**CNN Design for Real-Time Traffic Sign Recognition**

*Nguyễn Công Quý*

*Department of Mechatronics Engineering, Ho Chi Minh City University of Technology and Education (HCMUTE), Ho Chi Minh City, Vietnam;* [*20134022@student.hcmute.edu.vn*](mailto:20134022@student.hcmute.edu.vn)

**Abstract:** These days, convolutional neural networks are used to perform an increasing number of object identification tasks (CNN). The majority of computer vision problems, both old and new, have been improved by convolutional neural networks due to their high recognition rate and quick execution. In this paper, we provide a convolution neural network implementation of the traffic sign recognition algorithm. On my own computer, the full detection and recognition process for traffic signs is carried out in real-time. The experimental findings supported the designed computer vision system's high efficiency.

*Keywords:* TensorFlow; Convolutional Neural Networks; Traffic Sign Recognition; Image Processing; Computer Vision.

# **1. Introduction**

Many automakers were able to integrate computer vision systems into passenger cars because to advancements in the technical capabilities of contemporary mobile CPUs. These systems play a key role in implementing a crucial step toward autonomous driving and considerably enhancing safety. The traffic sign recognition (TSR) challenge is one of the most well-known and extensively debated problems among other computer vision-based jobs. The main issues with such systems, however, are their poor detection accuracy, expensive hardware computational performance requirements, and the inability of some systems to categorize traffic signs from other nations.

The two steps of localization and classification are typically used to solve the problem of traffic sign recognition. Numerous localization techniques exist [1], [2], and [3]. The authors proposed efficient real-time implementations of traffic sign localization and image preprocessing algorithms in papers [4] and [5].

The answer enabled the precise location of a traffic sign in the acquired image using a modified Generalized Hough Transform (GHT) algorithm. As a result, the basic template-matching algorithm was employed during the classification stage. This algorithm demonstrated a final traffic sign recognition accuracy of 98.3% when combined with a precise localization stage. The developed algorithms were trained and tested using the GTSRB [6] and GTSDB [7] datasets Fig.1 displays the images used to test the localization algorithm and train the algorithm to recognize traffic signs.



*Fig. 1. Images from GTSDB and GTSRB*

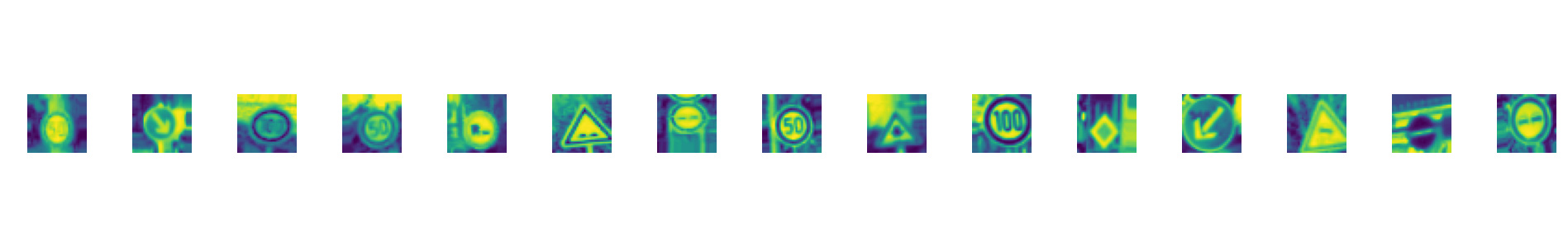
The end-to-end technology demonstrated a significant decrease in efficiency when used in real-life situations, such as using videos from cameras mounted on a windshield, to test the developed technology for detecting and classifying traffic signs. According to studies, this decrease resulted from images of localized traffic signs having too many variations in the lighting, contrast, and rotation angle. Because there are only a few predefined templates, a straightforward classification algorithm like template matching was unable to achieve high-quality recognition. Convolutional neural networks, which have seen such widespread use recently [8], [9], can be used to combine the localization algorithm, which has produced good results, with recognition to improve system performance.

In this paper, we present an updated end-to-end technology for real-time traffic sign detection and recognition. The created system makes use of the vehicle's speed. By doing so, you can foretell not only the existence of the object but also its size and precise coordinates in the adjacent frame. As a result, detection accuracy rises while computational complexity stays the same. Convolutional neural networks are used to implement the classification of locally located objects (CNNs). This paper's description of the convolutional neural network design process is among its key contributions. Real-time processing of the frames in the video sequence is made possible by the personal computer.

# **2. Traffic Sign Localization and Tracking**

The developed technology for traffic signs recognition consists of three steps: image preprocessing, localization and classification.

During image preprocessing, the input image’s color space is transferred to gray by the CV2 library Due to errors in the process of the image acquiring and the presence of small colored objects, some point-like noise occurs in the images after applying a threshold filter.



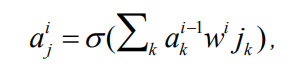
*Fig.2. Images of preprocessing stage*

The algorithms for identifying and following traffic signs are covered in Paper [5]. The time constraints for processing a single frame have been taken into consideration when developing the localization method, which is a modification of the generalized Hough transform. The algorithm performs well with the preprocessed images and produces useful results. The system's performance has increased as a result of tracking using the vehicle's current speed value because it allows for a significant reduction in the search area in subsequent frames. Additionally, the confidence in accurate recognition is significantly increased by the presence of a sign in the sequence of adjacent frames in predicted areas. The final step, classification, verifies that the entire process was carried out successfully.

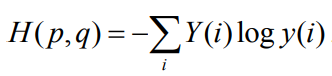
# **3. Traffic Sign Classification**

## **3.1 Convolutional Neural Networks**

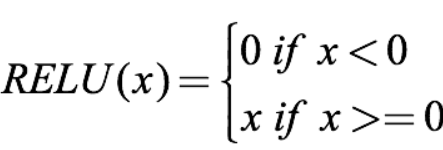
Artificial neural network classification is a very well-liked method for addressing pattern recognition issues. A neural network is a mathematical representation of a biological neural network made of artificial neurons connected to one another. Neurons are typically arranged in layers, and connections are only made between neurons from adjacent layers. The first layer contains the input low-level feature vector, which is then transformed into the high-level feature vector as it moves through the layers. The number of classifying classes is the same as the number of output layer neurons. As a result, the probability vector representing the likelihood that the input vector belongs to the specified class makes up the output vector. The weighted adder is implemented by an artificial neuron, and its output is described as follows [11]:

(1)

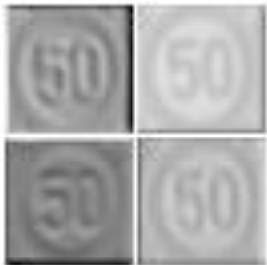
where 𝑎ij is the jth neuron in the ith layer, 𝑤kij stands for weight of a synapse, which connects the jth neuron in the ith layer with the kth neuron in the layer i-1. Widely used in regression, the logistic function is applied as an activation function. It is worth noting that the single artificial neuron performs the logistic regression function. The training process is to minimize the cost function with minimization methods based on the gradient decent also known as backpropagation. In classification problems, the most commonly used cost function is the cross entropy:

(2)

Training networks with large number of layers, also called deep networks, with sigmoid activation is difficult due to vanishing gradient problem. To overcome this problem, the RELU function is used as an activation function [12]:

 (3)

Today, classifying with convolutional neural networks is the state-of-the-art pattern recognition method in computer vision. Unlike traditional neural networks, which work with one-dimensional feature vectors, a convolutional neural network takes a two-dimensional image and consequentially processes it with convolutional layers. Each convolutional layer consists of a set of trainable filters and computes dot productions between these filters and layer input to obtain an activation map. These filters are also known as kernels and allow detection of the same features in different locations. For example, Fig. 3 shows the result of applying convolution to an image with 4 kernels.



*Fig. 3. Input image convolution*

## **3.2 Proposed Implementation**

We employed the deep learning library Keras to resolve the task of recognizing traffic signs. The dataset from GTSRB was used for training and testing [6]. The sixteen most common types of traffic signals can be categorized using the new system.

In order to construct a network architecture, there are several rules. Despite this, the majority of the network architecture design process is heuristic. Layers are chosen so that when they are added, data dimensionality decreases. However, there are no guidelines on specific layer macro settings.

Data volume and network depth should be correlated. Large networks and little data are likely to result in overfitted models. However, a shallow network with a lot of data would not provide adequate accuracy.

Table 1 describes the first developed network architecture. The architecture consists of several convolutional layers, maxPooling2D layer. All convolutional layers have a parameter stride equal to 2. This parameter determines the stride of the convolution sliding window, so layers with a parameter stride greater than 1 also combine the pooling operation.

Table 1. Neural network architecture.

|  |
| --- |
| Layer 1 |
| Convolutional, stride 2, kernel 5x5x60 |
| Convolutional, stride 2, kernel 5x5x60 |
| MaxPooling2D, pool\_size 2x2 |

When training a network proposed in Table 1 architecture, the classification accuracy reached a value of more than 0.9. However, this architecture seems to be exceeded due to a large number of layers. Thus, I decided to reduce the number of convolutional layers, which after several unsuccessful attempts resulted in the architecture, and add a Dropout layer presented in Table 2.

Table 2. Second neural network architecture.

|  |
| --- |
| Layer 2 |
| Convolutional, stride 2, kernel 3x3x30 |
| Convolutional, stride 2, kernel 3x3x30 |
| MaxPooling2D, pool\_size 2x2 |
| Dropout, 0.5 |

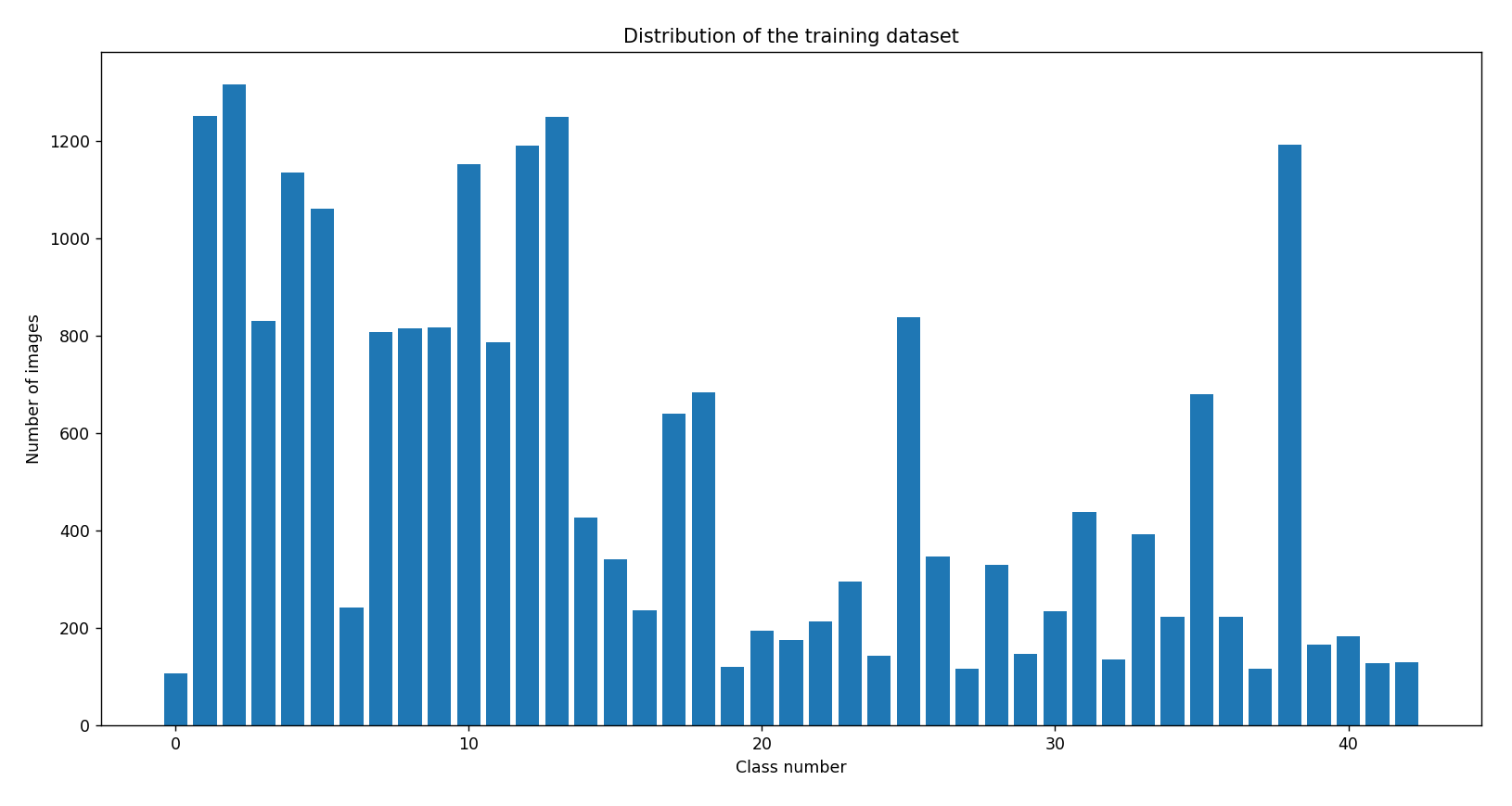
Finally, Table 3 shows the modified architecture of CNN. The reason why I add one more Dropout layer to reduce overfit while training and my model totally has 20 epochs.

Table 3. Final neural network architecture.

|  |
| --- |
| Layer 2 |
| Fully connected -512 |
| Dropout, 0.5 |
| Softmax |

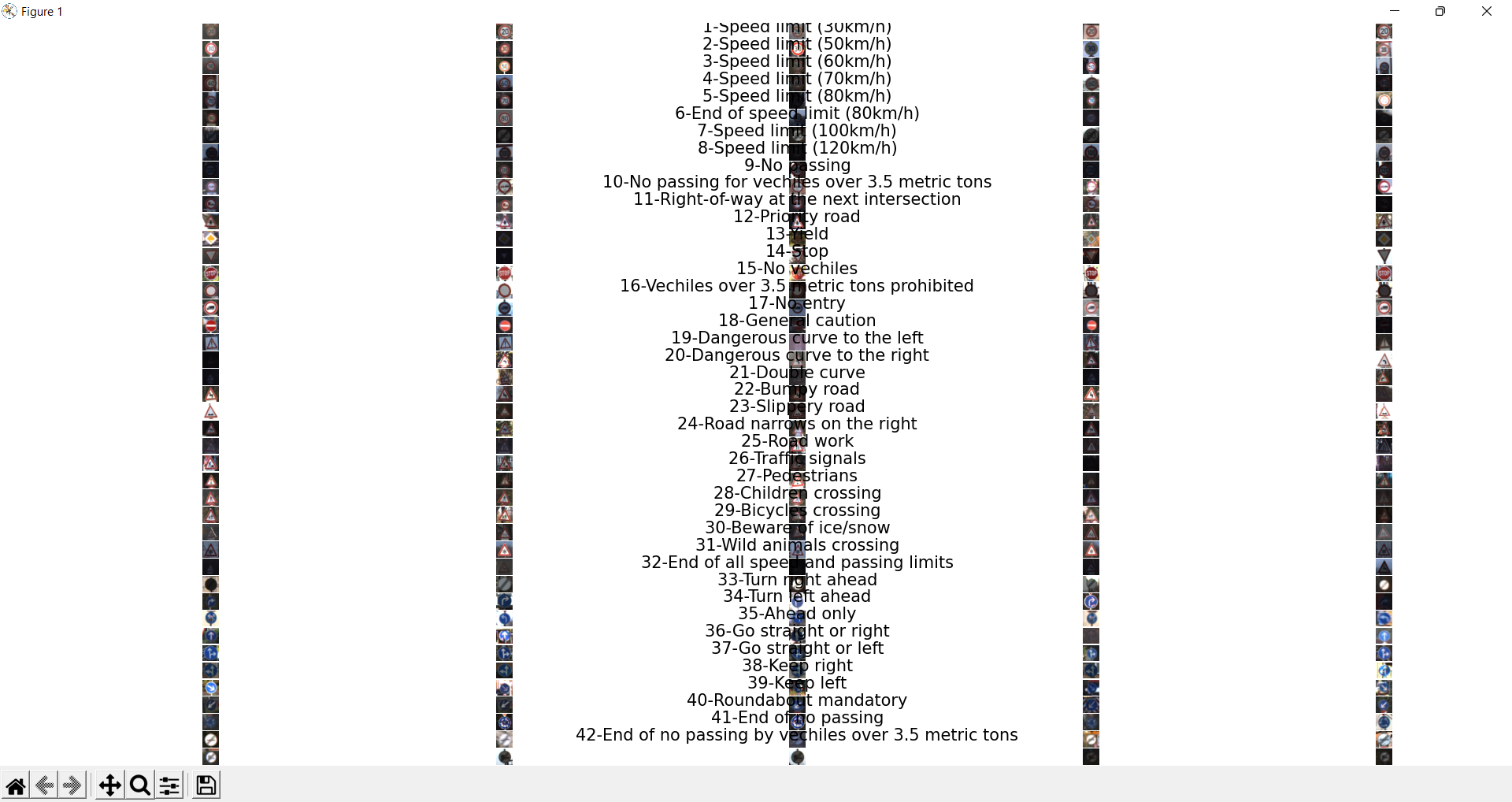
# **4. Training and Experimental Results**

To train and evaluate the model, the initial dataset was divided into the train and test datasets with the ratio 80/20 correspondently. More details, the Training dataset has a total of 22271 images with the size (32x32x3) and Validation has 5568 images with the size (32x32x3) and Test dataset has 6960 images with the size (32x32x3). My datasets are obtained from the GTSRB [6] and GTSDB [7] datasets and then are divided into the 43 most popular traffic signs. Fig.4 presents the distribution of the number of each class to the training dataset.



*Fig.4. Distribution of training dataset*

Besides, I also demonstrate all of the 43 most popular traffic signs and what each class is in Fig.5.



*Fig.5. Detail of each class*

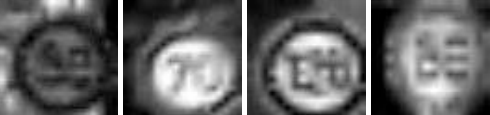
After preparing all of the necessary datasets for training I used my laptop ( AND Ryzen 5000 series) and it took me more than 1 hour to obtain the model. The training process ended shows results reaching precision equal to 99.89% when detecting a sign and 93.55% when classifying it.

The traffic sign images in Fig. 6 were effectively recognized by the CNN implementation suggested in this paper. The image demonstrates that the technology used produces good recognition results even with photographs of traffic signs, which are challenging for a person to recognize.



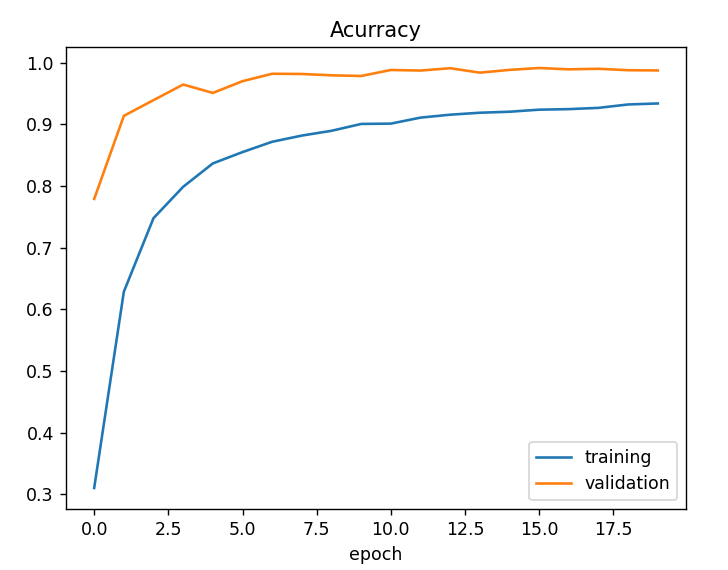
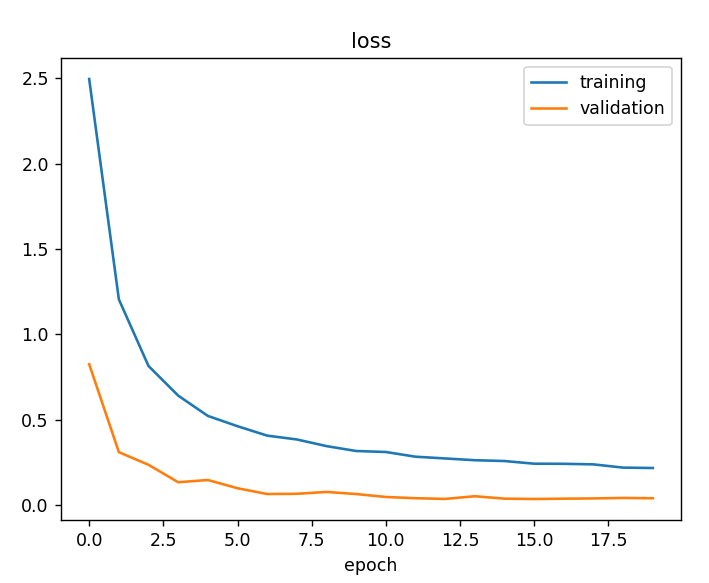
*Fig. 6. Successful classification*

However, the accuracy doesn’t reach 100 %. Fig. 7 shows the images of traffic signs that were recognized incorrectly



*Fig. 7. Unsuccessful classification*

Fig.8 also illustrates loss and accuracy during training with 20 epochs.



*Fig.8. Loss and Accuracy*

# **5. Conclusions**

The classification algorithm's implementation for the task of recognizing traffic signs is taken into consideration in this work.The suggested method for traffic sign classification exhibits very good results: 93.55% of correctly categorized photos when combined with preprocessing and localization procedures from other publications.

The use of our TSR algorithms allows the processing of video streams in real-time with CV2, and therefore at greater distances and with better quality than similar TSR systems have. FullHD resolution makes it possible to detect and recognize a traffic sign at a distance up to 50 m.

In future research, we plan to train the CNN to consider more traffic sign classes and possible bad weather conditions. Also, we plan to use a CNN not only for classification but for object detection too.

**References**

[1] A. Nikonorov, P. Yakimov, M. Petrov, Traffic sign detection on GPU using color shape regular expressions, VISIGRAPP IMTA-4, Paper 8 (2013).

[2] R. Belaroussi, P. Foucher, J.P. Tarel, B. Soheilian, P. Charbonnier, N. Paparoditis, Road Sign Detection in Images, A Case Study, 20th International Conference on Pattern Recognition (ICPR), 2010, pp. 484-488.

[3] A. Ruta, F. Porikli, Y. Li, S. Watanabe, H. Kage, K. Sumi, A New Approach for In-Vehicle Camea Traffic Sign Detection and Recognition,IAPR Conference on Machine Vision Applications (MVA), Session 15: Machine Vision for Transportation, 2009.

[4] V. Fursov, S. Bibkov, P. Yakimov, Localization of objects contours with different scales in images using Hough transform [in Russian],Computer Optics. 37, 4 (2013) 502-508.

[5] P. Yakimov, Tracking traffic signs in video sequences based on a vehicle velocity [in Russian], Computer Optics. 39, 5 (2015) 795-800.

[6] J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition,Neural networks. 32 (2012) 323-332.

[7] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, C. Igel, Detection of Traffic Signs in Real-World Images: The {G}erman {T}raffic {S}ign {D}etection {B}enchmark, in: Proc. International Joint Conference on Neural Networks, 2013.

[8] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, S. Hu, Traffic-Sign Detection and Classification in the Wild. Proceedings of CVPR, 2016, рp. 2110-2118.

[9] Y. LeCun, P. Sermanet, Traffic Sign Recognition with Multi-Scale Convolutional Networks, Proceedings of International Joint Conference on Neural Networks (IJCNN'11), 2011.

[10] P. Yakimov, Preprocessing of digital images in systems of location and recognition of road signs [in Russian], Computer Optics. 37, 3(2013) 401-405.

[11] V. Terehov, D. Efimov, I. Tiukin, Neural network control system [in Russian], Textbook for high schools, High school, 2002.

[12] Djork-Arné Clevert, Thomas Unterthiner, Sepp Hochreiter, Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs), 2015.