BILINGUAL PUBLISHING CO.
Pioneer of Global Academics Since 1984

Artificial Intelligence Advances
https://ojs.bilpublishing.com/index.php/aia

ARTICLE

To Perform Road Signs Recognition for Autonomous Vehicles Using Cascaded Deep Learning Pipeline

Riadh Ayachi[1] Yahia ElFahem Said[1,2*] Mohamed Atri[1]

1. Laboratory of Electronics and Microelectronics (EμE), Faculty of Sciences of Monastir, University of Monastir, Tunisia
2. Electrical Engineering Department, College of Engineering, Northern Border University, Arar, Saudi Arabia

ARTICLE INFO

Article history
Received: 26 February 2019
Accepted: 6 April 2019
Published Online: 30 April 2019

Keywords:
Traffic signs classification
Autonomous vehicles
Artificial intelligence
Deep learning
Convolutional Neural Networks CNN
Image understanding

ABSTRACT

Autonomous vehicle is a vehicle that can guide itself without human conduction. It is capable of sensing its environment and moving with little or no human input. This kind of vehicle has become a concrete reality and may pave the way for future systems where computers take over the art of driving. Advanced artificial intelligence control systems interpret sensory information to identify appropriate navigation paths, as well as obstacles and relevant road signs. In this paper, we introduce an intelligent road signs classifier to help autonomous vehicles to recognize and understand road signs. The road signs classifier based on an artificial intelligence technique. In particular, a deep learning model is used, Convolutional Neural Networks (CNN). CNN is a widely used Deep Learning model to solve pattern recognition problems like image classification and object detection. CNN has successfully used to solve computer vision problems because of its methodology in processing images that are similar to the human brain decision making. The evaluation of the proposed pipeline was trained and tested using two different datasets. The proposed CNNs achieved high performance in road sign classification with a validation accuracy of 99.8% and a testing accuracy of 99.6%. The proposed method can be easily implemented for real time application.

1. Introduction

In the recent years, we notice that the number of accidents increases with a huge way. According to the American safety council [13] more than 40000 dies because of cars accidents. The main cause of accident was non-respect of the road rules and speed limits. Automated technologies have been developed and reaches a significant result. Autonomous vehicles are proposed as a solution to make roads safer by taking the control. An autonomous vehicle based on artificial intelligence will not make error in judging situation like human does. Traffic signs classifier is the feature key for developing autonomous vehicles. It provides a global overview about the road rules to control the vehicle and the way how it reacts according to given situation.

Generally, an autonomous vehicle is composed from a big number of sensors and cameras. The visual information provided by the cameras can be used to recognize the road signs. To process visual information, a well-known Deep Learning model, Convolutional Neural Networks (CNN) [1], are proposed. They are widely used in image

**Corresponding Author:*
Yahia ElFahem Said,
Electrical Engineering Department, College of Engineering, Northern Border University, Arar, Saudi Arabia;
Email: said.yahia1@gmail.com

 DOI: https://doi.org/10.30564/aia.v1i1.569

processing tasks such as object recognition, image classification[2] and object localization[3]. CNNs are successfully used to solve computer vision tasks[4] because of their power in visual context processing that mimic the biological system were every neuron in the network is applied in a restricted region of the receptive field[5]. Then all the neurons of the network overlapped to cover the entire receptive field. So, features from all the receptive field are shared everywhere in the network with less effort. The major advantage of the Convolutional Neural networks is the ability to learn directly from the image[6], unlike other classification algorithm that need a hand-crafted feature to learn from.

For human, recognizing and classifying a traffic sign is an easy task and the classification will be totally correct but for an artificial system, it is a hard task that needs a lot of computation effort. In many countries the shape and the color of the same road sign is different. Figure 1 illustrates an example of the stop sign in different countries. In addition, the road sign can look different because of the environment factors like rain, sun and dust. Though the mentioned challenges need to be processed successfully to make a robust road sign classifier with the minimum of error.

Figure 1. Stop Sign in Different Countries

In this paper, we propose a pipeline based on data preprocessing algorithm and deep learning model to recognize and classify traffic signs. The data preprocessing pipeline is composed by five stages. First, data loading and augmentation are performed. Then, all the images are resized and shuffled. All the images are then transformed to gray scale channel. After that, we apply a local histogram equalization[8, 9, 10]. Finally, we normalize the images to feed them to the proposed convolutional neural network.

As CNN model, we propose two different networks. The first one is 14 layers subset from the VGGNet model[12], which is invented by VGG (Visual Geometry Group) from University of Oxford, and was the 1st runner-up of the classification task in the ILSVRC2014 challenge[32] and the winner of the localization task. The second one is the Deep Residual Network ResNet[11]. It was arguably the most groundbreaking work in the computer vision/deep learning community in the last few years. ResNet makes it possible to train up to hundreds or even thousands of layers and still achieves compelling performance.

By testing the proposed networks, we achieve high performance in both validation and tests. The best performance was achieved using the 34 layers ResNet architecture with a validation accuracy of 99.8% and a testing accuracy of 99.6%. Also achieving an inference speed of more than 40 frames per second, the pipeline can be implemented for real time applications.

The remainder of the paper is organized as follows. Related works on traffic signs classification are presented in Section 2. Section 3 describes the proposed pipeline to recognize and classify road signs. In Section 4, experiments and results are detailed. Finally, Section 5 concludes the paper.

2. Related Works

The need for a robust traffic sign classifier became an important benchmark that must be solved. Many research works were presented in the literature[14,15,36]. Ohgushi et al.[16] introduced a traffic signs classifier based on color information and Bags of Features (BoF) as a features extractor and a support vector machine (SVM) as a classifier. The proposed mothed struggle in recognizing the traffic signs in real condition especially when the sign is intensively illuminated or partially occluded.

Some research investigated the detection of the traffic sign without performing the classification process[17,18]. Wu et al.[17] proposed a method to detect only round traffic signs in the Chinese roads. In other side, researchers focus on detecting and recognizing the traffic sign[19]. The proposed method only detects round signs and cannot detect other signs shapes.

A three steps method to detect and recognize traffic signs was proposed by Wali et al.[20]. The first step was data preprocessing. The second was detecting the existence of the sign and the third was classifying it. For the detection process, they apply the color segmentation with shape matching and for the classification process they use SVM as a classifier. The proposed method achieves 95.71% of accuracy. Lai et al.[21] introduced a traffic signs recognition method using smart phone. They used color detection to perform color space segmentation and shape recognition method using template matching by calculating the similarity. Also, an optical character recognition

DOI: https://doi.org/10.30564/aia.v1i1.569

(OCR) was implemented inside the shape border to decide on the sign class. The proposed method was very limited on red traffic signs only. Gecer et al.[38] propose to use color-blob-based COSFIRE to recognize traffic signs. The proposed method was based on a Combination of Shifted Filter Responses with compute the response of different filters is different regions in each channel of the color space (ie. RGB). The proposed method achieves 98.94% as accuracy on the GTSRB dataset.

Virupakshappa et al.[22] used a machine learning method by combining the bag-of-visual-words technique with Speeded up Robust Features (SURF) for features extraction then feed the features to an SVM classifier to recognize the traffic signs. The proposed method achieves an accuracy of 95.2%. A system based on a BoW descriptor enhanced using spatial histogram was used by Shams et al.[23] to improve the classification process based on an SVM classifier.

Lin et al.[24] introduced a two-stage fuzzy inference model to detect traffic signs in video frame the they apply a two-stage fuzzy inference model to classifier the signs. The method provides high performance only on prohibitory and warning signs. In[25], Yin et al. presented a revolutionary technique for real time processing based on Hough transformation to localize the sign in the image the use the rotation invariant binary pattern (RIBP) descriptor to extract features. As a classification method they use artificial neural networks.

A cascade Convolutional Neural Network model was introduced by Rachmadi et al.[26] to perform the traffic signs classification process of the Japanese road signs. The proposed method achieves a performance of 97.94% and can be implemented for real time processing with a speed less than 20 ms per image. The mothed of Sermanet et al.[39] was based on a multi-scale convolutional neural network. This method introduces a new connection way by skipping layers and the use of pooling layers with down sampling ratios for connection that skip layers different than those that do not skip layers. The proposed method improves its efficiency by reaching 99.1% accuracy. Cireçsan et al[37] used a combination of CNNs and train them in parallel using differently preprocessed data. It uses an arbitrary number of CNNs each is combined from seven layers, input layer, two convolution layers, two max pooling layers and two fully connected layers. The prediction is provided by averaging the output of all the CNNs. The proposed technique further boosts the classification accuracy to 99.4%. The use of convolutional neural networks has led to enhance the classification accuracy compared with the machine learning techniques.

In the recent years, several vehicle manufactories develop new techniques to perform traffic signs classification. As an example, BMW announced the integration of a traffic sign classifier in the BMW 5 series. Moreover, other vehicle manufactories were trying to implement those technologies[27]. Volkswagen implement a traffic sign classifier in the Audi A8[28]. All the existing researches on the traffic signs classification proved the important of this technology for autonomous cars.

3. Proposed Method

As mentioned above many traffic signs classification techniques are proposed. Our method focusses on the data preprocessing technique to enhance the images quality and to reduce the number of features learned by the convolutional Neural Network so we ensure the real time implementation. As shown in figure 2, the preprocessing technique contain five phases: data loading and augmentation, images resizing and shuffling[29], gray scaling, local histogram equalization[30] and data normalization.

As a first phase, we load the data and we generate new examples using a data augmentation technique. The data augmentation process is applied to maximize the amount of the training data. Also, the data augmentation was used in the tests by generating more points of view of the tested image to ensure better prediction.

In the second phase, we resize all the images to height*width*3 where 3 denotes the 3 channels color space. Then the images are shuffled to avoid obtaining minibatches of highly correlated examples. So, the training algorithm will choose a different minibatch each time it iterates. In third phase, we perform gray scaling to reduce the number of channels of the image so the images are scaled to height *width*1. As result of the gray scaling technique the number of learned filters was reduced in the convolutional neural network. Also, the training and inference time can be reduced. In the fourth phase, we apply local histogram equalization[31] to enhance the images contrast by separating the most frequent intensity values. Usually, this increases the global contrast of the images and allows to the areas of lower local contrast to gain a higher contrast. The fifth phase consists of data normalization which is a simple process applied to get the same data scale of all the examples ensuring an equal representation of all the features. The preprocessing pipeline is an important stage to enhance the data injected to the network in both training and testing process.

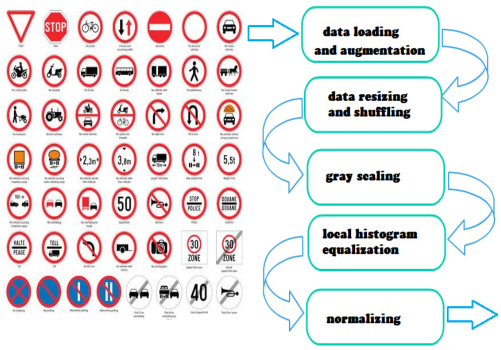

Figure 2. Data Preprocessing

The second part of our method is the Convolutional Neural Network (CNN). Generally, a convolutional neural network is feedforward neural network used to solve computer vision tasks. Usually, a CNN contains six types of layers: input layer, convolution layers, nonlinear layers, pooling layers, fully connected layers and an output layer. Figure 3 illustrates a CNN architecture.

The complete proposed pipeline is composed from a data preprocessing stage and a convolutional neural network for traffic signs classification. The proposed pipeline can be summarized by the pseudo code presented in algorithm 1.

Algorithm 1: proposed pipeline for traffic signs classification
Train input: images, labels Test input: images Output: images classes Mode: choose the mode (training or testing) Batch size: choose a batch size (number of images per batch) Image size: choose the images size Number of batches: choose a number of batches If mode: training For batch in range (number of batches): Load the data (images and labels) Apply data augmentation Resize the images Shuffle the images Apply local histogram equilibration Normalize the images Fit the images into the convolutional neural network Initialize the CNN parameters (load weights from pretrained model) Compute the mapping function Generate the output Repeat Compute the loss function (difference between output class and input label) Optimize CNN parameters (apply backpropagation algorithm) Until output class input label Chose next batch Else (mode: testing) Load the data (images) Apply data augmentation Resize the images Apply local histogram equilibration Normalize the images Fit the images into the convolutional neural network Load parameters from trained model Compute the mapping function Generate the output

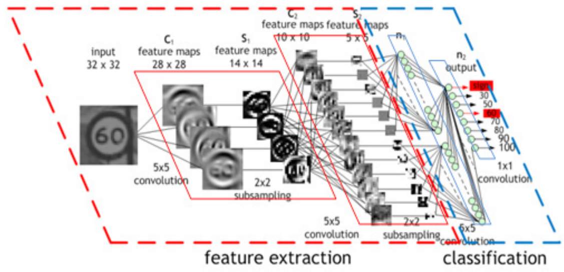

Figure 3. Convolutional Neural Network Architecture

The first CNN to use is VGGNet[12]. VGGNet have two main architectures: the VGG16 which is a 16 layers CNN and the VGG19 which is a 19 layers CNN. The VGGNet architectures are presented in figure 4. VGGNet achieves a top 5 error in the ILSVRC2014 classification challenge [32] of 7.32%. In our work we will just use 14 layers from the VGGNet by saving the first 10 layers and the 4 last layers. Also, in the third block we will use just 2 convolutional layers and a pooling layer.

Figure 4. VGGNet Architecture

The second CNN that we will explore is ResNet[11] which presents a revolutionary architecture to accelerate the convergence of the very deep neural networks (more than 20 layers) by implementing residual blocks instead of classic plain blocks used in VGGNet. An illustration of the residual block is shown in figure 5. ResNet wins the ILSVRC2015 classification contest [32] achieving the top-5 validation error of 3.57%[11]. To perform traffic signs classification, we choose ResNet 34 architecture. Figure 5 presents the structure of ResNet 34 which is a 34 layers CNN with residual blocks. A residual block is an accumulation of the input and the output of the block.

VGGNet and ResNet are trained to classify natural images according to the ImageNet [32] with 1000 classes. To make it perfect for the traffic signs classifier, the transfer learning technique was applied by replacing the output layers of those architectures by another layer contains the classes of the traffic signs. The transfer learning technique is well known technique in deep learning which helps to use existing architecture to solve new tasks by freezing some layers and fine tuning the other layers or retrain them from scratch. The transfer learning is used to speed up the training process and to improve the performance

  DOI: https://doi.org/10.30564/aia.v1i1.569

of the used deep learning architecture. Using the transfer learning technique allows to use the pre-trained weights as a starting point to optimize the existing architecture for the news task.

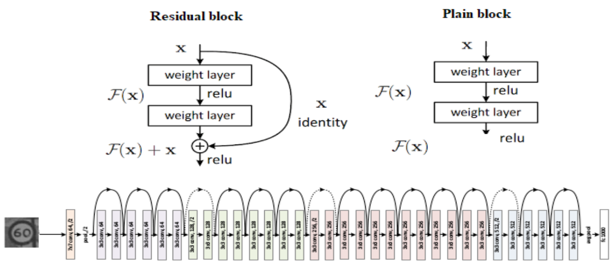

Figure 5. ResNet34 Structure

Another advantage of the transfer learning is possibility to use a small amount of data to train the deep learning model and achieve high performance.

4. Experiments and Results

In this work two datasets were used to train and evaluate the networks. The first dataset is the German traffic signs dataset GTSRB[34], which is a large multi-class dataset for traffic signs classification benchmark. In this dataset there is a training directory and a testing directory, each contain 43 traffic signs classes providing more than 50000 total images of traffic signs in real conditions. Figure 6 represents the classes of the German traffic signs dataset. The second data set is the Belgium traffic signs dataset BTSC[35]. This dataset provides a training and teasing data separately. The training and the testing data contain 62 traffic signs classes and more than 4000 images of real traffic signs in the Belgium roads.

Figure 6: the German Traffic Signs Dataset Classes

In all our experiments, all the networks are developed using the TensorFlow deep neural network framework. The training is performed using a desktop with Intel i7 processor and an Nvidia GTX960 GPGPU.

To achieve good performance, we use a variant of configuration by manipulating the images sizes, the batch size, the dropout probability and choosing the learning algorithm (optimizer). We start by resizing the images to 32*32. Also, we start by using a large batch size (1024), the dropout probability of 0.25 and as learning algorithm we use stochastic gradient descent and we perform training the network.

The final used images resizing value was determined after testing many different values such as 32*32, 64*64, 96*96 and 128*128, and after several tests, we end up by the best configuration which is resizing the images to 96*96, using a minibatch of 256, a dropout probability of 0.5 and the Adam optimizer. The Adam optimizer is an extension of the stochastic gradient descent optimizer which guarantee a better and faster converge. In addition, it does not need a learning rate, it will generate its own learning rate and optimize it until finding the best value.

Figure 7. the Belgium Dataset Classes

In the data pre-processing pipeline, the data was prepared for training and testing the model. First, loading the data and applying the data augmentation technique. Figure 8 shows an example of the generated data using the proposed data augmentation technique. Second, resizing the data and shuffle it to generate mixed mini batches. Then, images were transformed to the gray scale space color. Figure 9 illustrates an example of the gray scaled images.

 DOI: https://doi.org/10.30564/aia.v1i1.569

Figure 8. Data Augmentation

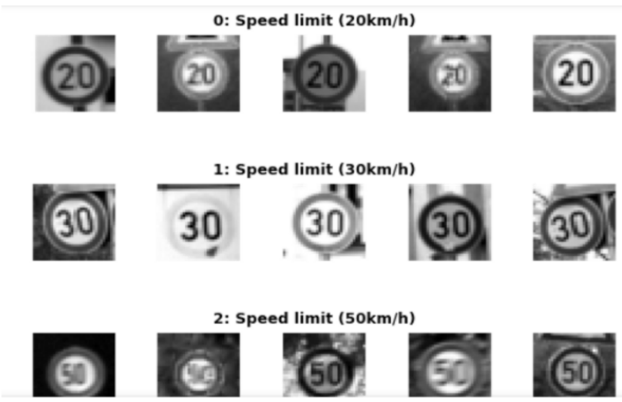

Figure 9. Gray Scaling

The local histogram equalization was then applied to equilibrate the images contrasts. Figure 10 present images after applying the local histogram equalization. Finally, normalizing the data and feed it to the convolutional neural network. An example of the normalized data is presented in figure 11.

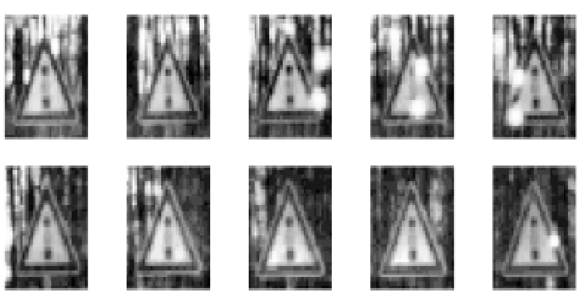

Figure 10. Local Histogram Equalization

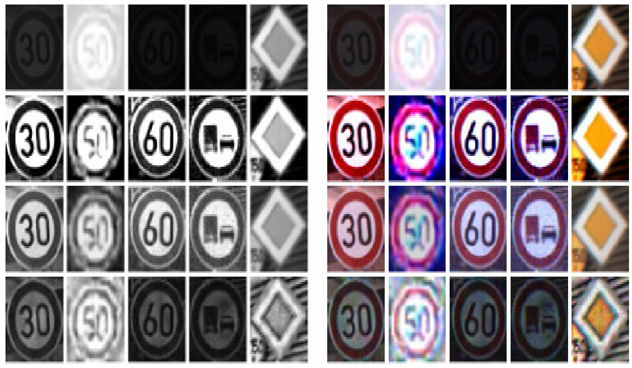

Figure 11. Normalized Gray Images and the Original Color Images

In the training process, the data was injected to the CNN architectures and the parameters are optimized. In the ResNet 34, the first convolution layer was used to perform feature extraction and down sampling in the same time by using 7*7 kernels to incorporate features with larger receptive field and a stride of 2. Figure 12 presents the output feature maps of the first ResNet 34 convolution layer. The residual blocks are used for features extraction using 2 convolutional layers with 3*3 kernels and zero padding was applied. The input and the output of each residual block are accumulated to control parameters number explosion. Figure 13 presents the output feature maps of the first ResNet 34 residual block.

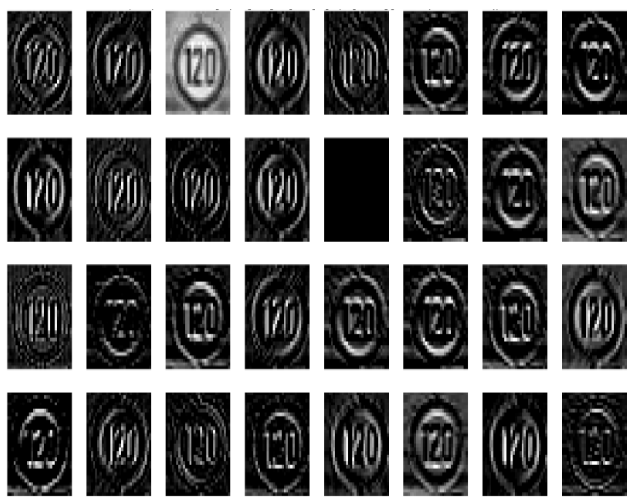

Figure 12. Features Maps of the First ResNet34 Convolution Layer

DOI: https://doi.org/10.30564/aia.v1i1.569

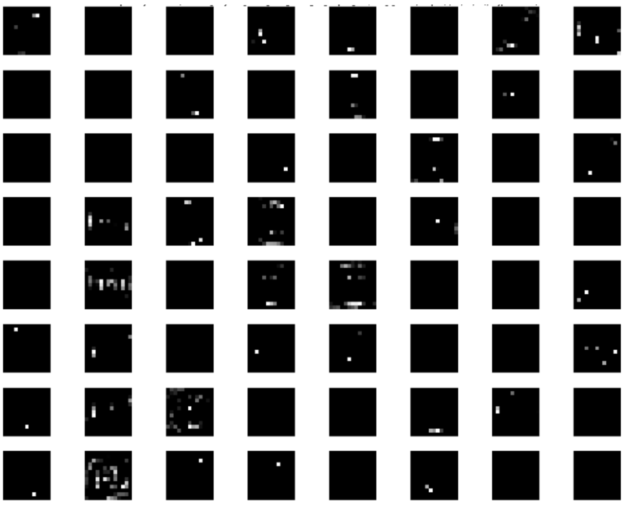

Figure 13. Features Maps of the First ResNet34 Residual Block

A way to visualize the CNN performance is by representing the corresponding confusion matrix. The confusion matrix shows the ways in which the classification CNN model is confused when it makes predictions. Figure 14 shows the confusion matrix of the ResNet on the GTSRB dataset.

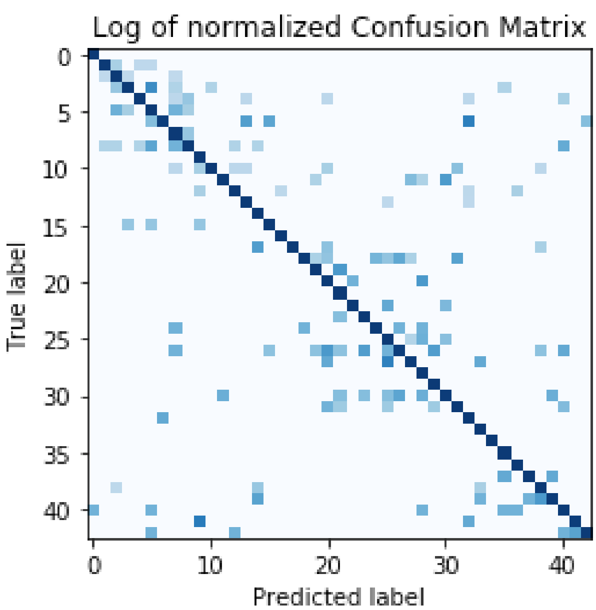

Figure 14. Confusion Matrix of ResNet34

Table 1. Performance of the Proposed Architectures in Term of Accuracy in Both Datasets

	Accuracy (%)	
Dataset	GTSRB	BTSC
VGG (12 layers)	99.3	98.3
ResNet 34	99.6	98.8

Table 1 summarize the obtained accuracy on the testing data of the trained models on the GTSRB and the BTSC datasets. As shown in table 1 the best performance is obtained on the GTSRB dataset using the ResNet 34 architecture and this proves the importance of the residual block to enhance the network performance without any explosion in the complexity when using very deep convolutional neural network. The results obtained on the BTSC data set are lower because of the lack of data. The dataset contains only 4965 images divided on training data and testing data. The reported data on the GTSRB dataset proved that the proposed traffic sign classifier outperformed the human accuracy which is 98.32%. The most of the false negative examples are caused by totally or partially damaged images after performing the data pre-processing. Figure 15 illustrate an example of the damaged images.

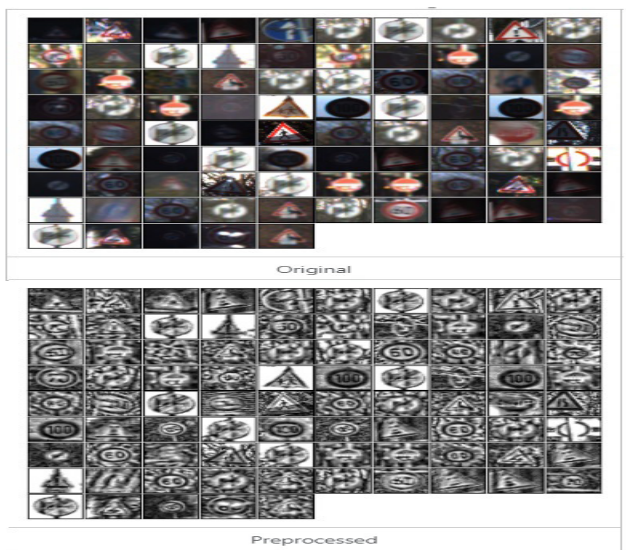

Figure 15. Damaged Images after Preprocessing

Table 2. Inference Speed of Each Architecture

Architecture	frames/second
VGG (12 layers)	57
ResNet 34	43

Table 2 summarize the number of images processed per second by each architecture. For real time implementation, we need an equilibration between accuracy and speed. Our best proposed CNN achieve an accuracy of 99.621% which is an acceptable value in comparison of human accuracy and outperform the state-of-the-art models in the traffic signs classification task.

Table 3 presents a comparison between our architectures and other proposed architectures and methods tested on the GTSRB dataset.

 DOI: https://doi.org/10.30564/aia.v1i1.569

Table 3. Accuracy Comparison

Architecture	Accuracy (%)
Committee of CNN [37]	99.4
Color-blob-based COSFIRE filters [38]	98.9
Sermanet [39]	99.1
Proposed VGG (12 layers)	99.3
Proposed ResNet 34	99.6

As reported in table 3, our proposed ResNet 34 architecture outperform state of the art methods in traffic signs classification. Also, our architecture can be easily implemented for real time applications. A real time application needs at least a 25 frames per second and as reported in table 2, the lowest architecture processes 43 frames per second. In other hand, all the proposed architecture outperforms human accuracy in the traffic signs classification benchmark.

To make it useful for real word application and human interpretable, we implement the ResNet 34 architecture in traffic signs classification application where we label the images with human understandable labels. In both training and tests label were encoded as integers. As example the labels were encoded from 0 to 42 range in the GTSRB dataset. The testing images was collected from the web and does not belong to the datasets. The top 5 probabilities of the softmax layer were visualized. Figure 16 presents an example of the top 5 probabilities of the softmax layer and their corresponding input images. The classifier achieves a good performance when applied to the new images and proves the generalization power.

5. Conclusion

Traffic signs classification was and still an important application for autonomous cars. Cars need real time and embedded solutions that is why we need to provide a balance between speed and accuracy. In this paper, we propose an artificial intelligence technique based on deep learning model, Convolutional Neural Network to perform the traffic signs classification benchmark. The reported results prove that the proposed solutions can be effectively implemented for real time applications and provide an acceptable accuracy outperforming human performance. The proposed architectures can be more optimized for embedded implementation.

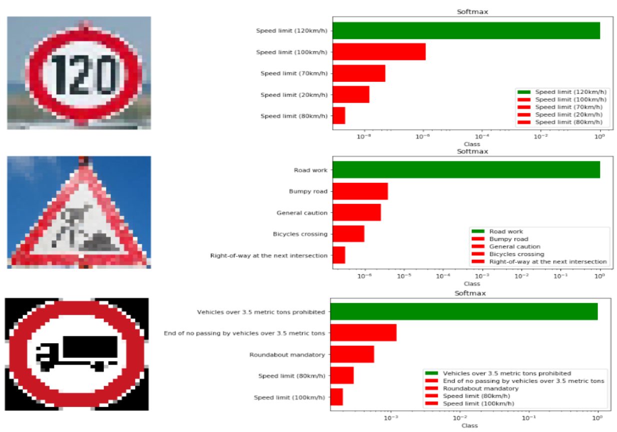

Figure 16. ResNet34 Softmax Probabilitie

Conflicts of Interest:

The authors declare no conflict of interest.

References

[1] O'Shea, K., & Nash, R. An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458, 2015.

[2] Ciresan, D. C., Meier, U., Masci, J., Maria Gambardella, L., & Schmidhuber, J. Flexible, high performance convolutional neural networks for image classification. In IJCAI Proceedings-International Joint Conference on Artificial Intelligence, 2011,22(1): 1237.

[3] Tompson, J., Goroshin, R., Jain, A., et al. Efficient object localization using convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 648-656.

[4] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[5] Simard, P. Y., Steinkraus, D., & Platt, J. C. Best practices for convolutional neural networks applied to visual document analysis. In null (p. 958). IEEE, 2003.

[6] LeCun, Y., Jackel, L. D., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., ... & Vapnik, V. Learning algorithms for classification: A comparison on handwritten digit recognition. Neural networks: the statistical mechanics perspective, 1995, 261: 276.

[7] Zhu, H., Chan, F. H., & Lam, F. K. Image contrast enhancement by constrained local histogram equalization. Computer vision and image understanding, 1999, 73(2): 281-290.

[8] Kim, J. Y., Kim, L. S., & Hwang, S. H. An advanced contrast enhancement using partially overlapped sub-block histogram equalization. IEEE transactions on circuits and systems for video technology, 2001,

  DOI: https://doi.org/10.30564/aia.v1i1.569

11(4): 475-484.
[9] Stark, J. A. Adaptive image contrast enhancement using generalizations of histogram equalization. IEEE Transactions on image processing, 2000, 9(5): 889-896.
[10] He, K., Zhang, X., Ren, S., & Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 770-778.
[11] Sercu, T., Puhrsch, C., Kingsbury, B., & LeCun, Y. Very deep multilingual convolutional neural networks for LVCSR. In Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, IEEE, 2016: 4955-4959.
[12] U.S. vehicle deaths topped 40,000 in 2017, National Safety Council estimates. https://www.usatoday.com/story/money/cars/2018/02/15/national-safety-council-traffic-deaths/340012002
[13] Vitabile, S.; Pollaccia, G.; Pilato, G.; Sorbello, F. Road signs recognition using a dynamic pixel aggregation technique in the HSV color space. In Proceedings of the 11th International Conference on Image Analysis and Processing, Palermo, Italy, 2001, 26–28: pp. 572–577.
[14] Zeng, Y.; Lan, J.; Ran, B.; Wang, Q.; Gao, J. Restoration of motion-blurred image based on border deformation detection: A traffic sign restoration model. PLoS ONE, 10, e0120885. 2015.
[15] Ohgushi, K.; Hamada, N. Traffic sign recognition by bags of features. In Proceedings of the TENCON 2009—2009 IEEE Region 10 Conference, Singapore, 2009, 23–26: 1–6.
[16] Wu, J.; Si, M.; Tan, F.; Gu, C. Real-time automatic road sign detection. In Proceedings of the Fifth International Conference on Image and Graphics (ICIG ’09), Xi’an, China, 2009: 540– 544.
[17] Belaroussi, R.; Foucher, P.; Tarel, J.P.; Soheilian, B.; Charbonnier, P.; Paparoditis, N. Road sign detection in images: A case study. In Proceedings of the 20th International Conference on Pattern Recognition (ICPR), Istanbul, Turkey; 2010: 484–488.
[18] Shoba, E.; Suruliandi, A. Performance analysis on road sign detection, extraction and recognition techniques. In Proceedings of the 2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT), Nagercoil, India, 2013: 1167–1173.
[19] Wali, S.B.; Hannan, M.A.; Hussain, A.; Samad, S.A. An automatic traffic sign detection and recognition system based on colour segmentation, shape matching, and svm. Math. Probl. Eng, 2015.
[20] Lai, C.H.; Yu, C.C. An efficient real-time traffic sign recognition system for intelligent vehicles with smart phones. In Proceedings of the 2010 International Conference on Technologies and Applications of Artificial Intelligence, Hsinchu, Taiwan, 2010: 195–202.
[21] Virupakshappa, K.; Han, Y.; Oruklu, E. Traffic sign recognition based on prevailing bag of visual words representation on feature descriptors. In Proceedings of the 2015 IEEE International Conference on Electro/Information Technology (EIT), Dekalb, IL, USA, 2015: 489–493.
[22] Shams, M.M.; Kaveh, H.; Safabakhsh, R. Traffic sign recognition using an extended bag-of-features model with spatial histogram. In Proceedings of the 2015 Signal Processing and Intelligent Systems Conference (SPIS), Tehran, Iran, 2015: 189–193.
[23] Lin, C.-C.; Wang, M.-S. Road sign recognition with fuzzy adaptive pre-processing models. Sensors, 6415. 2012.
[24] Yin, S.; Ouyang, P.; Liu, L.; Guo, Y.; Wei, S. Fast traffic sign recognition with a rotation invariant binary pattern-based feature. Sensors, 2015, 2161–2180.
[25] Rachmadi½, R. F., Komokata½, Y., Íchimura½, K., & Koutaki½, G. (2017). Road sign classification system using cascade convolutional neural network, 2017.
[26] Continental. Traffic Sign Recognition. Available online: 2017. http://www.contionline.com/generator/www/de/en/continental/automotive/general/chassis/safety/hidden/verkehrszei chenerkennung_en.html
[27] Choi, Y.; Han, S.I.; Kong, S.-H.; Ko, H. Driver status monitoring systems for smart vehicles using physiological sensors: A safety enhancement system from automobile manufacturers. IEEE Signal Process. 2016: 22–34.
[28] Dean, J., & Ghemawat, S. MapReduce: simplified data processing on large clusters. Communications of the ACM, 2008, 51(1), 107-113.
[29] Kim, J. Y., Kim, L. S., & Hwang, S. H. An advanced contrast enhancement using partially overlapped sub-block histogram equalization. IEEE transactions on circuits and systems for video technology, 2001, 11(4), 475-484.
[30] Abdullah-Al-Wadud, M., Kabir, M. H., Dewan, M. A. A., & Chae, O. A dynamic histogram equalization for image contrast enhancement. IEEE Transactions on Consumer Electronics, 2007, 53(2).
[31] Olga, R., Jia, D., Hao S., Jonathan, K., Sanjeev, S., Sean, M., Zhiheng, H., Andrej, K., Aditya, K., Michael, B., Alexander, C. B., and Li, F. ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015.
[32] Hinton, Geoffrey E, Srivastava, Nitish, Krizhevsky,

Alex, Sutskever, Ilya, and Salakhutdinov, Ruslan R. Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580, 2012.

[33] J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, Neural Networks. 2012, ISSN 0893-6080.
http://www.sciencedirect.com/science/article/pii/S0893608012000457

[34] Radu Timofte, Karel Zimmermann, Luc van Gool, Multi-view traffic sign detection, recognition, and 3D localisation, IEEE Workshop on Applications of Computer Vision, WACV, 2009.

[35] Fredrik, L. and Michael, F., Using Fourier Descriptors and Spatial Models for Traffic Sign Recognition, In Proceedings of the 17th Scandinavian Conference on Image Analysis, SCIA, LNCS 6688, 2011: 238-24.

[36] CireşAn, Dan, et al. "Multi-column deep neural network for traffic sign classification." Neural networks 2012, 32: 333-338.

[37] Gecer, B., Azzopardi, G., & Petkov, N. Color-blob-based COSFIRE filters for object recognition. Image and Vision Computing, 2017, 57: 165-174.

[38] Sermanet, P., & LeCun, Y. Traffic sign recognition with multi-scale convolutional networks. In Neural Networks (IJCNN), the 2011 International Joint Conference on. IEEE, 2011: 2809-2813.

DOI: https://doi.org/10.30564/aia.v1i1.569