Genetic method of image processing for motor vehicle recognition

Ievgen Fedorchenko[1[0000-0003-1605-8066]], Andrii Oliinyk[2[0000-0002-6740-6078]], Alexander Stepanenko[3[0000-0002-0267-7440]], Tetiana Zaiko[4[0000-0003-1800-8388]], Anton Svyrydenko[5 [0000-0003-0114-8632]] and Dmytro Goncharenko[6[0000-0001-8973-3742]]

[1,2,4,5,6] Dept. of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia 69063, Ukraine

evg.fedorchenko@gmail.com, olejnikaa@gmail.com, alex@zntu.edu.ua, nika270202@gmail.com, antonsvyrydenko@gmail.com, gdimk99@gmail.com

Abstract. An analysis of scientific articles on similar topics was conducted; it was de-termined that the best recognition accuracy rate is achieved by using convo-lutional neural networks. Modifications of a simple genetic algorithm (Alfa-Beta, Alfa-Beta Fixed, Fixed) were developed. The implementation of the program for recognition of road users (cars, bicycles, pedestrians, motorcycles, trucks, etc.) was developed. Also, a comparison was made between the use of modifications of a simple genetic algorithm and the best approach for solving the problem of road user recognition. The purpose of the research conducted was to find an optimal approach for solving the problem of road user recognition, since the system that hasn't been implemented yet can recognize road users accurately. It was found that the improved Alpha-Beta modification is the best approach from the considered ones used to solve the problem. This modification allowed getting the best accuracy rate in less time selection, in comparison with other base modifications and simple genetic algorithm. The obtained results have a high practical value, since the developed modification allows optimizing the process of selection of values in other subject areas.

Keywords: Pattern Recognition, Genetic Algorithm, Evolutionary Algorithm, Neural Networks, Python, OpenCV, Keras.

1 Introduction

The theory of pattern recognition exists as a branch of informatics and related disciplines. It develops methods for the classification and identification of objects of various nature: signals, situations, objects that are characterized by a comprehensive number of certain features [1]. The problem of object recognition can also be a purpose of interdisciplinary researches - including work on creation of artificial intelligence. It is also often used in solving practical problems in the field of computer vision. When setting the classical problem of object recognition, it is necessary to apply a mathematical language based on logical thinking and mathematical principles. In contrast to this approach, there are methods for recognizing objects using machine learning and 6 artificial neural networks, formed not so formalized approaches to recognition, and show not worse, but in some cases, a much better result.

One of these areas is road safety, which includes many narrow areas: number-plate recognition, traffic congestion determination, recognition of road users in self-driving cars, etc.

However, for today, problems of recognition quality are not completely solved. For example, the vulnerability was detected in the Tesla Autopilot system: the system in most cases cannot recognize a bicycle. It identifies a bicycle as a person or a small car; accordingly, the system can make decisions that may endanger cyclist's life [1].

Vulnerabilities in recognizing road users in systems such as Mazda Smart City Brake Support [2], Waymo Autopilot [3] are also known.

It can be said that recognizing bicycles and other road users is a topical task. Recognition of road users can be used in the system of control and / or traffic safety or it can be used to study the reasons of constructing specialized bicycle lanes, etc.

2 Analysis of literary data and problem statement

This work [4] presents a study of solving the problem of bicycle recognition on a video stream by combining the following approaches: Histogram of Oriented Gradients (HOG) [5, 6], Support Vector Machine (SVM) [7], cascade classifier (Violi-Jones method) [8, 9].

Presented results of performing bicycle recognition in different weather conditions, and it is argued that the developed approach can be used in systems of real time recognition.

However, using of a combination of HOG and SVM methods is not optimal, since it allows you to recognize large enough objects that are not always present in video in real time and due to the fact that it takes a lot of time to calculate complex parameters. Using a cascaded classifier is more optimal in terms of recognition time. However, this method has errors in recognition - because of the application of different scales and the size of the scanning window, one object can be recognized as two.

In general, low image quality, weather conditions and various lighting conditions make precise recognition more complicated for the methods used. The approach developed is complex and complicated, and each method requires sufficient computational resources that effect on results.

An expedient approach to the implementation of the bicycles recognition would be the use of neural networks.

Thus, in [10], the study of the analysis of the flow of machines on the CCTV (video surveillance systems) images, based on the solution of the problem of recognition of machines with the help of neural networks Faster R-CNN (the faster region-based convolutional neural network) [11] and SSD (single shot multibox detector) [12].

In general, SSDs are capable of quickly recognizing, but are often mistaken for small objects, while Faster R-CNNs are slower than SSD, but can provide more precision.

Work aim is to test the specified trained neural networks on images of low quality and under different weather conditions. As a result of the study, it was found that the considered neural networks are not capable of recognizing all the machines, and in the case of bad weather conditions (rain, snow, fog), the accuracy indicator falls significantly. Therefore, the study of neural networks to detect vehicles should use more diversity training set. In addition, the study implements the recognition of only cars, although the concept of "traffic" covers several types of transport (motorcycles, trucks, etc.).

In [13], the solution of the machine recognition problem by studying a convolutional neural network [14] is considered, which is based on reinforced learning mistake and samples of error-prone probes. That is, in the training of the convolutional neural network, it is proposed to use the training errors of the current stage as training data in the next stage. A comparison of histogram oriented gradient recognition with convolutional neural networks, one of which was taught in the standard way, and the second - by the proposed authors, was made, which resulted in the proposed approach showed better performance.

However, there are no examples of car recognition, because of the small amount of study sample used in the study, one cannot say with complete certainty that a significant improvement in the training of the firewall neural network has been achieved.

In [15] the decision of the recognition problem among the crowd of each pedestrian separately is considered. The main purpose of the study is to recognize pedestrians in black and white images in various situations, the pedestrian's identification of the image, even with partial overlap, and with high accuracy, localize it. Also, the goal is to determine how many pedestrians are presented in the image.

The approach developed in this article is based on the scale-independent expansion of the Implicit Shape Model (ISM).

Authors succeeded in realizing all goals, but the accuracy of pedestrian recognition is 71.3% and is not satisfactory.

Also, it's worth noting that the developed method works only with images. However, the method has a great potential and can, as a result of further development, develop towards image recognition and tracking.

In [16] we consider the solution of the problem of pedestrian recognition by combining several neural networks.

The proposed system consists of two subsystems: the main pedestrian detection system for generation of all detections and a system of semantic segmentation to improve the results. Pedestrian Detection system in turn consists of a generator "candidates" for pedestrians and classification system. In order to provide more information about the coordinates of the object of the classification network, it is proposed to use the new method of soft tokens, which relates the object to all classes. For the implementation of the classification system the idea of "combined training" was implemented. Also, the technique of "soft rejection" is proposed for combining the conclusions of all networks of classification and the network of semantic segmentation with the conclusion of the network of "candidates" generation.

Pedestrian recognition has been tested on four popular image sets: Caltech Pedestrian set, INRIA set, ETH set, KITTI set. The first three sets received the highest recognition accuracy. Also, it indicates the significant speed of the developed approach.

In [17] the implementation of pedestrian recognition is considered, using the combination of the method of histogram oriented gradients, the method of reference vectors and the convolutional neural network.

The purpose of work is to construct a semantic areas of interest in order to get the foreground object is to reduce the errors associated with error detection background.

First, using a convolutional neural network trained by the Caltech Pedestrian set, the thermal map is generated by the input image. Further, semantic interest regions are extracted from the thermal map by morphological image processing. In the rest, semantic interest regions share the whole image on the background and foreground, to facilitate the work of the detectors who make the decision.

It is noted that with the help of semantic regions of interest, the work of detectors to varying degrees improves.

However, the use of a combination of HOG and SVM methods is not optimal, as it allows you to recognize large enough objects that are not always present in video in real time and due to the fact that it takes a lot of time to calculate complex parameters.

In work [18] the implementation of the system of recognition of vehicles belonging to 7 classes (motorcycle, car, pickup, bus, truck, truck with a trailer, truck with several trailers) is considered on the CCTV video. The Deep Convolutional Neural Network (DCNN) was used in the implementation.

The proposed system algorithmically consists of two tasks: localization and classification. Initially, the localization is performed, by generating regions independent of the class, for each frame, and then using a deep convolutional neural network to produce feature descriptions for each region. Finally, using the Support Vector Machines (SVM) method, for each region, the resulting description descriptions are compared with the templates, and a conformity assessment and classification are performed.

The accuracy of vehicle recognition differs for each class, but in general, the accuracy is within the range of 92 - 95%. The accuracy of the recognition significantly decreases when downloaded traffic, rain and fog, with low video quality.

The disadvantage of this development is the use of a deep convolutional neural network, since its calculation requires a lot of computational resources, which is not acceptable. The use of this neural network is possible with the presence of a powerful CPU (CPU) or graphics processor (NVIDIA GPU), in conjunction with such artificial intelligence environments as CUDA, Caffe, Torch.

Also, the system detects recognition errors on a small scale, then the object of one class is recognized as an object of another class, for example, the "truck" is recognized as a "truck with a trailer", and so on.

The analysis [4, 10, 13, 15-26] in the subject area suggests that conducting research on the implementation of the system traffic participant's recognition and increasing the accuracy of recognition is a rather topical task.

3 Purpose and objectives of the research

The purpose of the study is to develop a program for solving the problem of road users recognition - cars, bicycles, pedestrians, motorcycles, trucks. Also, it is necessary to develop an evolutionary method that would allow to obtain the best indicators of recognition accuracy when selecting the parameters of teaching neural networks at lesser time.

To achieve this goal, the following tasks were solved:

1. Conduct an analysis of existing methods for solving the pattern recognition problem.
2. To develop methods of evolutionary search for selection of parameters for teaching neural networks.
3. To implement the program realization of developed methods of evolutionary search.
4. To conduct the study of the effectiveness of the developed method.

4 Materials and methods of studying the influence of parameters in the training of neural networks

4.1 Equipment used in the experiment

The study of the influence of the values of parameters in the training of neural networks on the accuracy and time of training was carried out on a computer with an Intel Core i5 7400 processor with a clock speed of 3 GHz, with a memory capacity of 8 GB.

To study neural networks, a study sample of 8,000 images was divided into 5 classes and 3 sub-sets: educational (1000 images), test (200 images), verifying (400 images).

4.2 Methods of determining the influence of parameters of teaching neural networks

The study of the effect of values of the parameters of training neural networks was carried out using a simple genetic algorithm and its modifications.

The sequence of actions of a simple genetic algorithm is as follows: generation of an initial population *Ps* consisting of *N* individuals occurs:

$$P_s=(I_1,\dots,I_N), \tag{1}$$

where *I* - the individual.

Each individual *I* (its genes) is encoded by certain values from the set of all parameters of the set *Q*. Let the parameters of the neural networks have a set of parameters *Q*:

$$Q=(n_j,m,f_{act},f_{opt}), \tag{2}$$

where n - number of neurons of a certain layer,

$j = 1, \dots, m$,

m - number of layers,

f_{act} - activation function
f_{opt} is a function of optimization.
Then the individual I_N of the initial population Ps can be encoded by formula (3).

$$Q(n_i)=R[32,64,\dots,n_{max}],$$
$$Q(m)=R[1,2,\dots,m_{max}], \tag{3}$$
$$Q(f_{act})=R[relu,elu,\tanh,sigmoid],$$
$$Q(f_{opt})=R[adam,sgd,ada\max,nadam,rmsprop,adagrad,adadelta],$$

where $i = 1, \dots, m,$
R - function of choice of a random variable,
n_{max} - maximum number of neurons,
m_{max} - maximum number of layers.

$$I_N=R(Q). \tag{4}$$

Next, to evaluate the optimality of each individual I of the population of the P_S, the fitness function f_{fit} is calculated:

$$f_{fit}(P_s)=max(I_i), \tag{5}$$

where $i=1,\dots,N$.

The next step is to check if the conditions for completing the search for the optimal solution are not fulfilled. Such conditions can be the number of generations N, the value of the fitness function f_{fit}, runtime, and so on. In the example of selecting parameters for teaching neural networks, the value of fitness function is the accuracy of the trained neural network. If the conditions are fulfilled, it is necessary to complete the execution of the algorithm and output the results, the most optimal generation of the individual.

Then the selection of specimens is performed in a new population P_n: individuals are ranked according to the fitting indicator, 40% of the most adapted individuals (T) are selected in a new population. Also randomly selected 10% less fitting individuals (L). Another 50% of the new population P_n is obtained by crossing the selected individuals (K).

$$P_n=T+L+K. \tag{6}$$

Intersection (C) of individuals has the following algorithm: subsets of B and W randomly selected individuals on M and F having a specific set of genes:

$$M=((n_1^1,\dots,n_m^1),m^1,f_{act}^1,f_{opt}^1), \tag{7}$$

$$F=((n_1^2,\dots,n_m^2),m^2,f_{act}^2,f_{opt}^2).$$

Then the crossing can be expressed as follows:

$$C=(M\times F). \tag{8}$$

The result of the crossing of individuals is two descendants K_1, K_2, which can be written as follows:

$$K_1=(R[(n_1^1,\dots,n_m^1),(n_1^2,\dots,n_m^2)],R[m^1,m^2],R[f_{act}^1,f_{act}^2],R[f_{opt}^1,f_{opt}^2), \tag{9}$$

$$K_2=(R[(n_1^1,\dots,n_m^1),(n_1^2,\dots,n_m^2)],R[m^1,m^2],R[f_{act}^1,f_{act}^2],R[f_{opt}^1,f_{opt}^2]),$$

where R - function of the choice of a random variable.

This approach of crossing individuals is called uniform cross-linking, since the coding of the offspring randomly selects the values for each gene from each ancestor.

For the received descendants the mutation operator μ is applied according to (3):

$$\mu(K)=R[R(n_i),R(m),R(f_{act}),R(f_{opt})]. \tag{10}$$

Then, after applying the mutation operator, the descendant can be written as follows:

$$K_\mu=((n_i,\dots,n_m),m,f_{act},f_{opt}). \tag{11}$$

After applying the mutation operator descendants included the new generation P_n. The bursts and mutations are held until a new generation of P_n of size N is created:

$$P_n=((T+L),K_j), \tag{12}$$

where $j = 1, \dots, (N- (T + L))$.

Next, the f_{fit} fitness function for each new generation P_n (5) is calculated.

It is checked whether the conditions for completing the search for an optimal solution were not fulfilled. If the conditions are fulfilled, it is necessary to complete the execution of the algorithm and output the results, the most optimal generation of the individual. If the conditions are not fulfilled - continue the implementation of the algorithm, proceed to the formation of the next generation P_{n+1} based on the generation P_n.

The proposed modifications to a simple genetic algorithm are as follows:

1) modification Alfa-Beta: for crossing in each generation, a different number of couples for crossing is selected, in which one individual refers to the most suitable one and the other to the least adapted individuals. Also, random mutations (base and double) or one mutation (base) can occur: Monte Carlo method generates a random number 0 or 1. If 0 falls, then one mutation arises if there is one mutation - there are two mutations.

The sequence of actions of this modification of a simple genetic algorithm is similar to its basic version, but it has some differences. At the stage of selection of individuals to a new population of P_n individuals are ranked according to the fitness parameter, then the number of pairs is determined randomly - a certain number of the most adapted individuals, and the same number of the least adapted.

The most suitable individuals form a subset of B, the least adapted - subset of W. Both subsets are included in the set of pairs V. The number of individuals that can be chosen in pairs is in the range of 20-60% of the total number of individuals. The rest of the new population P_n is obtained by crossing the selected individuals (K).

$$P_n=(V,K_j), \tag{13}$$

where $j = 1, ..., (N-V)$.

Also, in the proposed modification, two ($\mu 1, \mu 2$) or one mutation (μ) may occur randomly, whereas in the basic version of the algorithm, one mutation arises randomly. Moreover, in a situation when there are two mutations - one gene can mutate twice. After applying the mutation operator descendants included the new generation P_n. Crossing and mutation are held as long as we create the next generation of P_n size N (13).

2) modification Alfa-Beta fixed: the number of couples for crossing is selected in each generation, with one person referring to the most suitable one and the other to the least adapted individuals. Also, random mutations (base and double) or one mutation (base) can occur: Monte Carlo method generates a random number 0 or 1. If 0 falls, then one mutation arises if there is one mutation - there are two mutations . A fixed point of crossing is established - the first half of the genes participate in the crossing - the genes responsible for the number of neurons in the layers, the values of other genes are always passed on to the descendants of one of the individuals.

The sequence of actions of this modification is similar to the sequence of Alfa-Beta modifications, however, the crossing operation is excellent: according to formula 7, subsets B and W are randomly selected by the individual M and F having a certain set of genes. Intersection can be expressed as:

$$C^* =((\frac{1}{2} M \times \frac{1}{2} F), \frac{1}{2} F). \tag{14}$$

The result of the crossing of individuals is two descendants K_1, K_2, which can be written as follows:

$$K_1=(R[(n_1^1, ..., n_m^1),(n_1^2, ..., n_m^2)], m^2, f_{act}^2, f_{opt}^2), \tag{15}$$

$$K_2=(R[(n_1^1, ..., n_m^1),(n_1^2, ..., n_m^2)], m^2, f_{act}^2, f_{opt}^2),$$

where R is a random variable selection function,

$m^2, f^2{}_{act}, f^2{}_{opt}$ - the value of the genes transmitted from the individual F.

3) modification Fixed: a fixed point of crossing is established - half of the genes are involved in the crossing - the genes responsible for the number of neurons in the layers, the values of other genes are always passed on to the descendants of one of the

individuals. Also, in the mutation stage randomly, there are two mutations (base and double) or one mutation (base): the Monte Carlo method generates a random number, 0 or 1. If 0 falls, then there is one mutation, if there is 1 - occurs two mutations. The sequence of actions of this modification is similar to the sequence of Alfa-Beta modification, but there is a difference: the selection of individuals to a new population occurs both in the basic version of the genetic algorithm (6), and the crossing occurs as in the second modification (14), (15).

5 Experiments and results of research on the use of modifications of a simple genetic algorithm

There were 4 experiments in which the genetic algorithms and the simple simple genetic algorithm were considered. All experiments were carried out with the same parameters: the number of generations - 5, the size of the population - 20, the mutation rate - 0.01. Input data is a sample of 5-digit images (pedestrian, bicycle, motorcycle, auto, truck), in which each class has 1600 images.

To modify Alpha-Beta and a simple genetic algorithm, the elite coefficient was also set at 0.4 and the randomly selected individuals were 0.1.

In the figures below, graphs of parameter selection parameters are presented in the training of the neural network by the proposed modifications of the genetic algorithm. After selecting the parameters of learning by the genetic algorithm and its modifications, the recognition accuracy for the best neural network and the selection time were obtained. These results are shown in the Table 1. Figure 1 shows a graph of the mean accuracy for each generation of selection parameters for training neural networks. Figure 2 shows a chart of indicators for the 5 best individuals of the last generation of parameter selection for each neural network.

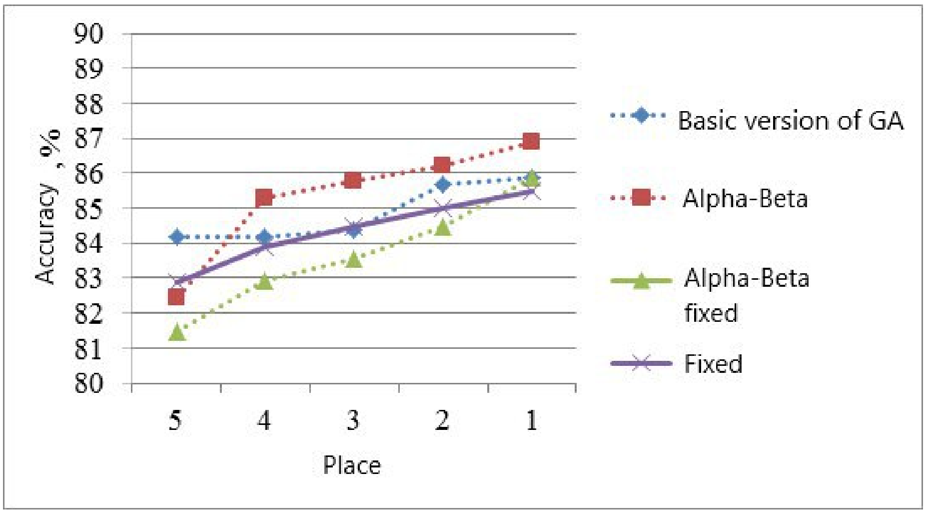

Fig.1. Schedule of mean accuracy for selection of parameters for teaching neural networks

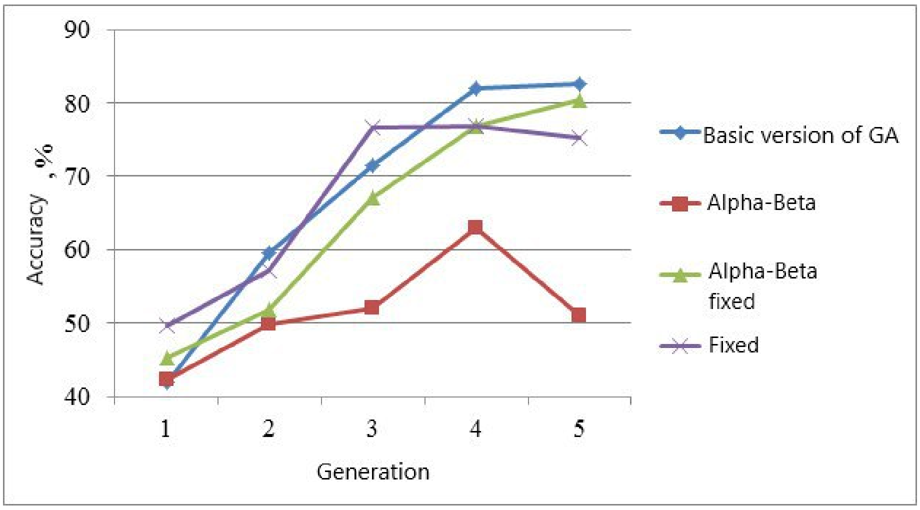

Fig.2. Schedule of accuracy indicators of the last generation when selecting parameters for teaching neural networks

Table 1. Results of selection of parameters using a simple genetic algorithm

GA version	Time of training	Accuracy on test data	Accuracy of auto-testing
Basic	8 days, 8 minutes, 39 seconds	79.45 %	85.89 %
modification Alpha-Beta	7 days, 7 hours, 16 minutes,43 seconds	82.12 %	86.90 %
modification Alpha-Beta fixed	10 days, 12 hours, 4 minutes, 58 seconds	80.02 %	85.89 %
modification Fixed	5 days, 22 hours, 27 minutes, 5 seconds	78.48 %	85.48 %

Based on the results presented in table 4, we can conclude that the use of the Alpha-Beta modification of the genetic algorithm is the best approach to achieving higher recognition accuracy in less time.

6 Discussion of the results of research on the use of modifications of a simple genetic algorithm

In the Fig. 4 and 5 are graphs of the execution time (in minutes) of the proposed method on computer systems, which depends on the number of cores involved. It can be seen from the graphs that the proposed method has an acceptable degree of parallelism and is effectively performed on both MIMD and SIMD systems. This way, the IPME cluster was able to reduce the method execution time from 1565 minutes (on one core) to an acceptable 147 minutes on 16 cores. On the ZNTU the computing system, the method execution time was reduced from 1268 minutes on a single core to 110 minutes on 16 cores. The differences in the performance of the systems are due to their architectural features: in the cluster cores are connected by means of the InfiniBand communicator, and in the multi-core computer they are located on a single chip, which explains the smaller impact of overhead (transfers and synchronizations). In addition, the processor in multi-core computer supports Turbo Boost technology [25-32], making the time of the method execution on the single core much less than the execution time on the core of the cluster that does not support this technology.

As a result of the study, a modification of a simple genetic algorithm - Alfa-Beta - was developed. Using this modification allows you to speed up the selection of the parameters of training neural networks, and increase the accuracy. It has been shown that the increase in the number of mutations and selection in pairs of different individuals provides a greater variety of gene combinations, which leads to better performance in less time.

Fig. 3-7 shows examples of recognition using a trained neural network.

Fig. 3. Example of recognizing the object "bicycle"

Fig. 4. Example of recognizing the object "car"

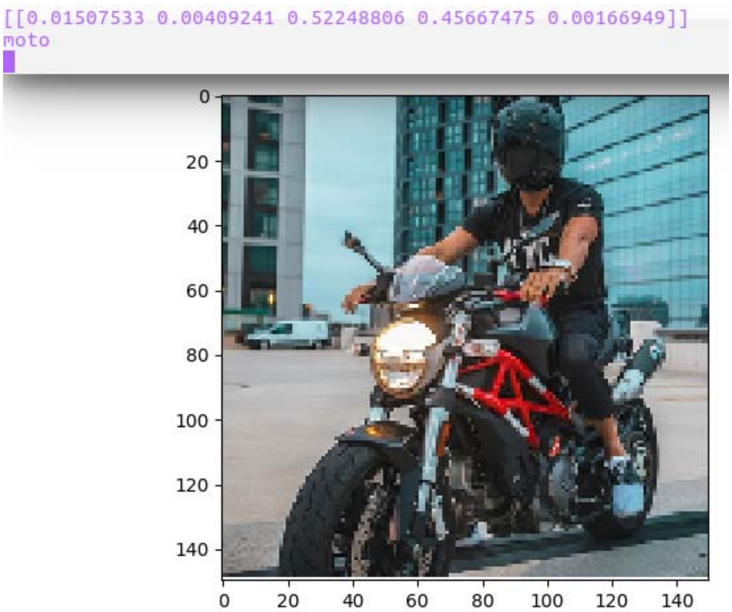

Fig. 5. Example of recognizing the object "motorcycle"

Fig. 6. Example of recognizing the object "pedestrian"

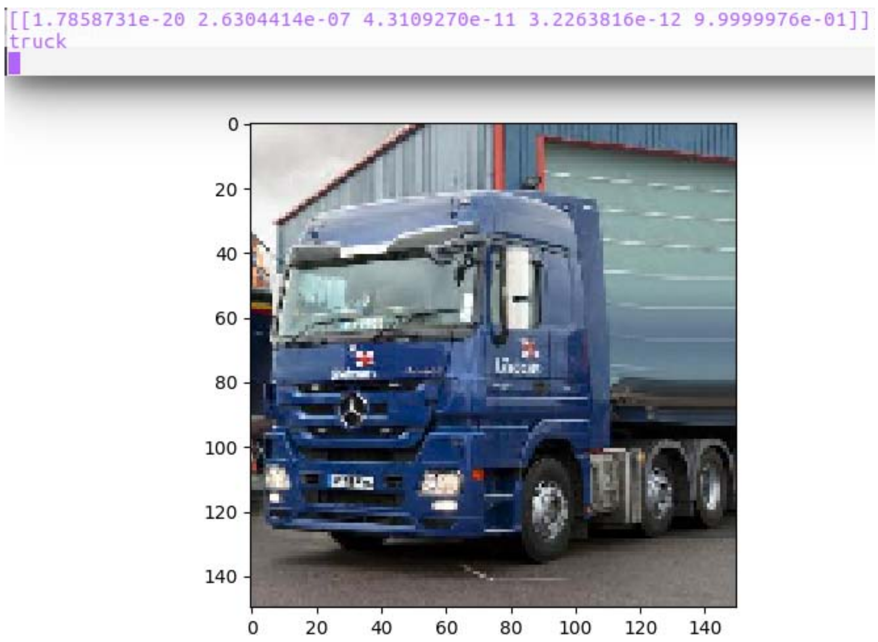

Fig. 7. Example of the recognition of the object "truck"

Based on the recognition results, we can conclude that a trained neural network is able to confidently determine the relevance of an object to a particular class.

7 Conclusion

An analysis of existing methods of pattern recognition from scientific articles on similar problems was carried out. It was determined that the best recognition accuracy rate is achieved using convolutional neural networks.

Three modifications of a simple genetic algorithm (Alfa-Beta, Alfa-Beta fixed and Fixed) were developed for selecting parameters of training neural networks.

A program using the developed modifications of a simple genetic algorithm for selecting parameters of training neural networks and recognition of road users was implemented. When applying the developed modifications, the following results were obtained:

- Alpha Beta - accuracy 86.90%, selection time - 7 days, 7 hours, 16 minutes, 43 seconds;
- Alpha Beta fixed - accuracy 85.89%, selection time - 10 days, 12 hours, 4 minutes, 58 seconds;
- Fixed - accuracy 85.48%, selection time - 5 days, 22 hours, 27 minutes, 5 seconds.

Comparison of the rates of a simple genetic algorithm was conducted; it was determined that the Alpha-Beta modification is the best approach to recognize road users since this modification allowed better accuracy (86.90%) for shorter selection time (7 days, 7 hours, 16 minutes, 43 seconds).

Acknowledgment

The work was performed as part of the project “Methods and means of decision-making for data processing in intellectual recognition systems” (number of state registration 0117U003920) of Zaporizhzhia National Technical University.

References

1. Lugovskaya, A. M.: Artificial neural networks in medical diagnostics [Iskusstvennyie neyronnyie seti v meditsinskoy diagnostike]. Computer systems and networks: proceedings of the 54th scientific conference of postgraduates, undergraduates and students 2018, vol. 1, pp. 182 – 183. BGUIR, Minsk (2018).
2. Goold, S.D., Lipkin, M.Jr.: The doctor-patient relationship: challenges, opportunities, and strategies. Journal of General Internal Medicine (1), 23–33 (1999).
3. Van Tuc, N.: Approximation contexts in addressing graph data structures. University of Wollongong Thesis Collection, 30–55 (2015).
4. Barkoulas, J. T., Baum, Ch. F., Travlos, N.,: Long Memory in the Greek StockMarket. Applied Financial Economics, vol. 10, no. 2, 177–184 pp. (2000).
5. Dalal, N. Histograms of Oriented Gradients for Human Detection. CVPR, 886–893 (2005).

6. Schmidhuber, J., Wierstra, D., Gagliolo, M., Gomez, F.: Training Recurrent Networks by Evolino. Neural computation. vol. 19(3), 757–779 pp. (2007). doi: 10.1162/neco.2007.19.3.757.
7. Bland, J.M., Altman, D.G.: Statistics notes: measurement error. BMJ. 312 (7047): 1654 (1996). doi:10.1136/bmj.312.7047.1654
8. Viola, P., Jones M.J. Rapid Object Detection using a Boosted Cascade of Simple Features. Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 1–9 (2001).
9. Viola, P. Robust real-time face detection. International Journal of Computer Vision, 57, 2, 137–154 pp. (2004).
10. Peppa M. V., Bell D., Komar T., Xiao W. Urban traffic flow analysis based on deep learning car detection from CCTV image series. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., 499–506 pp. (2018).
11. Ren S., He K., Girshick R., Sun J.. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 6, 1137–1149pp. (2017)
12. Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C., Berg A. SSD: Single Shot MultiBox Detector. Computer Vision ECCV, 21–37 (2016).
13. A CNN Vehicle Recognition Algorithm based on Reinforcement Learning Error and Error-prone Samples, http://iopscience.iop.org/article/10.1088/1755-1315/153/3/032052/pdf .
14. Yarymbash, D., Yarymbash, S., Kotsur, M., Divchuk, T.: Analysis of inrush currents of the unloaded transformer using the circuitfield modelling methods. Eastern-European Journal of Enterprise Technologies, vol. 3, № 5 (93), pp. 6-11 (2018). doi: 10.15587/1729-4061.2018.134248
15. Boden M.: A guide to recurrent neural networks and backpropagation. Halmstad University, pp. 1–10 (2001).
16. Yarymbash, D., Yarymbash, S., Kotsur, M., Divchuk, T.: Enhancing the effectiveness of calculation of parameters for short circuit of threephase transformers using field simulation methods. Eastern-European Journal of Enterprise Technologies, vol. 4, № 5 (94), pp. 22-28 (2018). doi: 10.15587/1729-4061.2018.140236.
17. Whitley D. Genetic Algorithms and Neural Networks. Genetic Algorithms in Engineering and Computer Science, pp. 203-216, (1995).
18. Yarymbash, D., Yarymbash, S., Kotsur, M., Divchuk, T.: Analysis of inrush currents of the unloaded transformer using the circuitfield modelling methods. Eastern-European Journal of Enterprise Technologies, vol. 3, № 5 (93), pp. 6-11 (2018). doi: 10.15587/1729-4061.2018.134248
19. Oliinyk, A., Subbotin, S., Lovkin, V., Leoshchenko, S., Zaiko, T.: Development of the indicator set of the features informativeness estimation for recognition and diagnostic model synthesis. 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET 2018), pp. 903–908, (2018). doi: 10.1109/TCSET.2018.8336342.
20. Leoshchenko, S., Oliinyk, A., Subbotin, S., Zaiko, T.: Methods of semantic proximity extraction between the lexical units in infocommunication systems. 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T 2017), 7–13 pp. (2017). doi: 10.1109/INFOCOMMST.2017.8246137.
21. Leoshchenko, S., Oliinyk, A., Subbotin, S., Gorobii, N., Zaiko, T.: Synthesis of artificial neural networks using a modified genetic algorithm. Proceedings of the 1st International Workshop on Informatics & Data-Driven Medicine (IDDM 2018), pp. 1-13 (2018). dblp key: conf/iddm/PerovaBSKR18

22. Stepanenko, O., Oliinyk, A., Deineha, L., Zaiko, T.: Development of the method for decomposition of superpositions of unknown pulsed signals using the secondorder adaptive spectral analysis. EasternEuropean Journal of Enterprise Technologies, vol. 92, issue 2/9, pp. 48–54, (2018). doi: 10.15587/1729-4061.2018.126578
23. Alsayaydeh, J.A., Shkarupylo, V., Hamid, M.S., Skrupsky, S., Oliinyk, A.: Stratified Model of the Internet of Things Infrastructure, Journal of Engineering and Applied Science, vol. 13, issue 20, pp. 8634-8638, (2018). doi: 10.3923/jeasci.2018.8634.8638.
24. Shkarupylo, V., Skrupsky, S., Oliinyk, A., Kolpakova T.: Development of stratified approach to software defined networks simulation. EasternEuropean Journal of Enterprise Technologies, vol. 89, issue 5/9, pp. 67–73 (2017). doi: 10.15587/1729-4061.2017.110142
25. Kolpakova, T., Oliinyk, A., Lovkin, V.: Improved method of group decision making in expert systems based on competitive agents selection. IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), Institute of Electrical and Electronics Engineers, pp. 939–943, Kyiv (2017). doi: 10.1109/UKRCON.2017.8100388
26. Oliinyk A., Fedorchenko I., Stepanenko A., Rud M., Goncharenko D.: Evolutionary method for solving the traveling salesman problem. Problems of Infocommunications. Science and Technology : 5th International Scientific-Practical Conference PICST2018, Kharkiv, 9–12 October 2018, Kharkiv, Kharkiv National University of Radioelectronics, pp. 331-339 (2018). doi: 10.1109/INFOCOMMST.2018.8632033