

Recognition of Traffic Signs with Artificial Neural Networks: A Novel Dataset and Algorithm

Abdulrahman Kerim
Dept. of Computer Engineering
Hacettepe University
Ankara, Turkey
ahaje@hacettepe.edu.tr

Mehmet Önder Efe
Dept. of Computer Engineering
Hacettepe University
Ankara, Turkey
onderefe@hacettepe.edu.tr

Abstract—Traffic sign classification is a prime issue for autonomous platform industries such as autonomous cars. Towards the goal of recognition, most recent classification methods deploy Artificial Neural Networks (ANNs), Support Vector Machines (SVMs) and Convolutional Neural Networks (CNNs). In this work, we provide a novel dataset and a hybrid ANN that achieves accurate results that are very close to the state-of-the-art ones. When training and testing on German Traffic Sign Recognition Benchmarks (GTSRB) a top-5 classification accuracy of 80% was achieved for 43 classes. On the other hand, a top-2 classification accuracy of 95% was reached on our novel dataset for 10 classes. This accomplishment can be linked to the fact that the proposed hybrid ANN combines 9 different models trained on color intensity, HOG (Histograms of Oriented Gradients) and LBP (Local Binary Pattern) features.

Index Terms—artificial neural networks, image classification, pattern recognition, data preprocessing

I. INTRODUCTION

The recent decade witnessed a great advancement in technology which created a wave of improvements, questions and paradigms. That wave propagated influencing almost all related scientific fields. The availability of high speed, powerful and cheap computing resources, motivated the researchers to investigate further the capability of Artificial Intelligence (AI) to solve the most sophisticated and highly nonlinear problems. These include, but are not limited to, visual object tracking, semantic segmentation, optical flow recognition and depth estimation. Machine Learning (ML) and Deep Learning (DL) received a great attention recently because of the high accuracy they were able to achieve as compared to the other methods.

Nevertheless, ML and DL contain a massive number of algorithms and techniques, ANN approach is the heart of it. ANN gets its power from its ability to link some input space with some output space provided with some data implicitly representing the relation between the two domains. Once ANN learns the mapping between the two domains, it can easily recognize and correctly map a new unseen sample from the input to the output space. The main mechanism used to teach this mapping to the ANN (i.e. to train the ANN) is the Error Backpropagation (EBP) [1]. This algorithm simply achieves the training by adjusting the connections of the network (i.e. the weights and the biases) by backpropagating the error measured at the output to all the previous layers using the

chain rule and the gradients. This adjustment of weights should be in the direction that minimizes the training loss of the ANN. Backpropagation algorithm is usually enhanced by including momentum term [2] which helps in eliminating the oscillations in the adjustable parameter space.

Despite many advantages, ANN approach cannot be claimed to be the best for all problem types. Basically, the main drawback of ANN comes from its nature of being a supervised learning method. In other words, to train the network correctly, one should feed it with the suitably labeled data. In many scenarios the availability of the data and the preprocessing of the data have a higher impact on the success of the trained ANN. This shifts the problem from designing a good learning algorithm to providing a suitable data that presents a good proxy of the task or the problem in the real world. Since labeling of training data is expensive, data augmentation comes as a resort for increasing the number of training samples while improving the diversity. This is done by performing some simple data processing techniques. For example, when the data is a set of images, usually, scaling, shearing, translating and rotating the images are examples of data augmentation.

Basically, two fundamentally different sorts of problems where ANN approaches are involved in are classification and regression problems. In classification problems, ANN structure learns to map some continuous or discrete input space to a discrete output space. While, in the regression problems, the ANN learns to map some continuous or discrete input space to a continuous output space. For example, predicting the label of a traffic sign is a classification task, while predicting a house price is a regression task. This work falls into the classification segment.

Our main contribution in this paper is a novel hybrid ANN that is capable of providing accurate results even when used with a very few number of layers. Our second contribution is a new benchmark for training and evaluating traffic sign recognition models.

II. RELATED WORK

In the literature, we can see a number of works that discussed image classification using machine learning methods. However, what distinguishes each of them essentially is the

feature extraction approach and the process of combining these features to classify such images.

Applying HOG and color intensity features for classification and detection of traffic signs was shown in [3], which was able to achieve high accuracy for both classification and detection. Parallel to [3], we use HOG and color intensity features but we train each model for one feature then we combine them as a final hybrid model. i.e. for Red channel, we train one separate ANN and so on for all the other features.

In [4], the traffic sign detection and classification task is studied by combining an attention mechanism to the classifier and localizer models. The attention model predicts the most probably location of the traffic sign while the localizer locates it and the classifier predicts its class. They deploy the faster R-CNN [5] to do this much more faster than the standard algorithms. Unlike [4], we do not apply any attention mechanism since we handle exclusively the classification problem.

While in [6], the authors solved the classification and detection of the traffic signs by using Rectified Linear Unit (ReLU), max pooling and fully connected layers. The network has four layers and it was able to achieve acceptable results. The main advantage of the work in [6] is the simplicity of the network. However, the low accuracy was one main disadvantage.

On the other hand, another work [7] suggested using adaptive neuro-fuzzy inference system for traffic sign recognition task. They use the geometric information of the traffic sign as an input to the proposed system. Their system is composed of five different independent models and the final decision is made using a majority voting mechanism. Similar to this work, we use different independent models to guess the label of the traffic sign. In addition to that, a majority voting technique is used to give the final label. However, we do not use the geometric properties of the traffic sign in the current version of our system.

In [8], the traffic sign detection is performed using HOG and SVM. HOG technique was utilized for extracting the features from the input image. While, SVM was applied for deciding if the image contains a traffic sign or not. If a traffic sign was detected, a neural network was used to do the classification task. Similar to their work, we use ANN for recognition task. Unlike their work which uses a single neural network for the classification task, we use a hybrid ANN consisting of three sub-models. Each of them was trained on a different feature type to achieve a higher degree of robustness in terms lighting conditions and noise.

Alternatively, in [9] the authors suggested applying a simple preprocessing algorithm on the input images, by making the edges more pronounced and by improving the color contrast, to enhance the visual quality of a possible traffic sign contained in that image. They show that applying this technique can boost both the accuracy of detection and recognition of the traffic sign.

Since our second contribution is a novel dataset, we discuss some of the available traffic sign classification and localization datasets as well. The first widely known benchmark is the

German Traffic Sign Recognition Benchmark (GTSRB) [10]. GTSRB contains more than 15 thousand traffic sign samples distributed over 43 classes. The main drawback of this dataset is the fact that the number of observations per class is not uniform.

The second dataset is the Chinese Traffic Sign Database (Traffic Sign Recognition Database, TSRD) [11]. This dataset contains more than 6 thousand images spanning 58 classes.

There are other datasets such as Russian Traffic Sign Dataset [12], Swedish Traffic Sign Dataset [13], Netherlands Traffic Sign Dataset [14], and French Traffic Sign Dataset [15]. Our final dataset contains more than 73 classes taken from the two datasets, namely GTSRB and TSRD where the similar classes were merged from both datasets.

The availability of a wide range of datasets and algorithms under the scope of traffic sign recognition and detection, does not mean that the problem is tackled correctly. In [16], the authors discussed some challenges in the domain of traffic sign recognition and detection. The lack of a standard benchmark, the difficulty to compare different algorithms and the nature of the task making any non-real-time solution impractical are the main bottlenecks of the field. By proposing a general dataset and a simple yet powerful recognition system, our work comes as a step in the correct direction toward the optimal solution.

III. DATA PREPROCESSING AND NOVEL DATASET

As mentioned earlier, data preprocessing plays a vital role in developing a useful ANN. When we examine the histograms of the GTSRB and TSRD, we can see the distribution of the samples over the classes is quite non-uniform as shown in Fig. 1 and Fig. 2. Training merely on the combination of the two datasets creates two problems. The first problem is caused by the lack of enough training samples for some classes which prevents the ANN learning the actual representation of these classes with no enough samples. The second problem is originated from the imbalance of these classes i.e. different number of samples per class. This causes the network to develop a wrong tendency of predicting the class of the high number of training samples more often as compared to the one with less number of samples. We solve this problem by augmenting the data of the class with the few samples as will be explained in the sequel.

A. Preprocessing TSRD and GTSRB

First of all, we combine the training and testing partitions of the TSRD dataset into one portion. Then, given the bounding boxes of the traffic signs, we use them to crop these signs. Later, we resize all of the cropped images to 32×32 pixels. On the other hand, we join the training, validation and testing partitions of GTSRB into one partition.

B. Combining TSRD and GTSRB datasets

First of all, we inspect all the 101 (43+58) classes of the new dataset manually. By this manual inspection, we merge similar classes between the two datasets. Later, we discard classes that contain less than 50 or more than 1000 observations. At this

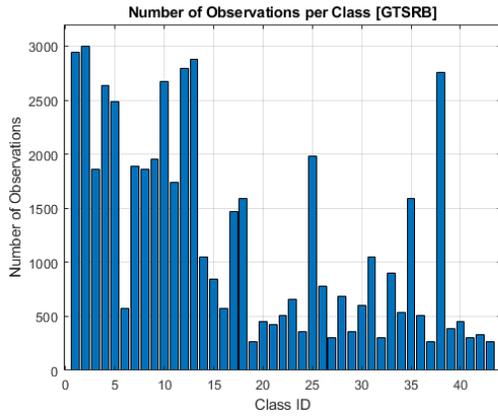


Fig. 1. The number of observations/samples/images for each class of the GTSRB dataset.

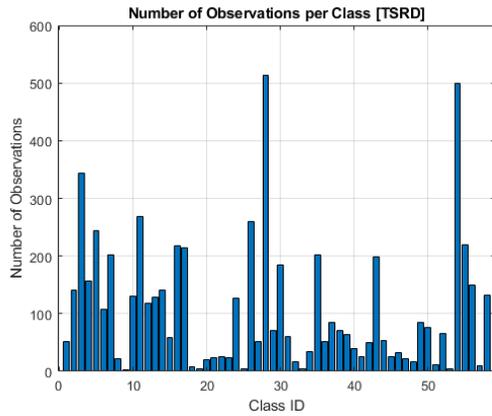


Fig. 2. The number of observations/samples/images for each class of the TSRD dataset.

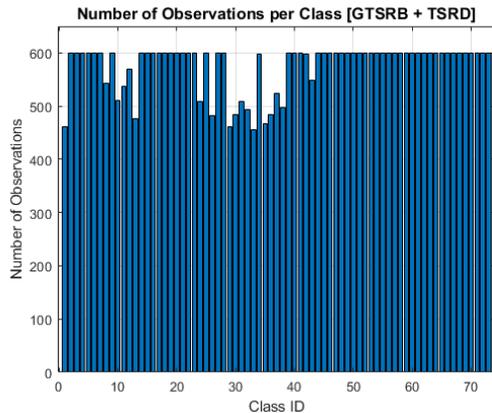


Fig. 3. The number of observations/samples/images for each class of our novel dataset after combining GTSRB and TSRD datasets.

step, we divide our dataset into three parts: training, validation and testing subsets. We follow the rule of thumb by taking 80% for training, 10% for validation and 10% for testing. Afterwards, the data augmentation is applied -only- on the

TABLE I
TOP-5 CLASSIFICATION ERROR IS SHOWN FOR COLOR AND HOG MODELS.
HOG WITH CELL SIZE OF 4 ACHIEVES THE BEST ACCURACY.

Classification Error on GTSRB	
Model Name	Classification Error
Red_Channel	0.242
Green_Channel	0.241
Blue_Channel	0.239
HOG_Cell_Size_2	0.319
HOG_Cell_Size_4	0.204
HOG_Cell_Size_8	0.259

training and validation sets.

C. Data Augmentation

At this step, we have 73 classes but with a non-uniform samples distribution. Thus, we need to increase the number of observations while not sacrificing the diversity of samples. It is common to use translation and rotation to augment the data. However, in addition to that, we perform noise addition, color and intensity jittering, and blurring:

- **Translation:** We translate the original images horizontally and vertically by 5 pixels (around 16% of the original size).
- **Rotation:** We rotate the original images clockwise and anti-clockwise by 10 and 20 degrees.
- **Noising:** We noise the original images using Gaussian white noise with mean of 0 and variance of 0.001. We also apply Poisson noise and speckle noise.
- **Color and Intensity Jittering:** We change pixels' intensity for the color channels randomly to account for different lighting conditions.
- **Blurring:** We apply four different blurring kernels i.e. 2 by 2, 3 by 3, 4 by 4, and 3 by 3 with intensity boosting.

We note that horizontal or vertical flipping should not be applied since this will alter the meaning of the traffic sign. At the same time, rotating the traffic signs with more than 20 degrees may change the semantic of the sign. Thus that should be avoided as well.

D. Best Samples Selection

After applying the previously mentioned data augmentation methods, we need now to pick the most useful samples for training process. To achieve this aim, we need to answer the question of what does make a sample much more useful than the others? Basically, as much as the sample does not repeat the features covered by a previously seen sample, it is considered to be useful in the aim of learning the set of features that represents that particular class. For that reason, we apply the following novel mechanism for solving this issue.

First of all, for each image, we calculate its similarity index with randomly selected 100 images from the same class using the Structural Similarity Index Measure (SSIM) as described by the famous work of Wang et al. in [17]. Then, we sort the images with lowest similarity score i.e. the ones that are most dissimilar to others. Following this, we pick the first 600



Fig. 4. Each row shows a small sample of one class of our novel dataset. Note the diversity and the challenging nature of these samples.

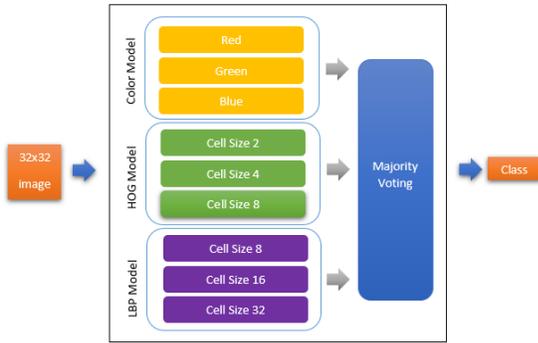


Fig. 5. The block diagram of the proposed hybrid ANN. It consists of Color model, HOG model, and LBP model. The Final decision of the type of traffic sign is taken after performing a majority voting.

samples. The rich diversity of the samples for each class can be seen from Fig. 4. At the same time, by referring to Fig. 3, it is clear to see the near-optimal match in the number of observations per class. To the best of our knowledge, this is the first study that discusses combining two different traffic signs datasets using the similarity measure.

IV. NOVEL CLASSIFICATION ALGORITHM

Feature extraction from images is still one of the hot topics because of its huge applications in computer vision field. Training using a single feature representation is not expected to let the ANN to develop a good understanding of the essence of the task it tries to learn. When the task is traffic sign recognition or classification, it is highly recommended to combine a set of features to tackle this task in an appropriate way. Since this task is closely related to autonomous cars, the classification system's accuracy and response time are major issues. In other words, failing to classify a pedestrian crossing

sign where the car is required to decrease its speed, may lead to injuries, damage to environment and properties, or even death. At the same time, being able to classify that sign correctly, however, with some time delay also may lead to the same terrible results.

In our work, we tackle the traffic sign classification problem under the previous two restrictions i.e. time response and accuracy. The time restriction forces us to use the simplest possible ANN while the accuracy one motivates us to develop a hybrid ANN for that task. We propose a novel hybrid ANN that combines 9 different ANN's. Each ANN is responsible for looking at the traffic sign using one particular set of features that are all embedded in the image of the traffic sign. Fig. 5 shows the main block diagram of the proposed hybrid ANN. The main three components are the RGB model, the HOG model, and the LBP model. Each model is composed of three different sub-models each one of them was trained using particular set of features.

A. Hybrid-ANN Classification Models

The RGB model consists of three sub-models: The first one was trained on the Red channel, the second on the Green channel and the third on the Blue channel. Each of these models captures the features that are presented in one particular color channel.

The HOG model i.e. the model based on histogram of oriented gradients also includes three sub-models. The three sub-models are trained on the HOG features for three different cell sizes i.e. 2×2 , 4×4 , and 8×8 . Basically, HOG allows us to train on the structure of the traffic sign which makes it more robust to lighting conditions and noise. It is important to note that using different cell sizes allows us to vary the details of the structural content of the traffic sign. Using very small cell size will cause the ANN to learn useless structures (e.g. noise). On the other hand, using very large cell sizes causes

TABLE II
TOP-1:5 CLASSIFICATION ERRORS ARE SHOWN BY EACH COLUMN FOR FUSING TOP 1:5 PER-MODEL PREDICTION(S).

Classification Error on GTSRB					
	Top-1	Top-2	Top-3	Top-4	Top-5
E1	0.393*	0.346	0.316	0.270	0.259
E2	0.402	0.310	0.285	0.259	0.242
E3	0.393	0.315	0.260	0.227	0.208
E4	0.440	0.318	0.256	0.213	0.190*
E5	0.436	0.299*	0.241*	0.201*	0.190

the network to miss some important structural features. Thus, the cell size is task-dependent and should be selected wisely.

The LBP model, the model based on the local binary patterns of the traffic sign, consists of three sub-models. As we did with HOG, we use 8×8 , 16×16 , and 32×32 cell sizes. Basically, LBP is used to learn the texture information in the images.

B. Hybrid-ANN Decision Model

Since we have many predictions, we apply a majority voting mechanism to decide the final label of a given traffic sign. We have designed a different set of experiments to show which is the optimal voting mechanism. At the same time, our decision model does not take into account the confidence of the predicted class by the classification model. Thus, it does not perform a weighted voting mechanism where the history of the classification model in the validation phase can be used to prioritize one classification model over another. These two points were not considered within this work’s scope for simplicity. However, they can be done easily because our ANN is trained to not just give the expected label but also the confidence score.

V. EXPERIMENTS

In this work, we performed two sets of experiments to further validate our results. In this section, we discuss these experiments in more detail.

A. Experiments on GTSRB Dataset

We train the ANN for all 43 classes. We do not do any data augmentation or preprocessing. We use only the color and HOG models. To simplify the problem, we allow the network to predict the most probable 5 classes (This is widely used in multi-classification problems). The training, validation and test partitions are taken as suggested by the GTSRB. The results of this setup are shown in Table I. In this setup, the performance of the individual models is inspected. In order to investigate the effect of allowing the models to make more than one prediction on the overall classification accuracy, the experiment shown in Table II was conducted. In this table, each row shows the final classification error when the actual class is within the n predicted classes (T_n , $n:1$ to 5). While, each column shows the final classification error when each learning model is allowed to make k predictions (E_k , $k:1$ to 5).

B. Experiments on the Dataset Created in This Work

In this set of experiments, we train, validate and test on the dataset we created. We pick 10 classes randomly and we perform similar experiments as in V-A. However, this is done with the following additions: First of all, we consider LBP in addition to the color and HOG features as shown in Table IV. Secondly, we investigate the performance of the individual models, color, HOG and LBP separately in addition to the hybrid ANN as depicted in Table V. Noting that for instance Red, Green and Blue sub-models were used only to make the ultimate guess for the ANN_Color model while all other models were disagreed. Finally, we examine the error tolerance -as presented earlier- in Table III.

VI. RESULTS

We discuss the results in this section as follows: In Experiment 1, we study the performance of the models on the GTSRB dataset where the training, validation and testing were performed only on this dataset. In parallel to 1, we perform another set of experiments on our novel dataset under the scope of Experiment 2.

A. Experiment 1

We can conclude from Table I that applying HOG feature extraction algorithm with a cell size of 4×4 obtained the best accuracy i.e. around 80%. This suggests that using HOG is promising for multi-classification tasks. On the other hand, Table II shows that allowing the weak learners to do more than one prediction can boost the performance of the final hybrid model. Note that the smallest error for each column and row is denoted by an asterisk and bold font, respectively.

B. Experiment 2

As it is depicted in Table V, combining color, HOG and LBP features can achieve the optimal performance compared to applying one model on its own. From Table IV, we can see HOG and LBP models were able to achieve better results than the color models. The reason for this could be that color models are very sensitive to lighting conditions and recording noise. Finally, we can see from Table III that models trained on our novel dataset achieve an 87% top-1 classification accuracy while a 95% was obtained for top-2 classification accuracy. Note that the smallest errors for each column and row are denoted by an asterisk and bold font, respectively.

TABLE III
TOP-1:5 CLASSIFICATION ERRORS ARE SHOWN BY EACH COLUMN FOR FUSING TOP 1:5 PER-MODEL PREDICTION(S).

Classification Error on Our Novel Dataset					
	Top-1	Top-2	Top-3	Top-4	Top-5
E1	0.130*	0.061	0.039	0.019	0.014
E2	0.147	0.051*	0.032	0.017	0.010
E3	0.159	0.058	0.025*	0.014	0.005*
E4	0.198	0.061	0.025*	0.007*	0.005*
E5	0.225	0.069	0.027	0.008	0.008

TABLE IV
TOP-1 CLASSIFICATION ERROR IS SHOWN FOR TRAINING AND TESTING ON OUR DATASET. LBP MODEL WITH A CELL SIZE OF 8 ACHIEVES THE BEST ACCURACY.

Classification Error on our Novel Dataset	
Model Name	Classification Error
Red_Channel	0.345
Green_Channel	0.327
Blue_Channel	0.313
HOG_Cell_Size_2	0.308
HOG_Cell_Size_4	0.311
HOG_Cell_Size_8	0.206
LBP_Cell_Size_8	0.205
LBP_Cell_Size_16	0.355
LBP_Cell_Size_32	0.516

TABLE V
TOP-1 CLASSIFICATION ERROR FOR COLOR, HOG, LBP MODELS AND OUR PROPOSED HYBRID ANN. OUR PROPOSED MODEL ACHIEVES THE LOWEST ERROR.

Classification Error on our Novel Dataset	
Model Name	Classification Error
ANN_Color	0.267
ANN_HOG	0.210
ANN_LBP	0.289
ANN_Hybrid	0.130

VII. DISCUSSION

We have shown that applying a relatively simple ANN on the task of traffic sign recognition can achieve acceptable performance. Increasing the complexity of the system is thought to increase the accuracy but will increase time complexity, too. Thus, it will limit the system usability. SSIM was preferred on other measures because it models better the perceived structural information of images and it is more robust. However, its time complexity is a key disadvantage. Using other alternatives to SSIM and comparing both accuracy and time complexity can be done in the future.

VIII. CONCLUSION

We presented a novel dataset that combines TSRD and GT-SRB datasets using data augmentation and SSIM techniques. We have shown that using color, HOG and LBP features can improve the accuracy of the classification methods considerably. We relate this improvement to the capability of the ANN to consider the distinctive features and to neglect irrelevant ones. However, we still need to see if the same conclusion can be found for state-of-the-art recognition algorithms, too.

This work can be extended by investigating the effect of increasing the number of classes and the advantages of assigning weights to classes on the overall classification accuracy. Alternatively, including more feature extraction methods could lead to even a better accuracy. On the other hand, studying the time complexity of such methods can be considered as well.

REFERENCES

- [1] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [2] D. C. Plaut *et al.*, "Experiments on learning by back propagation." 1986.
- [3] V. Balali, A. A. Rad, and M. Golparvar-Fard, "Detection, classification, and mapping of us traffic signs using google street view images for roadway inventory management," *Visualization in Engineering*, vol. 3, no. 1, p. 15, 2015.
- [4] Y. Lu, J. Lu, S. Zhang, and P. Hall, "Traffic signal detection and classification in street views using an attention model," *Computational Visual Media*, vol. 4, no. 3, pp. 253–266, 2018.
- [5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.
- [6] N. Kryvinska, A. Poniszewska-Maranda, and M. Gregus, "An approach towards service system building for road traffic signs detection and recognition," *Procedia Computer Science*, vol. 141, pp. 64–71, 2018.
- [7] M. Stojčić, A. Stjepanović, M. Kostadinović, G. Kuzmić, and M. K. Banjanin, "Adaptive neuro-fuzzy model for traffic signs recognition," in *2020 19th International Symposium INFOTEH-JAHORINA (INFOTEH)*. IEEE, 2020, pp. 1–6.
- [8] H.-Y. Lin, C.-C. Chang, V. L. Tran, and J.-H. Shi, "Improved traffic sign recognition for in-car cameras," *Journal of the Chinese Institute of Engineers*, vol. 43, no. 3, pp. 300–307, 2020.
- [9] J. A. Khan, Y. Chen, Y. Rehman, and H. Shin, "Performance enhancement techniques for traffic sign recognition using a deep neural network," *Multimedia Tools and Applications*, pp. 1–16, 2020.
- [10] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: A multi-class classification competition." in *IJCNN*, vol. 6, 2011, p. 7.
- [11] L. Huang, "Chinese traffic sign database," 2019.
- [12] V. I. Shakhuro and A. Konouchine, "Russian traffic sign images dataset," *Computer Optics*, vol. 40, no. 2, pp. 294–300, 2016.
- [13] F. Larsson, M. Felsberg, and P.-E. Forsen, "Correlating Fourier descriptors of local patches for road sign recognition," *IET Computer Vision*, vol. 5, no. 4, pp. 244–254, 2011.
- [14] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view traffic sign detection, recognition, and 3d localisation," *Machine vision and applications*, vol. 25, no. 3, pp. 633–647, 2014.
- [15] R. Belaroussi, P. Foucher, J.-P. Tarel, B. Soheilian, P. Charbonnier, and N. Paparoditis, "Road sign detection in images: A case study," in *2010 20th International Conference on Pattern Recognition*. IEEE, 2010, pp. 484–488.
- [16] B. Sanyal, R. K. Mohapatra, and R. Dash, "Traffic sign recognition: A survey," in *2020 International Conference on Artificial Intelligence and Signal Processing (AISIP)*. IEEE, 2020, pp. 1–6.
- [17] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.