Journal of Computer Science

Original Research Paper

RF-CNNS: Thin Deep Learning Networks For Accelerating Traffic Signs Recognition

Btissam Bousarhane and Driss Bouzidi

Smart Systems Laboratory (SSL), National School of Computer Science and Systems Analysis ENSIAS, Mohammed V University in Rabat, Morocco

Article history
Received: 18-03-2023
Revised: 10-07-2023
Accepted: 04-08-2023

Corresponding Author:
Btissam Bousarhane
Smart Systems Laboratory (SSL), National School of Computer Science and Systems Analysis ENSIAS,
Mohammed V University in Rabat, Morocco
Email: ibtissam_bousarhane@um5.ac.ma

Abstract: Road scene analysis is a wide domain of research that aims to ameliorate the environmental perception in intelligent transportation systems, including autonomous vehicles and advanced driver-assistance systems. It also plays a crucial role in road safety improvement, by contributing to the reduction of traffic accidents' rate. Ensuring an efficient recognition of traffic signs contributes enormously to making cars safer, which in consequence helps to save more lives on roads. To ensure this recognition, many approaches are adopted, especially deep learning ones and more specifically convolutional neural networks. In effect, these networks have proven their high performances in many fields of computer vision research, including traffic sign recognition. However, although their high performances, many limitations still face their implementation, especially in real-time applications and resource-constrained environments. From this perspective, creating a certain balance between the model complexity and the classification accuracy, through a computationally efficient network, is the main objective of this study. To achieve this goal, a receptive fields architecture is adopted to preserve and optimize the connectivity between the different units of the network. Based on this architecture, two receptive field networks are proposed, with reduced complexity and enhanced generalizability. Using two public datasets, the obtained results show that the adopted approach ensures high classification accuracies and considerably accelerates the inference stage. The obtained accuracy is about 98.49%, using the Belgium traffic signs classification dataset, while the inference time is less than 500 us per image.

Keywords: Traffic Signs Classification, Receptive Fields Feedforward Network, Receptive Fields CNN, Locally Connected Network

Introduction

Traffic sign recognition is not a new field of research. It presents, indeed, a part of a wider domain, which is more precisely road scene analysis. In fact, researchers have been interested in this domain for more than 30 years and the first work realized in this context dates back to the 1980s (Akatsuka and Imai, 1987) in Japan. After that, many approaches have been adopted by researchers to ensure this recognition.

The application fields of traffic sign recognition are multiple. These fields of research include among others: Road infrastructure maintenance, road heritage inventory, road safety improvement, data mapping in geographic information systems, the new generation of web multimedia tools for 3D navigation, etc. In addition to these areas, recently autonomous vehicles, Advanced Driver Assistance Systems (ADAS), and self-driving cars (Zang *et al*., 2018) present the main application fields of traffic sign recognition. To ensure this recognition, multiple approaches are adopted by researchers.

In this context, we find that Deep Learning approaches (DL) are used by many researchers in this field of research. However, many challenges still face the implementation of this type of approach, especially for real-time applications, characterized by high-speed mobility and also in low-resources environments, such as mobile devices, with limited memory, CPU, battery, etc. Furthermore, the training time in addition to the problem of overfitting and vanishing gradient, related especially to deep architectures, decreases considerably their performances. From this perspective, to ensure the efficiency and speed of traffic signs classification, using Convolutional Neural Networks (CNNs), the adopted approach proposed in our work consist in:

- Finding an optimal DL architecture that optimizes the interconnectivity between CNNs' hidden units
- Reducing the network complexity to accelerate the recognition process through hyperparameter tuning
- Enhancing the generalization abilities to face local minima and overfitting problems

Related Works

Self-driving vehicles and ADAS systems represent one of the main application fields of traffic sign recognition. This recognition plays in fact an essential role in decreasing the number of road accidents and improving traffic and vehicular safety in general.

The recognition process includes in effect two principal stages (detection and classification). The first stage consists in detecting the position of potential traffic signs in road scene images, while the second one consists in classifying the detected Regions of Interest (ROIs) within one of the different categories of traffic signs.

To ensure this recognition, multiple approaches are proposed by researchers. These methods are essentially based on Machine (ML) and Deep Learning approaches (DL). ML techniques usually used are k-Nearest Neighbor (k-NN), decision trees, random forest, and Support Vector Machines (SVMs) (Aziz and Youssef, 2018). Another type of ML based approach uses instead Artificial Neural Networks (ANNs).

In this context, we can mention the work of Vashisht and Kumar (2023), based on ANNs for traffic sign detection. To ensure efficient recognition under challenging conditions, Chi-Squared Feature Selection is used with the proposed network. Good results are achieved using the public dataset Mapillary.

In fact, this type of ML method is essentially based on the manual extraction of features. These features include a Histogram of Oriented Gradient (HOG), like the approach adopted by Huang *et al*. (2016), which uses the HOG variant with an Extreme Learning Machine (ELM). A combination of HOG, Gabor, and Compound Local Binary Pattern (CLBP) is used instead by Aziz and Youssef (2018), etc. The main advantage of this type of ML method is its simplicity and efficiency (Lim *et al*., 2023a). However, their performances are tightly related to the quality of the extracted features.

For this specific reason, many researchers adopt instead DL based approaches, to ensure an automatic extraction of high-level features from raw input data. We find hence that many deep neural networks are proposed, in the last decade, to ensure traffic signs recognition (Jurišić *et al*., 2015; Arcos-García *et al*., 2018; Li *et al*., 2019; Mehta *et al*., 2019; Zaibi *et al*., 2021; Fang *et al*., 2022), etc.

For the detection stage, DL-based approaches can be grouped into single and two-stage methods (Yan *et al*., 2023). For single stage approaches, they consist in directly processing input images, for example, Single Shot Multibox Detector (SSD) and You Only Look One (YOLO). Concerning two-stage algorithms, they consist instead in generating candidate boxes from input images, for example, region based neural networks, etc. In terms of detection accuracy, the second type of approach achieves better performances, while one-stage methods are faster and more suitable for real-time applications.

For the classification stage, recent DL approaches are essentially based on Convolutional Neural Networks (CNNs), Quantum CNNs (QCNNs), Region-based CNNs (R-CNNs), and Vision Transformers (ViT).

From this perspective, the work of Yan *et al*. (2023) is proposed to improve the recognition of traffic signs under complex illumination changes. The adopted approach consists in quickly locating the ROIs from low-level feature maps, affected by these severe conditions. The quality of these images is improved using an Adaptive Gamma Correction (AGC) for image enhancement. For attention mask generation, instead of using the difference between two feature maps, the Feature Difference attention-based block is proposed to detect and recognize traffic signs under complex illumination conditions. The adopted approach outperforms the other Attention-based modules.

Another approach is adopted by So and Kim (2022). The approach is based on Mask R-CNN for traffic sign detection and classification. The network is a combination of CNNs, Regions Mask, and Inception-v3. To evaluate the performances of the proposed network, images obtained by different mapping systems are used and grouped into 103 classes of traffic signs. The detection accuracy reaches 87.6%, while the classification accuracy is about 77.5%.

An improved YOLOv3 is proposed by Gong *et al*. (2022) for the detection of small traffic signs. The fusion of the spatial pyramid pooling structure into the network improves the recognition of small signs. The dataset Tsinghua-Tencent 100K (TT100K) is used with a K-means clustering algorithm. For training purposes, the database is expanded to achieve a balance between the different classes. The results show that the mean average precision of the improved YOLOv3 (77.3%) is higher than YOLOv3 (68.9%), while the detection rate is about 22 and 27 frames per second respectively. The obtained results show also that, the best real-time performances are achieved by SSD with 42 frames per second, while the detection speed of Faster R-CNN is just about two frames per second.

In the same context, an enhanced YOLOv4 tiny network is proposed by Sharma *et al*. (2023). To eliminate the redundancy of low-frequency features and enhance computational efficiency, octave convolution is incorporated. After that, an Attention module is added, where invalid features' weights are eliminated. To improve the detection rate, a feature fusion stage is added to the structure of the simplified path aggregation network. To evaluate the obtained performances, the two datasets TT100k and Chinese Traffic Sign Detection Benchmark (CCTSDB) are used. In terms of recognition accuracy and speed, the enhanced YOLOv4 tiny network outperforms the original YOLOv4 tiny and that using NVIDIA GeForce RTX1660Ti, 16 GB RAM.

An approach based on R-CNN and YOLOv5 is adopted by Xing *et al*. (2022a). The method is proposed for the recognition of road signs from wild and foggy weather, including rainy, hazing, and lightning conditions. The preprocessing stage consists in eliminating noises through guided image filtering. Three datasets are used to test and evaluate the performances of the proposed approach (FRIDA, FRIDA2, and FROSI). For the training stage, Faster RCNN spent 14 h, against 11 h for YOLOv5. The experiments are conducted using a Core i7-8th CPU with 16 GB of RAM and NVIDIA RTX2060 GPU. The results show that YOLOv5 is better and faster in recognizing small objects and objects characterized by a high mobility speed. In contrast, the performances of faster R-CNN become very low when the traffic signs are very far from the camera.

To ensure the recognition and classification of traffic signs under severe weather conditions, another approach is adopted by Dang *et al.* (2023). The method is also based on YOLO by replacing the Global Context (GC) block and the Squeeze-and-Excitation module. Using a collection of data from various types, the new network (YOLOv5s+C3GC) achieves better performances than YOLOv7, with an accuracy of 79.2%, while the obtained accuracy of YOLOv7 is about 78%. The enhanced YOLOv5s ameliorate the recognition of traffic signs, especially those affected by blur and distance.

A different approach is adopted by Yan *et al*. (2023). A Context and Attention-based Network is used for the detection of small and occluded signs. The approach is based on the fusion of shallow and deep semantic information. After feature extraction and fusion, the signs are detected. For the used dataset, 45 categories of road signs are selected from the 221 categories of TT100K. The network achieves high performances using this dataset and in comparison, to RetinaNet, EfficientDet, YOLOv5, Faster, and Libra R-CNN.

With the advances in Quantum Machine Learning, recently Quantum Convolutional Neural Networks (QCNN) are proposed for traffic sign recognition. In this context, we find the work of Cox (2022). By embracing Quantum Mechanics' principles, this type of hybrid Neural Network encapsulates complex features for enhancing traffic sign recognition.

A quantized or binarized neural network is used by Postovan and Erascu (2023). The weights and activations of this feedforward network are mainly binary and the convolutional layers are binarized (except the first one). For hyperparameter tuning, a bottom-up approach is adopted to design the architecture, and various combinations of different values (kernel size, layers number, etc.,) are tested. Adam is used as the optimizer. The obtained accuracy on German Traffic Signs Recognition Benchmark (GTSRB) is about 96,45%, using Intel Iris Plus Graphics 650 GPU. For Belgium Traffic Signs Classification Dataset (BTSCD), the accuracy is 88.17% for the 23 selected classes of BTSCD (1818 images).

An extreme learning approach is adopted by Batool *et al.* (2022). Hence, the improved Extreme Learning Network (iELMNet) is proposed for traffic sign detection. The method is based on a CNN architecture inspired by DenseNet. Using the Challenging and Real Environment for Traffic Sign Detection (CURE-TSD), TT100K, and German Traffic Signs Detection Benchmark (GTSDB), the proposed network outperforms state-of-the-art approaches for traffic signs detection and with an average precision of 93.31, 95.22 and 99.45% respectively.

A different method is adopted by Akshaya *et al*., (2023) for the recognition of circular traffic signs. After the preprocessing stage, a shape-based approach using Hough Transform is adopted for region detection and segmentation, while the identification and classification stage is based on Convolutional Neural Networks.

For the recognition of traffic signs in complex environments, including rain, snow, and night conditions, a segmentation method is adopted by Zou *et al*. (2022). The proposed approach is based on spatiotemporal convolution. Therefore, to improve the recognition speed and accuracy, saliency detection with octave convolution is used. After the octave convolution, a lightweight and efficient spatiotemporal Network is used in addition to the octave residual model. The obtained results show that the adopted approach is efficient in improving the segmentation accuracy in comparison to other methods.

Ahangi and Möckel (2023) adopt instead a Multiexpert CNN-based approach. In their work, several experts or deep networks are combined into a decision system by averaging their weights. The training of the experts is independent and parallel across distributed computing. The optimal parameters are chosen using a validation set with a 10-fold cross-validation approach. These hyperparameters include the regularization, dropout, overlap, and epochs number. The total number of used parameters is almost 11.2 million (eight times less than the committee of CNNs approach). This gating network with switching system, for multi-experts' recognition, achieves a high accuracy of 99.10% using the GTSRB dataset.

Khan *et al*. (2023), adopt also an approach based on CNNs. A preprocessing stage is applied, including rescaling (100×100), data normalization, and augmentation. The developed network includes convolutional, pooling, and batch normalization layers, in addition to a final classification layer, based on global average pooling. The choice of parameters is based on a grid search with cross-validation, within a range of values, to find the best combination of parameters. ReLU is used as the activation function, in addition to the Adam optimizer and categorical cross-entropy. The proposed network achieves high accuracy with fewer parameters, using two datasets. The obtained performances reach 98, 41, and 92,06% on GTSRB and BTSCD respectively. The total number of used parameters is almost 2.63 million parameters. In terms of recognition accuracy and speed, the proposed network outperforms the performances of GoogleNet, AlexNet, VGG16, VGG19, MobileNetv2, and ResNetv2.

An Ensemble Learning CNN is proposed by Lim *et al*. (2023b). The adopted approach uses three Convolutional Neural Networks for the recognition of traffic signs. The used networks are DenseNet121, ResNet50, and VGG16. For ResNet50, it includes 50 layers, while VGG16 contains 16 layers. In the training stage, a certain number of layers are added to each network: A flattened layer, two dense layers, batch normalization, leaky ReLU activation, and drop out. After training separately each of the three networks, their predictions are fused after majority voting. The used datasets are first converted to grayscale and after that, histogram equalization is applied. Data augmentation is also applied to balance and increase the size of the training sets. The obtained performances of the ensemble learning are 98,33% using BTSCD.

In addition to CNNs, recently some researchers opt for Vision Transformers. In this context, a comparison between the performances of seven Convolutional Neural Networks (CNNs) and five Vision Transformers (ViT) is conducted by Zheng and Jiang (2022). Transformers are Attention Mechanism-based Networks, which are generally used in the area of natural language processing. For the evaluation, three different datasets are used: German, Indian, and Chinese datasets. The seven used CNNs are VGG16, ResNet, DenseNet, MobileNet, SqueezeNet, ShuffleNet, and MnasNet. For Transformers, the evaluated networks are ViT, RealFormer, Sinkhorn Transformer, Nyströmformer, and Transformer in Transformer. The obtained results show that CNNs outperform Vision Transformers for traffic signs classification, especially when increasing data size and using larger datasets. Furthermore, Transformers have higher computational complexity, while CNNs converge faster; have fewer parameters and higher generalization abilities for traffic signs classification. These experiments are conducted using Intel i5-9600K CPU and NVIDIA GeForce RTX 2070 GPU.

From the realized review, we find that although major advances were realized in the field of traffic sign recognition, many challenges still face the implementation of these methods in real-world scenarios (Lim *et al*., 2023a), especially using deep learning approaches, as presented in the next section.

Deep Learning Challenges

From the literature review (Bousarhane *et al*., 2021), we find that deep learning approaches have proven their high performances compared to other methods, especially CNNs. This type of network is widely used and has shown very high performances in many fields related to computer vision, specifically image recognition, and classification (SivaSai *et al*., 2021). In some cases, the accuracy obtained by CNNs outperforms human performances.

However, there are multiple challenges that still face their implementation, specifically in real-time applications and constrained resource environments (Lim *et al*., 2023a), where hardware optimization is not an option. This situation is due to the fact that DL approaches are very expensive in terms of computational load and hardware requirements because of the huge number of used parameters.

Indeed, to optimize their performances, some researchers opt for increasing CNNs' depth, which as a consequence increases the number of used parameters. In fact, these parameters exceed in many cases millions of parameters. For state-of-the-art methods, we find for example that, DenseNet121 includes almost 8 million parameters (Lim *et al*., 2023b), ResNet-50 25 million, AlexNet 61 million, ResNetv 40 million, VGG16 134 million and VGG19 138 million (Khan *et al*., 2023).

Furthermore, some approaches opt for using multiple CNNs, instead of one in order to improve the obtained results, which hence increases the calculation and reduces the training and inference speed. The problem of vanishing gradient presents another important issue related to these deep architectures. In fact, in larger and deeper networks the error usually fails to reach the further layers, which leads to the degradation of CNNs' performances.

Moreover, very deep architectures could not extract features that reflect the overall characteristics of the group and tend instead to fit the training data (overfitting), while they get poor results for unseen data. To overcome this problem, some researchers opt for big data to prevent model overfitting. However, data collection represents a hard task, which is at the same time very time-consuming, in addition to the annotation process that makes the task even more complicated.

On the other hand, data collection should take into consideration the balance between the different classes, without mentioning the adverse conditions that face the recognition process in real-world scenarios, like illumination and weather changes, fading colors, etc. In fact, CNNs' performances decrease, considerably, when they are tested using datasets that include more real-world challenging conditions (Temel *et al*., 2017).

Another important challenge, faced by deep learning approaches, is the time required to train the network. Indeed, this training could take several days or weeks, which makes the validation process a very hard task, especially when using limited hardware resources and big data for training.

To overcome some of these challenges, especially those related to real-time recognition of traffic signs, we find a certain number of works realized in this context. From these works, we can mention the approach of Li *et al*. (2022), based on Faster R-CNN, for traffic sign detection. The improved network uses ResNet50-D for feature extraction and Attention-guided Context Feature Pyramid Network (ACFPN). The training is based on transfer learning and data augmentation. In comparison to state-of-the-art methods, the enhanced Faster R-CNN achieves higher performances using two datasets, CCTSDB and TT100K. For the CCTSDB database, the mean Average Precision is almost 99.5%, while the detection rate is about 29.8 frames per second.

A YOLOv4 tiny based network (TSR-YOLO) is proposed by Song and Suandi (2023) for Chinese road sign detection in complex scenarios. These conditions include severe weather and light intensity changes. To improve the accuracy of the enhanced algorithm, k-means++ clustering, attention mechanism, and dense spatial pyramid pooling are added. The objective is to improve the extraction of key features and handle more effectively the fusion of local and global ones. Using the new dataset CCTSDB2021 (Zhang *et al*., 2022), the obtained results show that, the adopted approach is more efficient and fast in comparison to the original YOLOv4 tiny. The performances of the network achieve a detection accuracy of 96.62% and a mean average precision of 92.77%, with a detection rate of about 81 frames per second.

YOLOv4 is also used by Tong *et al*. (2023), in addition to the improved Oriented Fast and Rotated Bridge (ORB) for feature extraction and description. The proposed approach is a cloud-based multi-sensor fusion system for the recognition of vehicles, pedestrians, and five types of traffic signs. The obtained results show that the system has a good recognition of traffic signs with an accuracy of 96,67%, where each sign is recognized within 2 sec.

The proposed work of Zhu *et al*. (2022) aims to evaluate the performances of YOLOv5 in comparison to SSD for traffic sign recognition. After the process of training, validation, and testing with the used datasets, the results show that YOLOv5 achieves an accuracy of 97.70%, while SSD obtains just 90.14%. For the categories with a smaller number of samples, the accuracy of SSD decreases to only 78.32%. Concerning the recognition speed, YOLOv5 is faster with 30 frames per second, against 3.49 for SSD (using GPU).

The work of Xing *et al*. (2022b) confirms this conclusion. Hence, after conducting many experiments concerning the different approaches for traffic sign detection, the results show that YOLOv5 is more suitable for real-time applications. The authors present also a defogging approach for traffic signs images based on a guided filtering algorithm.

From the presented works, we find that hardware optimization is generally used to accelerate the recognition process because the complexity and the size of DL networks affect considerably the processing time and memory requirements (Khan *et al*., 2023). For that, real-time recognition, using constrained resource environments, still presents a huge challenge for DL based approaches (Lim *et al*., 2023a).

From this perspective, the aim of our work is to ensure an efficient and real-time classification of traffic signs, using a Deep Learning approach. The main objective of the proposed approach is to create a certain balance between the model complexity and the classification accuracy through a computationally efficient convolutional network. The adopted approach is presented in the next section.

Materials and Methods

To ensure traffic signs classification, different types of approaches are proposed by researchers, including "traditional" and Artificial Intelligence-based methods. In terms of recognition performances, DL methods have proven their superiority, especially CNNs.

Generally, the dimensionality of input images is high (thousands of pixels), so the main role of the convolutional layers is to map this raw data to a feature space of reduced dimensionality and complexity through new representations of the original images.

In addition to the dimensionality reduction, the main advantage of these layers is the preservation of the relationship between the pixels by keeping their locations' order through convolutions. However, this interconnectivity is lost in the hidden layers, because the generated maps are transformed to a single 1D vector.

Which represents the main drawback of the fully connected hidden layers.

Keeping the connections between neighboring pixels is very helpful for improving classification accuracy and generalization abilities in general. From this perspective, the objective of our work is to adopt an optimal architecture that preserves and optimizes this connectivity between the different units of the network.

To achieve this goal, two receptive fields architectures are combined. Therefore, the proposed approach is a combination of LeNet and Receptive Fields Neural Networks (RF-NNs) architectures. The goal is to obtain a lightweight architecture with minimal complexity, fast convergence, and high generalization abilities.

Accordingly, two architectures are proposed, where each includes two principal Receptive Fields modules. A first features extraction module based on LeNet architecture. While the second hidden module is based, for the first architecture, on Image Receptive Fields Feedforward Network (IRF-NN). For the second architecture, the hidden module is based instead on a Partially-Connected Neural Network (PCNN).

Receptive Fields Based Architectures

In the adopted approach, the first features extraction module is based on LeNet, which is a state-of-the-art network for handwritten digit recognition. Although its high performance, the network is characterized by the simplicity and the efficiency of its lightweight architecture.

LeNet has two feature maps construction layers, followed each by a dimensionality reduction layer, while the final one is a fully-connected, with ten classes of digits. The size of input images is fixed to 16×16 pixels, for the 1st version of the network and 32×32 for the last version (LeNet-5), as presented in Figure 1. For the first layer, it has just six kernels of 5×5 size, followed by an average layer with a pooling size of 2×2. The third layer has 16 filters of 5×5 size, followed also by a subsampling layer.

For the hidden layers, they are based on full connections between the nodes, while their input is the 1D vector, generated by flattening the 2D output maps of the convolutional layers. However, this one-dimensional representation of features induces the loss of all spatial information in these hidden layers.

Handling multi-channeled inputs and preserving the spatial dependence within the maps represent the main objective of the proposed approach, in order to enhance the classification accuracy. To achieve this goal, Receptive Fields' hidden layers are adopted. Hence, two types of RF-Networks are used (IRF-NN and PCNN).

Unlike conventional Feedforward Networks, the IRF-NN uses the input images directly without prior feature extraction (Smagghe *et al*., 2013). Furthermore, the hidden units are partially connected to all the previous units, instead of being totally connected to these nodes (Fig. 2).

The activation vector H is the dot product of the input image I with g_i (the two-dimensional weight vectors). For each RGB channel, the color is selected with random amplitude a_c. The activation function used for the output is a sigmoid function (1). Where i is the neuron and x, y are the coordinates of the image pixels (Smagghe *et al*., 2013):

$$H_i = tanh\left(\sum_{c}\sum_{x,y} a_{ic} I_c\left(x,y\right).g_i\left(x,y\right)\right) \tag{1}$$

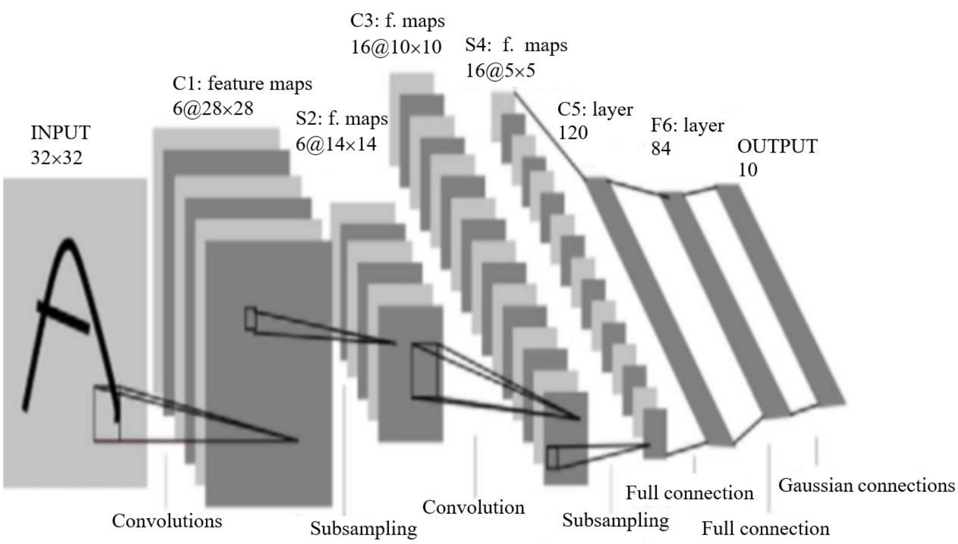


Fig. 1: LeNet-5 architecture (Kayed *et al*., 2020)

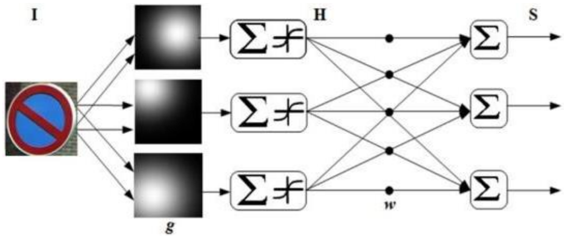


Fig. 2: IRF-NN architecture (Smagghe *et al*., 2013)

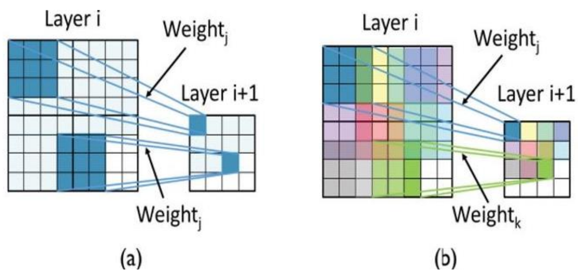


Fig. 3: Shared vs Unshared weights (Achararit *et al*., 2018)

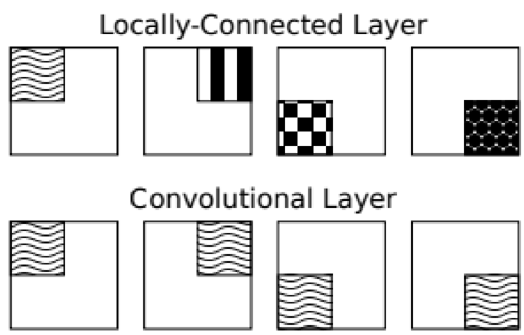


Fig. 4: Generated maps in LC layers and CNNs (Chen *et al*., 2015)

IRF-NN is a Feedforward Network that is more suitable for image recognition. It ensures a fast and accurate classification, by dealing directly with original images and without the need of a preprocessing stage to extract features from raw data. Furthermore, the network ensures a fast-training process using a limited number of parameters.

For the Partially or Locally-Connected Network (PCNN), unlike conventional CNNs, different Receptive Fields detectors are applied to each region of the image (Achararit *et al*., 2018), generating hence more sophisticated features to enhance the classification accuracy (Fig. 3).

Contrary to fully-connected layers, convolutional ones use locally connected kernels instead of full connections between the layers. They are essentially based on weight sharing, which reduces significantly the total number of needed parameters. The convolutional kernels consist in extracting feature maps from input images (2):

$$f_l^k(p,q) = \sum_c \sum_{x,y} i_c(x,y) e_l^k(u,v) \tag{2}$$

where, $i_c(x, y)$ represents an element of the input, $e_l^k(u, v)$ is a kernel element in a specific layer, and $f_l^k(p, q)$ is an element of the extracted feature map.

In comparison to PCNNs, CNNs induce fewer parameters, because the generated map belongs to one kernel, while in PCNNs each map is generated from different kernels (Chen *et al*., 2015), which makes this type of network more efficient in extracting local features and in ensuring a better and faster learning process (Fig. 4).

Based on these two types of RF architectures, two different networks are proposed to ensure a fast and efficient classification of traffic signs (RF-CNN and RFPCNN). The depth and the size of the networks are selected through hyperparameter tuning using a validation set.

Hyperparameters Tuning

The dimension of input data in LeNet is 16×16 (grayscale images). Unlike LeNet architecture, which is based on one-dimensional vectors in the hidden layers, the two modules of our approach are based instead on feature map extraction and concern more complex types of objects (traffic signs in RGB color space).

For that, we have adopted an input size of 32×32 as used in LeNet-5 (Yu *et al*., 2015), while adding the three-color channels (32×32×3). The objective is to create a certain balance between the input images and the generated feature maps from the two modules.

Concerning the number of layers, for our first module, we have used the same number of convolutional layers adopted in the original LeNet architecture. Although the simplicity of its structure, this network has proven its high efficiency and performance.

To prevent the loss of spatial detail information, our second module is based, for the first network (RF-CNN), on the IRF-NN architecture. Hence, the feature maps extracted from the first LeNet-based module represent the inputs to this hidden module. While two Receptive Fields hidden layers are added instead of the one used in IRF Network.

Concerning the selection of the number of units, choosing the appropriate number is a very challenging task. In the literature review, we find that there are in fact two main strategies to choose a network's size. These two principal techniques, used to ensure good generalization abilities for the networks, are more specifically network growing and network pruning (Bortman and Aladjem, 2009).

Network growth consists in using a light architecture and then continually adding more neurons based on a threshold until convergence. The approach begins then with a very small network, with one or two hidden units, and adds progressively more, when it is needed, to improve the learning capabilities of the network.

While pruning techniques consist instead in adopting a very large network and progressively reducing the number of used neurons (sensitivity methods) or deactivating their weights through regularization (penalty term methods). Genetic algorithms are also used in this context to generate smaller networks from more complex ones (parent networks).

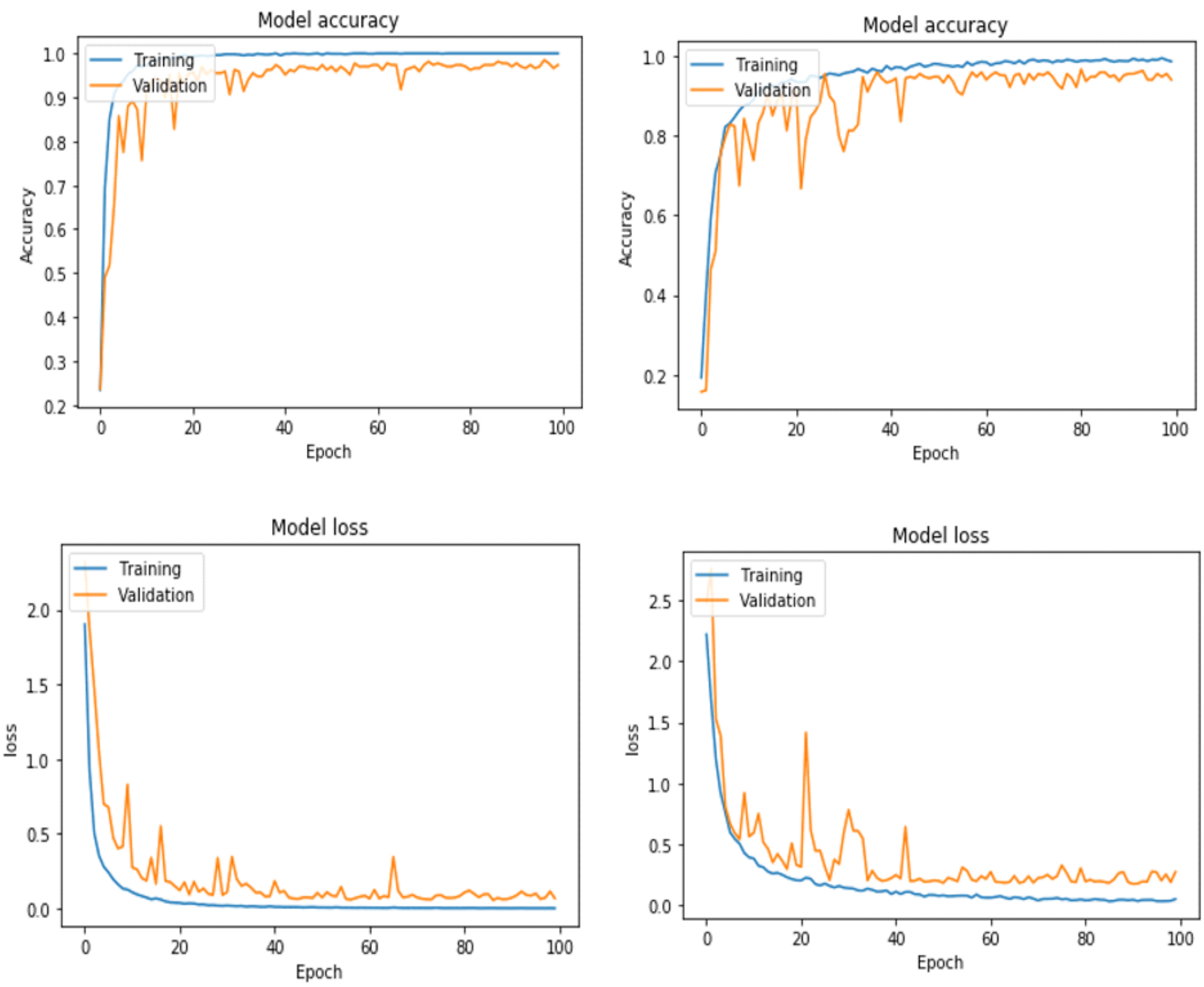
In effect, pruning techniques are very time-consuming and computationally expensive, which makes the training and the validation task even more hard and complex. Furthermore, these methods could lead to an excessive and unnecessary number of units and they are generally subject to overfitting and poor generalization problems.

To adapt the network's complexity to the classification accuracy more efficiently, we have adopted the first strategy for hyperparameter selection. Our constructive approach is then based on adopting an undersized network with just two kernels, for the two convolutional layers, while increasing the filters' number and size each time until convergence.

Hence for the first module, to choose the kernels' size that accentuates best the unique features of input images, we have evaluated the training performances, by changing each time the size of the used kernels with a stride of 1, for 100 iterations (epochs). The evaluated kernels are 2×2, 4×4, 8×8, and 16×16.

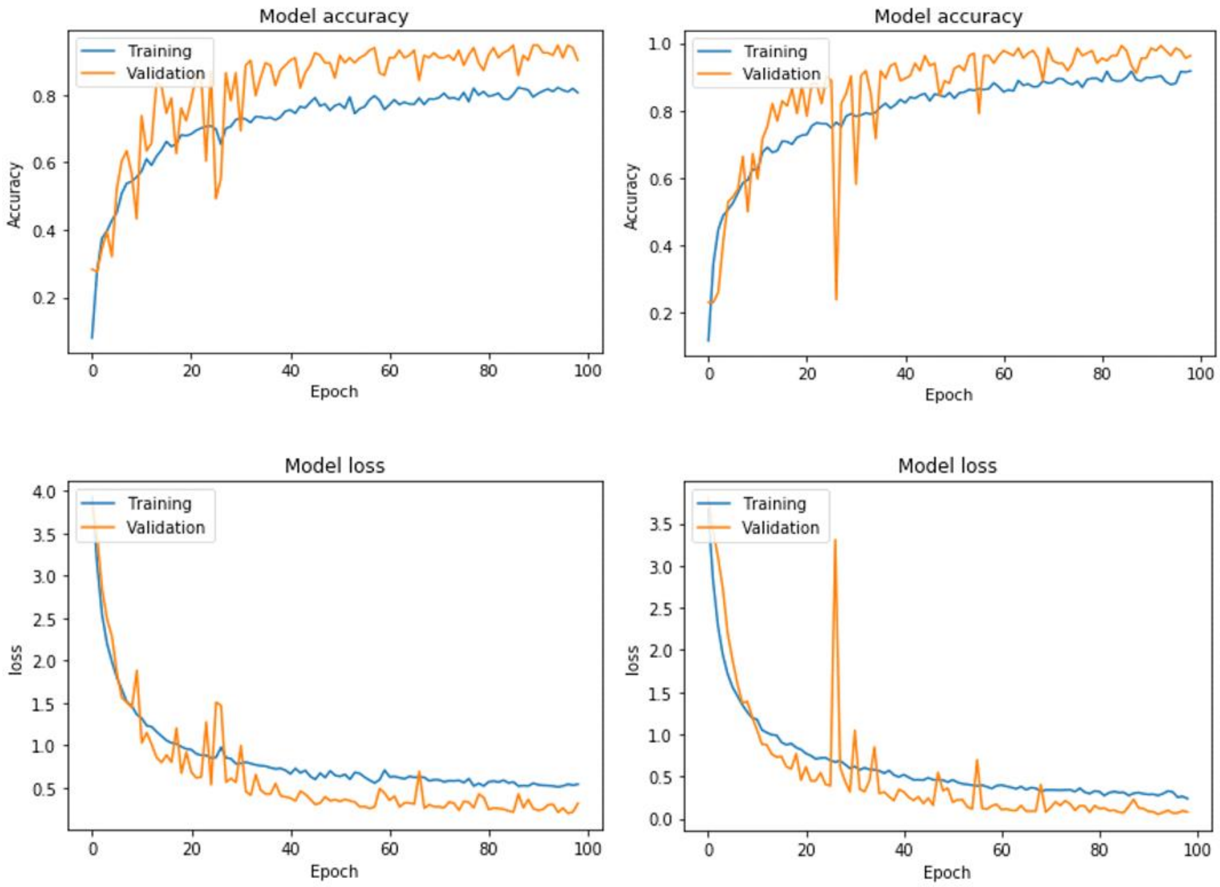
For the training dataset, it includes more than 1000 images (90% for training and 10% for validation), extracted from CURE-TSR. To get a representative validation set, we have used the cross-validation technique, which consists in repeating the division of the data, to guarantee the randomness of the selected set.

The obtained results show that the small filters (2×2 and 4×4) present better results, especially concerning the unseen data (validation set), as presented in Fig. 5.



(A) (B)

Fig. 5: Obtained accuracy and loss; (A) 2×2 kernels, (B) 4×4 kernels



(A) (B)

Fig. 6: Obtained accuracy and loss; (A) 8 kernels; (B) 16 kernels

For the number of used kernels, as for the kernels' size, we have evaluated the training performances using different numbers of filters (2, 4, 8, and 16 kernels).

We find that the accuracy of the training and the validation sets increases when adding more kernels. Figure 6 shows the accuracy and loss curves when adding 8 and 16 kernels.

To ensure the performance of the model, regardless of training data, we have used Dropout (Shen and Shafiq, 2018). This technique randomly deactivates some nodes in the training process, which helps to prevent overfitting. The method is generally used in fully-connected layers, but it has proven also its performance in convolutional layers.

To select the percentages that should be dropped out from the training process, we have compared the training performances of the two convolutional layers when dropping respectively (0.2, 0.2), (0.2, 0.4), (0.4, 0.2), and (0.4, 0.4).

We find that Dropout ameliorates the obtained results for the training and the validation sets. We notice also that the results are almost similar for the used percentages.

Figure 7 shows the obtained results when dropping (0.4, 0.2) and (0.4, 0.4). For the second module of the proposed approach, as for LeNet-5 architecture, we have used just two hidaden layers based on two different types of receptive field networks.

According to the obtained results, we have adopted the (2×2) filter for the two convolutional layers in the first network (RF-CNN). To ameliorate the obtained performances, we have used 20 kernels for each of these two layers, with a Dropout of (0.4, 0.4). To find the optimal number of hidden units, we have evaluated the performances of the training process, by changing each time the number of units (2, 4, 8, and 16).

When analyzing the obtained results, we find that using more nodes helps to obtain higher accuracies much faster than when using just two or four nodes. Figure 8 shows the obtained performances when adding 8 and 16 nodes respectively to the network.

DOI: 10.3844/jcssp.2023.1263.1282

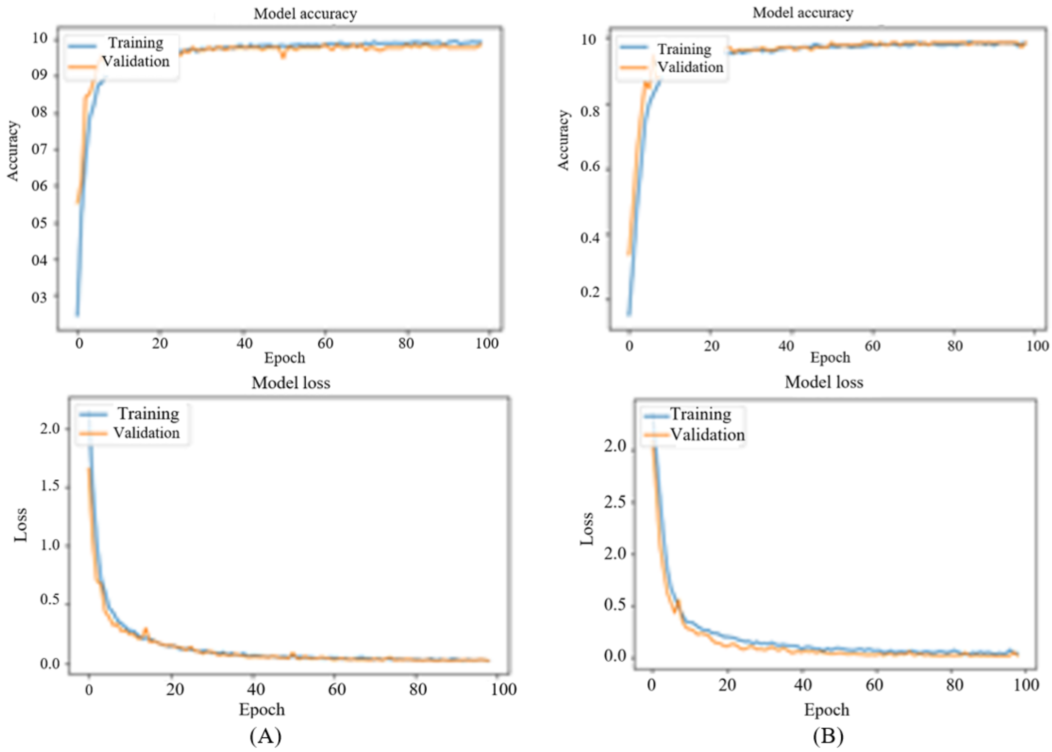


Fig. 7: Accuracy and loss; (A) 0.4, 0.2 and (B) 0.4, 0.4 dropout

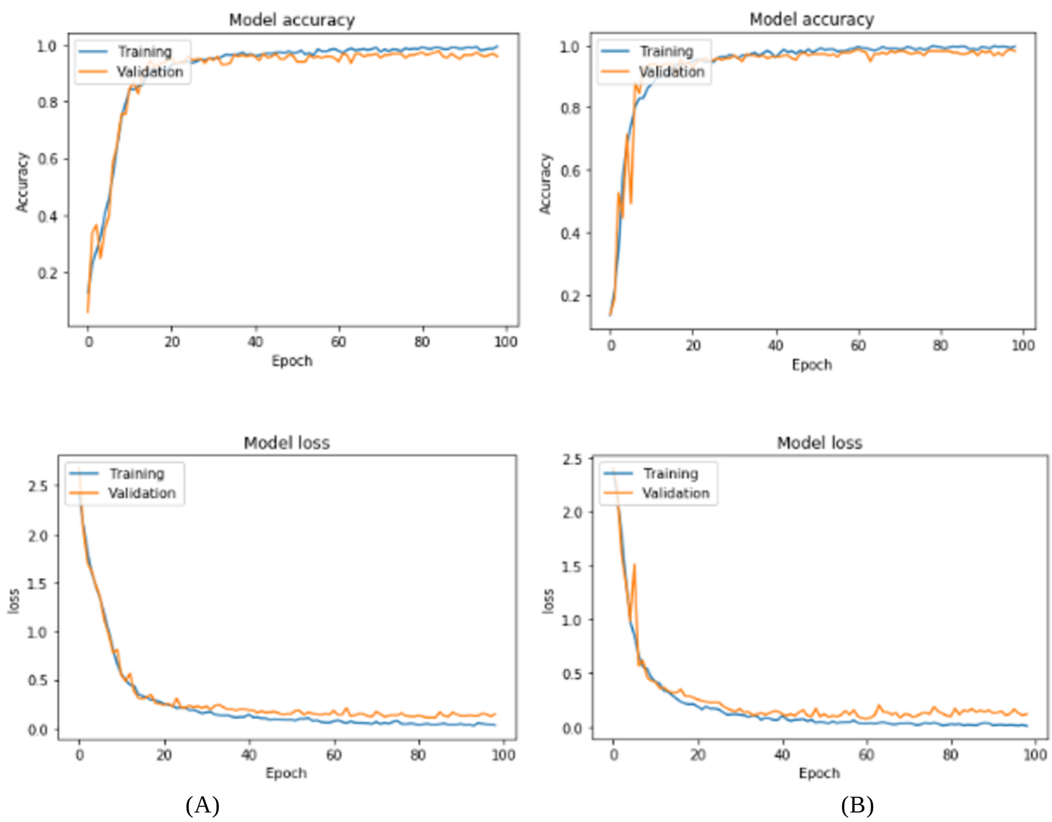


Fig. 8: Obtained accuracy and loss; (A) 8 nodes and (B) 16 nodes

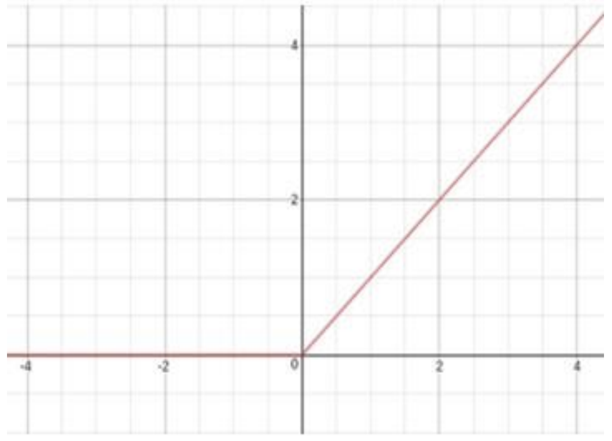


Fig. 9: Rectified Linear Unit (ReLU)

To reduce the maximum network complexity and accelerate further the process, just 10 nodes are added. As for the convolutional layers, we have also used Dropout to train only randomly selected nodes from the hidden nodes to face overfitting problems. The deactivated nodes are (0.2, 0.2).

The behavior of each of these units is determined by a decision function. There are, in fact, multiple activation functions available. In recent years, Rectified Linear Unit (ReLU) (3) represents the most popular activation function (Fig. 9), used in deep CNNs, because it ensures a faster training process (Glorot *et al*., 2011). Furthermore, it ensures a better transmission of the error:

$$f(x) = \begin{cases} 0 \; for \; x \geq 0 \\ x \; for \; x \geq 0 \end{cases} \tag{3}$$

For that, we have used ReLU as the activation function for both the convolutional and hidden layers of the first network. Finally, to achieve the predictions at the final layer, we have used the Softmax function (Kouretas and Paliouras, 2019).

For the second network, RF-PCNN induces more parameters than a traditional CNN. For that, we have added just 10 kernels, with a filter size of 4×4 for the two first layers (Fig. 5). In order to reduce the network complexity, just eight and seven kernels are added to the hidden layers respectively.

To face overfitting problems, a Dropout of (0.4, 0.2) is used in the first two layers, while the randomly deactivated nodes from the hidden layers represent (0.2, 0.1).

For the activation function, we have used Scaled Exponential Linear Units (SELU), proposed by (Klambauer *et al*., 2017) (4). This activation function induces self-normalizing properties (Paoletti *et al*., 2018) and efficiently learns robust features, which accelerates enormously the training convergence. Finally, to get the predictions at the last layer, the sigmoid function is used (5):

$$Selu(x) = \begin{cases} \lambda x & , x > 0 \\ \lambda\alpha\left(e^{x} - 1\right), & x \leq 0 \end{cases} \tag{4}$$

$$\left(\lambda = 1.0507 \; and \; \alpha = 1.6733\right)$$

$$g(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

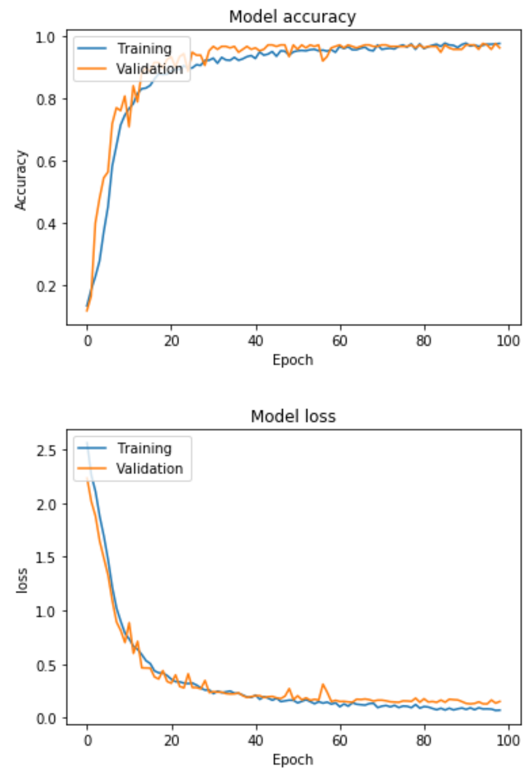


Fig. 10: Training and validation curves using the 1st network

Networks' Generalization Abilities

In order to ensure the performances of the two networks regardless of input data, the process of regularization is used by applying Batch Normalization (Chen *et al*., 2017) to normalize the output of each layer. Figure 10 shows the performances of the first network after batch normalization.

Furthermore, to face vanishing gradient problems, we have used the Cross-Entropy (Zhou *et al*., 2019) learning rule. This function is more sensitive to the error and provides higher performances during the training process (6):

$$f^t(W) = -\sum_{i=1}^{m} \tilde{y}_i log(y_i) \tag{6}$$

W = Training weight matrix
M = Total number of classes
y_i: = i^{th} prediction class
$\tilde{y}_i$: = i^{th} true class of training samples

The learning process is an optimization problem that aims to reduce the max error function. However, this function is characterized by the presence of several local minima. Determining the optimal or global minima, in this cost function surface, needs necessarily an optimal selection of the initial weights. Using null, equal, or high initial weights lead generally to the saturation of the network. Where comes the importance of randomly selecting these initial weights with small values?

Furthermore, finding the optimal steps for the learning rate presents also another challenging problem. Using large rates leads to poor generalization, while small values make the learning process very slow. To solve this problem, we have started the training with a relatively small learning rate (0.001), while using an adaptive learning optimizer (Adam). This optimizer is computationally efficient and has low memory requirements (Lim *et al*., 2023b).

To enhance the generalizability of the networks, a batch size of 32 is used, because larger batches affect considerably the error estimation (Lim *et al*., 2023b). Finally, to find the optimal starting point of the initial weights, we have retrained the networks each time with new sets of randomized weights, to choose the initial weights that ensure a high recognition accuracy, on the validation set and a fast convergence speed. This approach helps to optimize the performance, while it reduces the need to excessively increase the network's size. These weights are randomly set at first and continually updated, during the training, using He Uniform Initializer (Henry *et al*., 2018).

Results

To evaluate the performances of the two models (RFCNN and RF-PCNN), in relation to different classification problems and challenges, we have used an extracted set from "The Challenging Unreal and Real Environments for Traffic Sign Recognition" (CURE-TSR) database (Temel *et al*., 2017), which contains many levels of adverse conditions including snow, haze, rain, darkness, blur, etc.

Furthermore, instead of opting for data augmentation, which is widely used by researchers in this field, we have adopted the opposite approach based on reducing the number of training samples. In effect, in the literature review, we find that deep learning is tightly related to big data. However, in the proposed approach, we have adopted instead a "Data Reduction" process. The adopted approach is based on structuring and diversifying the training set using a minimum number of instances.

Hence, for training our two networks, we have extracted a training set, from CURE-TSR, that includes 11 classes with 1331 images. Figures 11-12 show the types and the numbers of used traffic signs.

Figures 13-14 show the loss and accuracy curves obtained during the training of the first and second networks using 10% for validation and 90% for the training process.

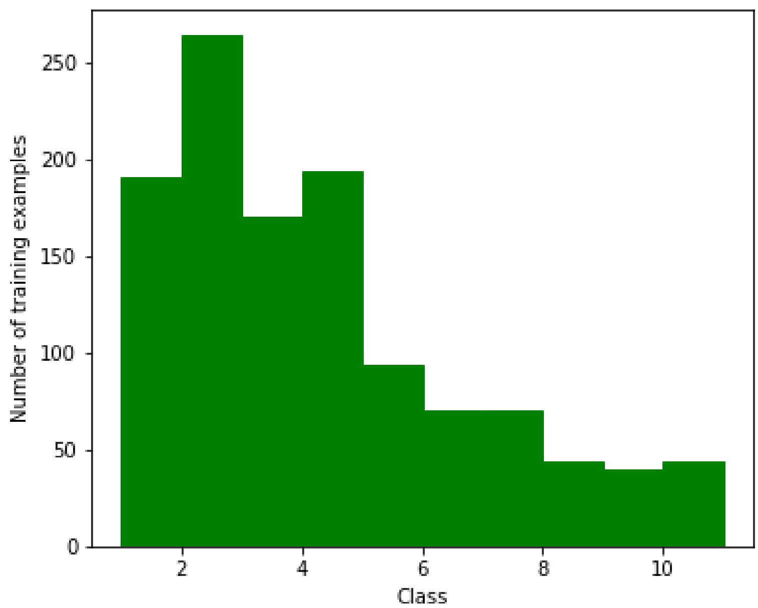


Fig. 11: Balance between the 11 training classes

Fig. 12: Types and numbers of used traffic signs

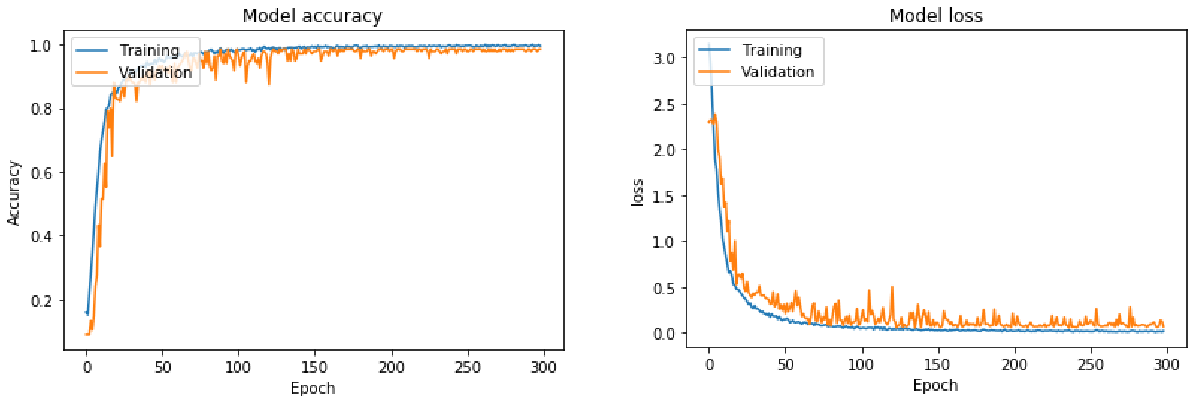


Fig. 13: Loss and accuracy obtained during the training (1st network)

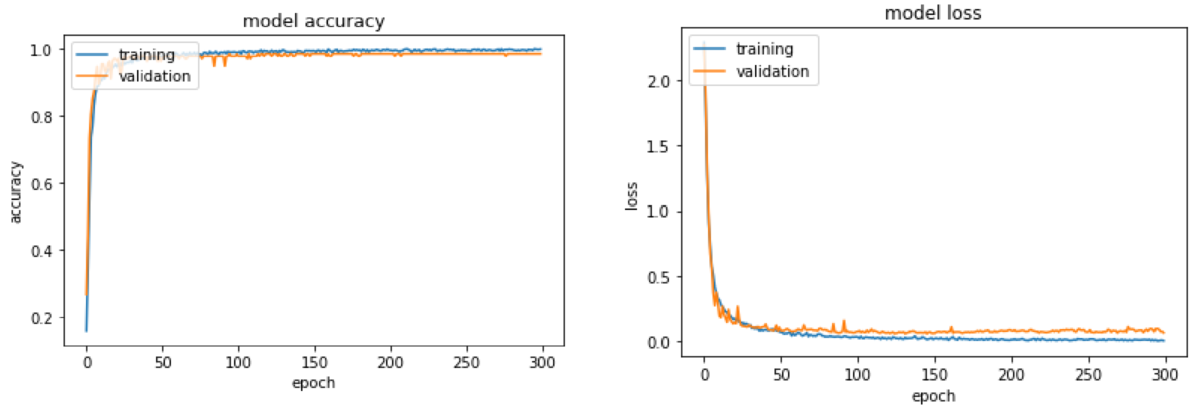


Fig. 14: Loss and accuracy obtained for the training (2nd network)

Fig. 15: Traffic signs affected by some adverse conditions

DOI: 10.3844/jcssp.2023.1263.1282

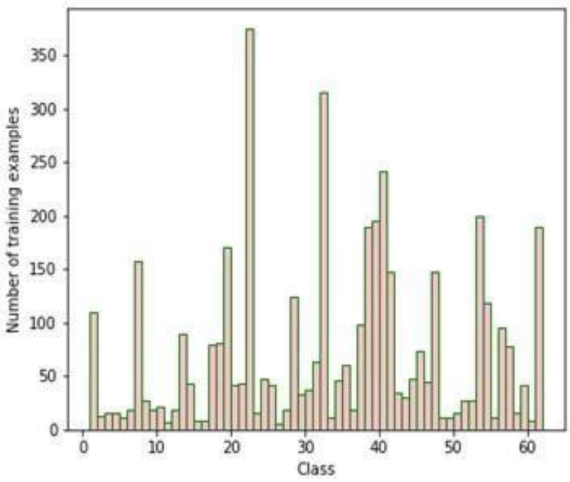


Fig. 16: Balance between classes in BTSCD

For the testing dataset, it includes 4 039 images, extracted from the CURE-TSR database. Fig. 15 shows the number of traffic signs for each of these classes.

For the classification of the 11 classes (affected by some adverse conditions), the adopted approach has obtained an accuracy of 96. 60% using the first network. While for the second one, the accuracy obtained reaches almost 97.59%.

To further evaluate the performances of the adopted approach, we have tested our two networks using another public dataset.

To facilitate the benchmark of traffic sign recognition methods, many public datasets have been created. Within these datasets, there are databases that respect the Vienna Convention on Road Signs, for example, GTSRB (Stallkamp *et al*., 2011), BTSRD (Mathias *et al*., 2013), Austrian Dataset ATSD (Maletzky *et al*., 2023), etc. On the other hand, we find that some other countries have their own standards. In this context, we find for example LISA in the USA (Møgelmose *et al*., 2015), TT100K in China, etc.

To evaluate the performances of the adopted approach, we have used Belgium Traffic Signs Classification Dataset (BTSCD), which is widely used by researchers in this domain. This dataset includes 62 classes, with more than 4 500 images for the training and almost 2 500 images for the testing set (Figs. 16-18). The accuracy graph and the loss curve obtained by applying our first network to BTSCD are presented in Fig.19.

The classification accuracy achieved, on the BTSCD dataset, by applying our first network is 97.93%, where each sign is classified within less than 500us. To test even more the impact of data reduction on classification accuracy, we have reduced the number of training images from 4 575-4 328. The obtained results show that the approach reaches a higher accuracy with 98.01% for the first network.

Fig. 17: Types and number of traffic signs in BTSCD

DOI: 10.3844/jcssp.2023.1263.1282

Fig. 18: Signs affected by some challenges from BTSCD

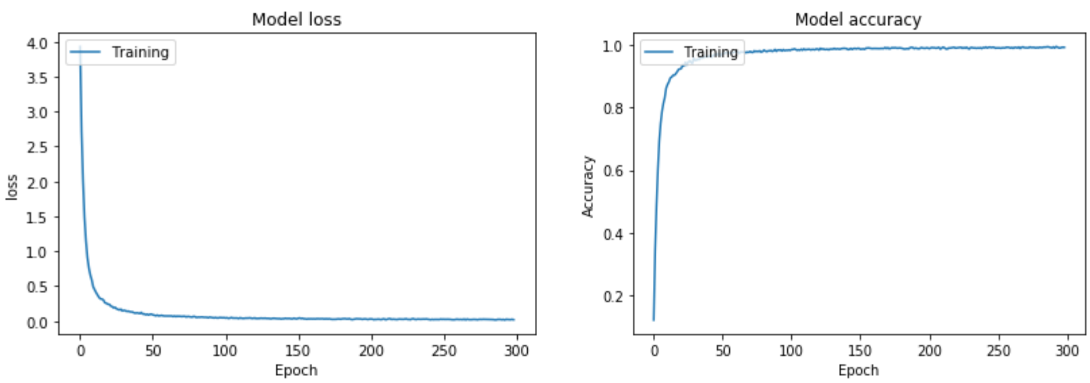


Fig. 19: Accuracy and loss obtained using BTSCD (1st network)

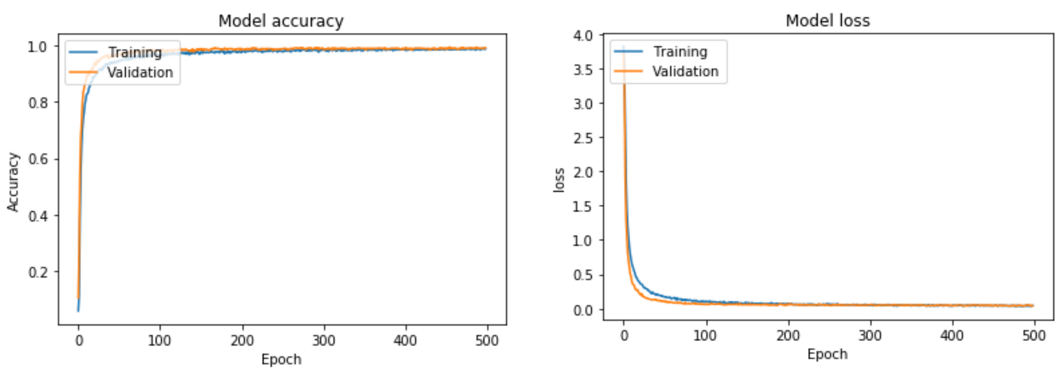


Fig. 20: Accuracy and loss curves using BTSCD (2nd network)

To evaluate the performances of the second network, we have also trained the RF-PCNN network using BTSCD (80% for training and 20% for validation). The accuracy and loss curves obtained for the training and the validation sets are presented in Fig. 20.

Discussion

Table 1 shows a comparison between our second network (RF-PCNN) and some state-of-the-art approaches, including the works that are particularly designed for the real-time classification of traffic signs.

Table 1: Comparison between our approach and some state-of-the-art classification approaches

Reference	Approach	Accuracy %	Time	Parameters
Arcos-García *et al*. (2018)	CNN and Transformer	98.87	4280 ms	14 629 801
Huang *et al*. (2017)	HOGv+ELM	98.62	1.46 ms	-
Our approach	RF-PCNN	98.49,	450 us	12 K
Zaibi *et al*. (2021)	Enhanced LeNet-5	98.37	-	0.38 million
Lim *et al*. (2023b)	Ensemble Learning	98,33	-	160 million
Saouli *et al*. (2018)	ELM with HOG, Gabor and CLBP	98.30	30 ms	-
Jurišić *et al*. (2015)	OneCNN	98.17	2 ms	-
Li *et al*. (2019)	Small CNN	98.10	705.10 ms	-
Saouli *et al*. (2018)	SVM	97.15	2220 ms	-
Mehta *et al*. (2019)	CNN+Adam	97.06	-	10 million
Khan *et al*. (2023)	Lightweight CNN	92,06	-	2.63 million
Postovan and Erascu (2023)	Binarized Neural Network	88.17	-	-
Fang *et al*. (2022)	MicronNet-BF	82.12	-	0.4 million

In comparison to the presented methods from Table 1, the proposed approach reaches a high accuracy of 98.49% using BTSRD, while it ensures a fast classification with almost 450us per image (2.0 GHz CPU) and that using a very limited number of parameters (12K).

For the other presented methods, we find that CNNs reach very good performances. Hence, the single CNN with Spatial Transformers (Arcos-García *et al*., 2018) achieves a higher accuracy of 98.87%, while the number of used parameters exceeds 14 million, with an inference time of almost 4280 ms/ image (using GPU).

We find also that, Extreme Learning Machine (ELM) with a Histogram of Oriented Gradient variant (HOGv) reaches almost the same performances (98.62%), while it has a major advantage concerning the inference speed, which is less than 1.5 ms per image (Huang *et al*., 2016).

An accurate classification is also ensured by the Enhanced LeNet-5 (Zaibi *et al*., 2021) with 98.37%, while the number of parameters is considerably decreased, with almost 0.38 million against the 10 million of the Spatial Transformer CNN (Arcos-García *et al*., 2018).

We find also that, using Extreme Learning with the combination of multiple features like HOG, Gabor, and Compound Local Binary Pattern (CLBP) (Aziz and Youssef, 2018) decreases the obtained accuracy and time to 98.30% with 30 ms/step. While using ELM with HOGv reaches higher performances (98.62% with 1.46 ms/step) (Huang *et al*., 2016).

The obtained results show that CNNs could ensure a fast classification process, as the performances achieved by One CNN (Jurišić *et al*., 2015). Where the inference time is 2 ms per image and the accuracy is 98.17%. However, the process is based on GPU unlike the performances obtained by HOGv ELM, which ensures a fast and accurate classification (1.46 ms) using CPU instead.

We also notice that even using GPU, the inference time of some CNN-based approaches remains high in comparison to other methods. In this context, the approach based on TS-Module and global average final layer (Li *et al*., 2022) reaches high performances (98.1%). However, each step is realized within more than 705ms using GPU.

Furthermore, Table 1 shows that Machine Learning-based approaches such as SVM reach also good accuracies (97.15%), while the inference time of the proposed approach is almost 2100 ms per image (Aziz and Youssef, 2018).

In contrast, Table 1 shows that increasing CNN's depth and size does not necessarily guarantee better performances. Hence, for the CNN based on Adam, ReLU, and Softmax functions (Mehta *et al*., 2019), an accuracy of 97.06% is achieved using more than 10 million parameters with CPU. While a higher accuracy of 98.37% is achieved, by the Enhanced LeNet-5 which uses just 0.38 million parameters. In the same context, we find that the CNN adopted by Khan *et al*. (2023) achieves an accuracy of 92,06% while using a higher number of parameters (2.63 million).

Using more than 160 million parameters, we find that the Ensemble Learning CNN, proposed by Lim *et al*. (2023b), reaches a high accuracy of 98,33% using BTSCD. For the last presented approaches, the Binarized Neural Network used by Postovan and Erascu (2023) achieves an accuracy of 88.17% (for 23 classes of BTSCD). For MicronNet-BF (Fang *et al*., 2022), which is based on factorization and batch normalization, it achieves 82.122%, while using a relatively small number of parameters (0.44 million), with GPU.

Conclusion

The standard or conventional CNNs architecture is widely used by researchers, whereas deeper architectures gain more popularity for solving different types of recognition and classification problems. However, many factors affect the efficiency, speed, and generalizability of these deep learning networks.

In effect, this architecture includes typically several convolutional kernels applied to the three channels of input images, followed by pooling layers. The hidden ones are totally based on full connections between the different nodes. To further enhance the obtained performances, we find that increasing CNNs' depth & size is generally the adopted option, although the fact that convolutional and fully-connected layers are very time-consuming and computationally expensive.

From this perspective, our approach calls into question the nature itself of these networks, by adopting instead a Receptive Field architecture, with reduced complexity and higher performances. The goal is to ensure the efficiency and speed required for the recognition process in real-time applications and constrained resource environments.

The obtained results show that the adopted approach helps to reach state-of-the-art methods' performances with an accuracy of 98.49%, using the public dataset BTSRD. Furthermore, the proposed architecture helps to ensure a very fast classification process of almost 500us per image.

Acknowledgment

We would like to thank the editor and the reviewers for their constructive comments and helpful suggestions to improve the quality of the paper.

Funding Information

The authors have not received any financial support or funding to report.

Author’s Contributions

Both authors have equally contributed to this manuscript.

Ethics

The authors declare that there is no conflict of interests or ethical issues. The authors have read and approved the final manuscript.

References

Ahangi, A., & Möckel, R. (2023). Switching network for mixing experts with application to traffic sign recognition. *Multimedia Tools and Applications*, 1-24. https://doi.org/10.1007/s11042-023-14959-0

Achararit, P., Hida, I., Marukame, T., Asai, T., & Hara-Azumi, Y. (2018). Structural exploration of stochastic neural networks for severely-constrained 3D memristive devices. *Nonlinear Theory and Its Applications, IEICE*, *9*(4), 466-478. https://doi.org/10.1587/nolta.9.466

Akatsuka, H., & Imai, S. (1987). *Road signposts recognition system* (No. 870239). SAE Technical Paper. https://doi.org/10.4271/870239

Akshaya, M., Srikanth, A., Diana, D. C., & Hema, R. (2023, March). Deep CNN based Road Sign Recognition for Vehicles using Raspberry Pi 3B. In *2023 International Conference on Innovative Data Communication Technologies and Application (ICIDCA)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICIDCA56705.2023.10099553

Arcos-García, Á., Alvarez-Garcia, J. A., & Soria-Morillo, L. M. (2018). Deep neural network for traffic sign recognition systems: An analysis of spatial transformers and stochastic optimization methods. *Neural Networks*, *99*, 158-165. https://doi.org/10.1016/j.neunet.2018.01.005

Aziz, S., & Youssef, F. (2018). Traffic sign recognition based on multi-feature fusion and ELM classifier. *Procedia Computer Science*, *127*, 146-153. https://doi.org/10.1016/j.procs.2018.01.109

Batool, A., Nisar, M. W., Shah, J. H., Khan, M. A., & El-Latif, A. A. A. (2022). iELMNet: Integrating novel improved extreme learning machine and convolutional neural network model for traffic sign detection. *Big data*. https://doi.org/10.1089/big.2021.0279

Bortman, M., & Aladjem, M. (2009). A growing and pruning method for radial basis function networks. *IEEE Transactions on Neural Networks*, *20*(6), 1039-1045. https://doi.org/10.1109/TNN.2009.2019270

Bousarhane, B., Bensiali, S., & Bouzidi, D. (2021). Road signs recognition: state-of-the-art and perspectives. *International Journal of Data Analysis Techniques and Strategies*, *13*(1-2), 128-150. https://doi.org/10.1504/IJDATS.2021.114672

Chen, L., Zhao, G., Zhou, J., & Kuang, L. (2017, November). Real-time traffic sign classification using combined convolutional neural networks. In *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)* (pp. 399-404). IEEE. https://doi.org/10.1109/ACPR.2017.12

Chen, Y. H., Moreno, I. L., Sainath, T., Visontai, M., Alvarez, R., & Parada, C. (2015). Locally-connected and convolutional neural networks for small footprint speaker recognition.

Cox, D. (2022, August). Experimental Design of a Quantum Convolutional Neural Network Solution for Traffic Sign Recognition. In *Proceedings of SAI Intelligent Systems Conference* (pp. 605-617). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-16072-1_44

Dang, T. P., Tran, N. T., To, V. H., & Tran Thi, M. K. (2023). Improved YOLOv5 for real-time traffic signs recognition in bad weather conditions. *The Journal of Supercomputing*, 1-19. https://doi.org/10.1007/s11227-023-05097-3

Fang, H. F., Cao, J., & Li, Z. Y. (2022). A small network MicronNet-BF of traffic sign classification. *Computational Intelligence and Neuroscience*, *2022*. https://doi.org/10.1155/2022/3995209

Glorot, X., Bordes, A., & Bengio, Y. (2011, June). Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics* (pp. 315-323). JMLR Workshop and Conference Proceedings. http://proceedings.mlr.press/v15/glorot11a

Gong, C., Li, A., Song, Y., Xu, N., & He, W. (2022). Traffic sign recognition based on the YOLOv3 algorithm. *Sensors*, *22*(23), 9345. https://doi.org/10.3390/s22239345

Henry, C., Azimi, S. M., & Merkle, N. (2018). Road segmentation in SAR satellite images with deep fully convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, *15*(12), 1867-1871. https://doi.org/10.1109/LGRS.2018.2864342

Huang, S. C., Lin, H. Y., & Chang, C. C. (2017, June). An in-car camera system for traffic sign detection and recognition. In *2017 Joint 17th World Congress of International Fuzzy Systems Association and 9th International Conference on Soft Computing and Intelligent Systems (IFSA-SCIS)* (pp. 1-6). IEEE. https://doi.org/10.1109/IFSA-SCIS.2017.8023239

Huang, Z., Yu, Y., Gu, J., & Liu, H. (2016). An efficient method for traffic sign recognition based on extreme learning machine. *IEEE Transactions on Cybernetics*, *47*(4), 920-933. https://doi.org/ 10.1109/TCYB.2016.2533424

Jurišić, F., Filković, I., & Kalafatić, Z. (2015, November). Multiple-dataset traffic sign classification with OneCNN. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)* (pp. 614-618). IEEE. https://doi.org/10.1109/ACPR.2015.7486576

Kayed, M., Anter, A., & Mohamed, H. (2020, February). Classification of garments from fashion MNIST dataset using CNN LeNet-5 architecture. In *2020 international conference on Innovative Trends in Communication and Computer Engineering (ITCE)* (pp. 238-243). IEEE. https://doi.org/10.1109/ITCE48509.2020.9047776

Khan, M. A., Park, H., & Chae, J. (2023). A Lightweight Convolutional Neural Network (CNN) Architecture for Traffic Sign Recognition in Urban Road Networks. *Electronics*, *12*(8), 1802. https://doi.org/10.3390/electronics12081802

Klambauer, G., Unterthiner, T., Mayr, A., & Hochreiter, S. (2017). Self-normalizing neural networks. *Advances in Neural Information Processing Systems*, *30*.

Kouretas, I., & Paliouras, V. (2019, May). Simplified hardware implementation of the softmax activation function. In *2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST)* (pp. 1-4). IEEE. https://doi.org/10.1109/MOCAST.2019.8741677

Li, W., Li, D., & Zeng, S. (2019, November). Traffic sign recognition with a small convolutional neural network. In *IOP conference series: Materials Science and Engineering* (Vol. *688*, No. 4, p. 044034). IOP Publishing. https://doi.org/10.1088/1757-899X/688/4/044034

Li, X., Xie, Z., Deng, X., Wu, Y., & Pi, Y. (2022). Traffic sign detection based on improved faster R-CNN for autonomous driving. *The Journal of Supercomputing*, 1-21. https://doi.org/10.1007/s11227-021-04230-4

Lim, X. R., Lee, C. P., Lim, K. M., Ong, T. S., Alqahtani, A., & Ali, M. (2023a). Recent Advances in Traffic Sign Recognition: Approaches and Datasets. *Sensors*, *23*(10), 4674. https://doi.org/10.3390/s23104674

Lim, X. R., Lee, C. P., Lim, K. M., & Ong, T. S. (2023b). Enhanced Traffic Sign Recognition with Ensemble Learning. *Journal of Sensor and Actuator Networks*, *12*(2), 33. https://doi.org/10.3390/jsan12020033

Maletzky, A., Hofer, N., Thumfart, S., Bruckmüller, K., & Kasper, J. (2023). Traffic Sign Detection and Classification on the Austrian Highway Traffic Sign Data Set. *Data*, *8*(1), 16. https://doi.org/10.3390/data8010016

Mathias, M., Timofte, R., Benenson, R., & Van Gool, L. (2013, August). Traffic sign recognition How far are we from the solution?. In *The 2013 International joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE. https://doi.org/10.1109/IJCNN.2013.6707049

Mehta, S., Paunwala, C., & Vaidya, B. (2019, May). CNN based traffic sign classification using Adam optimizer. In *2019 International Conference on Intelligent Computing and Control Systems (ICCS)* (pp. 1293-1298). IEEE. https://doi.org/10.1109/ICCS45141.2019.9065537

Møgelmose, A., Liu, D., & Trivedi, M. M. (2015). Detection of US traffic signs. *IEEE Transactions on Intelligent Transportation Systems*, *16*(6), 3116-3125. https://doi.org/10.1109/TITS.2015.2433019

Paoletti, M. E., Plaza, J., & Plaza, A. (2018, July). An investigation on self-normalized deep neural networks for hyperspectral image classification. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium* (pp. 3607-3610). IEEE. https://doi.org/10.1109/IGARSS.2018.8517449

Postovan, A., & Eraşcu, M. (2023). Architecturing binarized neural networks for traffic sign recognition. *arXiv preprint arXiv:2303.15005*. https://doi.org/10.48550/arXiv.2303.15005

Saouli, A., Aroussi, M., & Fakhri, Y. (2018). Traffic Sign Recognition based on Multi-Feature Fusion and ELM Classifier. *Procedia Computer Science*, *127*, 146-153. https://doi.org/10.1016/j.procs.2018.01.109

Sharma, V., Dhiman, P., & Rout, R. K. (2023). Improved traffic sign recognition algorithm based on YOLOv4-tiny. *Journal of Visual Communication and Image Representation*, *91*, 103774. https://doi.org/10.1016/j.jvcir.2023.103774

Shen, J., & Shafiq, M. O. (2018, December). Deep learning convolutional neural networks with dropout-a parallel approach. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)* (pp. 572-577). IEEE. https://doi.org/10.1109/ICMLA.2018.00092

SivaSai, J. G., Srinivasu, P. N., Sindhuri, M. N., Rohitha, K., & Deepika, S. (2020). An automated segmentation of brain MR image through fuzzy recurrent neural network. In *Bio-inspired Neurocomputing* (pp. 163-179). Singapore: Springer Singapore. https://doi.org/10.1007/978-981-15-5495-7_9

Smagghe, P., Buessler, J. L., & Urban, J. P. (2013, April). Novelty detection in image recognition using IRF Neural Networks properties. In *ESANN*. https://www.esann.org/sites/default/files/proceedings/legacy/es2013-60.pdf

So, H. Y., & Kim, E. M. (2022). Deep-learning-based Automatic Detection and Classification of Traffic Signs Using Images Collected by Mobile Mapping Systems. *Sensors & Materials*, *34*. https://sensors.myu-roup.co.jp/sm_pdf/SM3144.pdf

Song, W., & Suandi, S. A. (2023). Tsr-yolo: A chinese traffic sign recognition algorithm for intelligent vehicles in complex scenes. *Sensors*, *23*(2), 749. https://doi.org/10.3390/s23020749

Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C. (2011, July). The German traffic sign recognition benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks* (pp. 1453-1460). IEEE. https://doi.org/10.1109/IJCNN.2011.6033395

Temel, D., Kwon, G., Prabhushankar, M., & AlRegib, G. (2017). CURE-TSR: Challenging unreal and real environments for traffic sign recognition. *arXiv preprint arXiv:1712.02463*. https://doi.org/10.48550/arXiv.1712.02463

Tong, R., Jiang, Q., Zou, Z., Hu, T., & Li, T. (2023). Embedded system vehicle based on multi-sensor fusion. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3277547

Vashisht, M., & Kumar, B. (2022). Traffic Sign Recognition Approach Using Artificial Neural Network and Chi-Squared Feature Selection. In *Next Generation of Internet of Things: Proceedings of ICNGIoT 2022* (pp. 519-527). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-19-1412-6_45

Xing, J., Nguyen, M., & Qi Yan, W. (2022a). The Improved Framework for Traffic Sign Recognition Using Guided Image Filtering. *SN Computer Science*, *3*(6), 461. https://doi.org/10.1007/s42979-022-01355-y

Xing, J., Luo, Z., Nguyen, M., & Yan, W. Q. (2022b, November). Traffic sign recognition from digital images by using deep learning. In *Pacific-Rim Symposium on Image and Video Technology* (pp. 37-49). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-26431-3_4

Yan, Y., Deng, C., Ma, J., Wang, Y., & Li, Y. (2023). A traffic sign recognition method under complex illumination conditions. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3266825

Yu, N., Jiao, P., & Zheng, Y. (2015, May). Handwritten digits recognition base on improved LeNet5. In *The 27th Chinese Control and Decision Conference (2015 CCDC)* (pp. 4871-4875). IEEE. https://doi.org/10.1109/CCDC.2015.7162796

Zaibi, A., Ladgham, A., & Sakly, A. (2021). A lightweight model for traffic sign classification based on enhanced LeNet-5 network. *Journal of Sensors*, *2021*, 1-13. https://doi.org/10.1155/2021/8870529

Zang, D., Wei, Z., Bao, M., Cheng, J., Zhang, D., Tang, K., & Li, X. (2018). Deep learning–based traffic sign recognition for unmanned autonomous vehicles. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, *232*(5), 497-505. https://doi.org/10.1177/0959651818758865

Zhang, J., Zou, X., Kuang, L. D., Wang, J., Sherratt, R. S., & Yu, X. (2022). CCTSDB 2021: A more comprehensive traffic sign detection benchmark. *Human-centric Computing and Information Sciences*, *12*. https://doi.org/10.22967/HCIS.2022.12.023

Zheng, Y., & Jiang, W. (2022). Evaluation of vision transformers for traffic sign classification. *Wireless Communications and Mobile Computing*, *2022*. https://doi.org/10.1155/2022/3041117

Zhu, Y., & Yan, W. Q. (2022). Traffic sign recognition based on deep learning. *Multimedia Tools and Applications*, *81*(13), 17779-17791. https://doi.org/10.1007/s11042-022-12163-0

Zhou, Y., Wang, X., Zhang, M., Zhu, J., Zheng, R., & Wu, Q. (2019). MPCE: A maximum probability based cross entropy loss function for neural network classification. *IEEE Access*, *7*, 146331-146341. https://doi.org/10.1109/ACCESS.2019.2946264

Zou, Q., Xiao, L., Xu, G., Wang, X., & Mu, N. (2022, December). Traffic Sign Image Segmentation Algorithm Based on Improved Spatio-Temporal Map Convolution. In *International Conference on Green, Pervasive, and Cloud Computing* (pp. 81-98). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-26118-3_6