 17, no. 7, pp. 2022 2031, Jul. 2016.