ВЕСТНИК ТОМСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА
2020 Управление, вычислительная техника и информатика № 51

ОБРАБОТКА ИНФОРМАЦИИ

УДК 004.855
DOI: 10.17223/19988605/51/8

A.A. Druki, V.G. Spitsyn, E.U. Arkalykov

SEMANTIC SEGMENTATION ALGORITHMS OF THE EARTH'S SURFACE PICTURES BASED ON NEURAL NETWORK METHODS

The reported study was funded by RFBR according to the research project № 18-08-00977A and in the framework of Tomsk Polytechnic University Competitiveness Enhancement Program.

The aim of the work is to develop algorithms that solve the problem of semantic segmentation of images. A convolutional neural network with an original architecture was developed. Performing a software implementation of the algorithm, which allows to build a map of segmented objects of a different class. A comparison of the results of the proposed algorithm with existing analogues is presented.
Keywords: computer vision; artificial neural networks; semantic segmentation; image processing.

Semantic image segmentation is the division of the original image into local areas (segments) corresponding to certain classes of objects. These systems are used in a wide variety of fields from security systems to various types of medical diagnostics. Today, methods and algorithms for classifying objects in images are actively developing, however, this task has not been fully solved and part of the work is done by people manually, which leads to errors and time costs.

There is a wide selection of computer vision methods and algorithms that can be used to solve this problem [1]. The main approach for semantic image segmentation is a combination of three algorithms: detectors, descriptors and classifiers. These algorithms determine the basic parameters of the image, select objects and classify them. Such image parameters include brightness, color, texture, borders and angles of objects, and the like. Image parameters are fixed using detector algorithms, and then mathematically described using descriptor algorithms. Next, you need to determine to which class the objects belong, classifier algorithms are involved in this procedure.

Among the detectors and descriptors, the following algorithms can be distinguished:

– SIFT detector and descriptor (Scale invariant feature transform) [2];
– SURF (Speeded up robust features) [3];
– FAST (Features from accelerated segment test) [4];
– MSER (Maximally stable extremal regions) [5];
– HOG (Histogram of oriented gradients) [6].

These algorithms in many respects have a similar principle of operation and results. Their advantage is a rather high resistance to various noise distortions in the frame and to change the scale of the object. The disadvantage is the decrease in stability when changing the camera angle, poor lighting and in the presence of reflective surfaces in the frame.

In machine learning, many different classification algorithms and their modifications are presented, so we will consider two classes of widely used classifiers: Bag of words and SVM (Support Vector Machine) and neural networks.

This algorithm is one of the most common classes of image classification algorithms. Its name largely determines the principle of its work, because in fact it uses a histogram of occurrences of individual templates in the image [7].

Basically, this algorithm was used to classify texts, in which a histogram was also constructed of the entry of words from a pre-prepared dictionary into a document, but can be effectively applied to other tasks.

The main disadvantages of this classifier in the large size of the dictionary, and its size should not exceed a certain value at which it will be resistant to noise. Also, the bag of words does not take into account spatial information about the object in any way, which, if there are similar points in descriptors of different points of different objects in the image, their descriptions may coincide [7].

The support vector machine (SVM is support vector machine) is one of the most popular methods for training classifiers. Initially, this method was used for binary classification problems, but it can be easily generalized for problems with a large number of classes, as well as for regression restoration problems.

The advantages of this algorithm in addition to the prevalence and simplicity include a small training sample, in which the classifier will provide an acceptable result. This is also a drawback: the algorithm does not use the whole set, but only a small part of them at the boundaries of the regions [8].

The main disadvantage of classical neural networks for image processing is the large size of the input vector (i.e. each pixel of the image). As a result of this, the number of neurons in each layer is growing, and for large images the network becomes too large and heavy for training. Also, classical neural networks cannot take into account the topology of the original image, since they accept it in its entirety [9].

Convolutional neural networks (CNN), which have special convolutional and subsampling layers, are devoid of these shortcomings. The basic idea of these layers is to create shared weights, i.e. some neurons use the same weights and are combined into feature maps, each neuron of which is associated with the previous layer. Each neuron of such a layer performs a mathematical convolution operation on a certain area of the previous layer, and such a layer is accordingly called a convolutional layer and models some features of human vision, responsible for detecting a specific symptom. Layers of downsampling are responsible for reducing the input vector by a number of times (usually 2 times).

Thus, CNN have a much smaller number of customizable weights, which allows the network to learn generalization of information, rather than memorization. Such networks provide high resilience to zooming, shear and bends.

1. Development of segmentation algorithm

Convolutional neural networks were chosen to develop an image segmentation algorithm. Despite the active development of neural network algorithms, to date, there are still no specific rules for choosing the structure and such networks. As a rule, developers experimentally choose network parameters: the number and organization of layers, the number and size of feature maps, the size of convolution matrices, the choice of a learning algorithm, the activation function, etc. You need to understand that too many network parameters can increase computational complexity, while this does not guarantee improved work results. Too few parameters can lead to reduced classification accuracy. Thus, the goal is to choose a neural network architecture that would provide high performance with the least number of tunable parameters.

In the process of doing work, several libraries of neural network construction were studied.

TensorFlow is an open software library for machine learning developed by Google to solve the problems of building and training a neural network in order to automatically find and classify images, achieving the quality of human perception. It is used both for research and for developing Google's own products. The main API for working with the library is implemented for Python, there are also implementations for C ++, Haskell, Java [10].

Torch is a library for scientific computing with broad support for machine learning algorithms. Developed by Idiap Research Institute, New York University and NEC Laboratories America. The library is implemented in Lua using C and CUDA. The fast scripting language Lua in combination with SSE, OpenMP, CUDA technologies allows Torch to show good speed compared to other libraries. Currently supported operating systems are Linux, FreeBSD, Mac OS X. The main modules also work on Windows [11].

Keras is an open neural network library written in Python [12]. It is an add-on to the frameworks TensorFlow and Theano. It is aimed at the operational work with deep learning networks, while it is compact, modular and expandable.

For comparison, library libraries used the CIFAR-10 data set containing 10 classes of images [13]. In total, the data set contains 60,000 color images 32 by 32 pixels in size, where for each class there are 6,000 images, respectively. Classes included in CIFAR-10: airplanes, cars, trucks, birds, cats, deer, dogs, frogs, horses and ships.

According to the results of the training time, the Torch library was the fastest to learn 168 seconds, the TensorFlow and Keras libraries showed about 250 seconds. The libraries of Torch and TensorFlow show the same results in terms of accuracy 78%, Keras is one percent less 77%.

All libraries have similar performance, but Torch has an advantage, since the dynamic graph of calculations allows you to change the structure of the neural network before each program start, without preliminary compilation, which allows you to spend less time building the architecture of the neural network, so the Torch library was chosen for further development.

In the process of experimental studies, CNN of various architectures were implemented, including a different number of parameters. The experiments showed that neural networks with a simplified architecture and a small number of parameters showed worse results. With a sequential complication of the architecture of the CNN, it was possible to select the optimal architecture that provides high classification results (Fig. 1). Further experiments to complicate the architecture and increase the number of SNA parameters did not provide an improvement in the quality of classification, but at the same time, the time and network training increased significantly.

This neural network consists of 21 layers and includes 11 convolutional layers, 4 pooling layers, 4 upsampling layers and one resulting layer.

As input, color images are used. The input layer has a size of 256 × 256 neurons. This layer does not perform any transformations and is intended only for supplying input data to it.

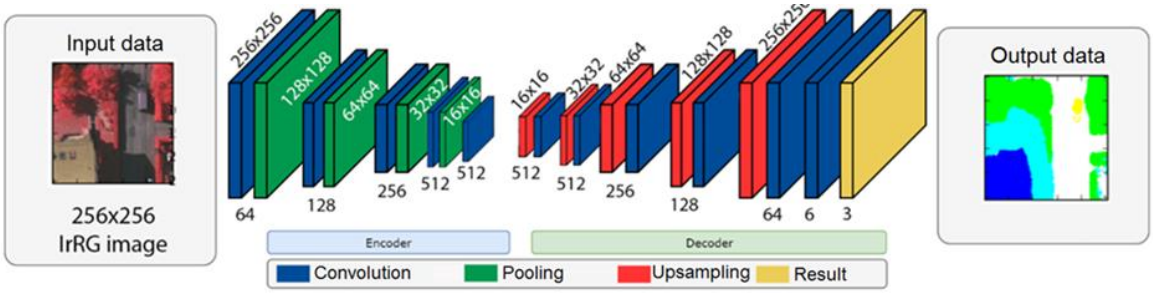

Fig. 1. Full convolution network architecture

Next to the input layer is the first hidden layer. This layer is a convolutional one, contains 64 feature maps, each of which has a size of 256 × 256 neurons.

The second hidden layer is a subsampling layer; it consists of 128 feature maps, each of which has a size of 128 × 128 neurons. The convolution matrix has a size of 2 × 2 neuron. Displacement is performed by 1 neuron. This layer reduces the size of the previous layer by half.

The third hidden layer C2 is a convolutional layer and consists of 128 feature cards, each of which has a size of 128 × 128 neurons. The convolution matrix has a size of 4 × 4 neuron. Displacement is performed by 1 neuron.

The fourth hidden layer is also a convolutional layer and consists of 256 feature cards of the size of 64 × 64 neurons. The convolution matrix has a size of 4 × 4 neuron. Displacement is performed by 1 neuron.

The fifth hidden layer is also a convolutional layer and consists of 512 symptom cards of size 32 × 32 neuron. The convolution matrix has a size of 4 × 4 neuron. Displacement is performed by 1 neuron.

The sixth hidden layer is a subsample layer; it consists of 512 feature maps, each of which has a size of 32 × 32 neurons. The first five layers of the network have a two-dimensional structure and are designed to extract features in the image. The sixth layer is also a convolution layer. Upsampling layers are used to enlarge the image to its original values, which take the data for enlargement from the subsample layers. At the last stages, we get a convolutional layer with 6 feature cards, which we later translate into 3 channels and get a classified image at the output.

2. Neural network training

For training neural networks, the error back propagation algorithm and its various modifications are usually used. The classical algorithm has a number of drawbacks, divergence is possible in some situations, if you choose a learning speed that is too high, if the convergence rate is too low, there is a chance of retraining. It was decided to carry out network training using the following gradient descent optimizers:

Nesterov Accelerated Gradient. This optimization method is based on the idea of momentum accumulation, i.e. with prolonged movement in one direction, the speed will remain for some time after. To do this, you need to store several previous values and calculate the average. The calculation of the average value takes up too much memory for a large number of occurrences, therefore, the average estimate is used [14].

Adagrad-adaptive Gradient. This is a family of algorithms, the main idea of which is to save some rarely found signs to protect them from noise. For this, a certain quantity is created, for example, the sum of the squares of updates or their modules for a certain parameter of an artificial neural network. Based on this value, element updates are regulated – frequently encountered updates are less common, freeing up space for rare ones, thereby producing an adaptive learning speed or attenuation of the learning speed [15].

Adam-adaptive Moment Estimation. This optimization method combines the pulse accumulation considered in the Nesterov method, as well as storing the frequency of the gradient change, similar to Adagrad [16].

To train the neural network, a data set from ISPRS.com for the city of Vaihingen Germany was used (Fig. 2). The spatial resolution of images is 9 cm per pixel. Image channels comprise near infrared-red-green (IRRG). The city is divided into sectors, each of which has its own unique number (Fig. 2).

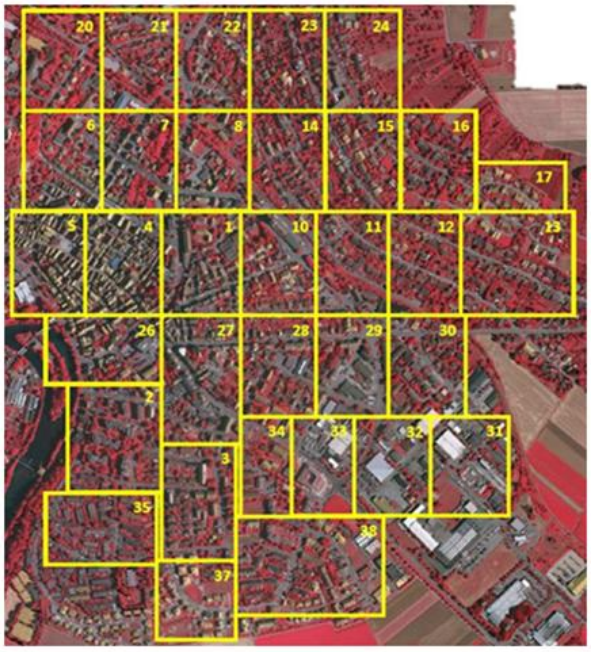

Fig. 2. Sectors of the city of Vaihingen

Depending on the color, each mask element represents a specific object in the picture. Roads and sidewalks are white, buildings are blue, trees are green, grass and bushes are turquoise, vehicles are yellow. A total of 22 images were used for training and 15 images for verification. The images used have a resolution of about 2000 by 2500 pixels [17]. The training was carried out on 50 eras of 2000 iterations in each, where fragments of 256 by 256 pixels were used (Fig. 3).

A convolutional neural network processes the input image with a scanning window of 64 × 64 pixels in accordance with the size of the input layer. In each section, the neural network performs data segmentation, forming the corresponding map with a size of 16 × 16 pixels at the output. This size difference is due to the fact that when sampling a small portion of the image it is often difficult to find out what is shown on it.

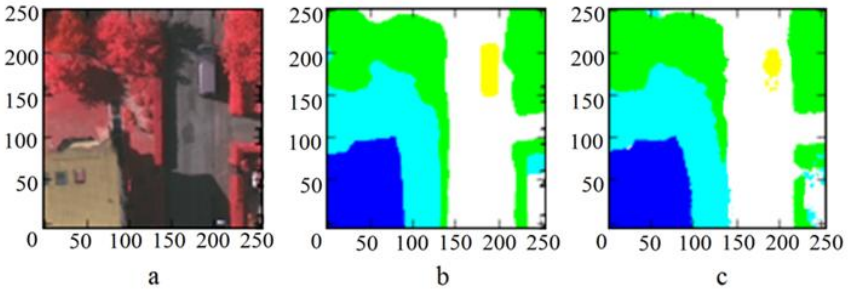

Fig. 3. The learning process:
a) the source image; b) a training example; c) the predicted result

To eliminate the problems of retraining, the DropOutmethod is included in the fifth fully connected network layer [18]. This method consists in randomly allocating some area of the neural network in which weights are updated. Neurons get on the subnet with a probability of 0,5.

We also used the L2 network regularization method, which consists in a large fine of too high a weight value and a small one at a low value. Also, during training, the loss function was minimized using the Mini-batch gradient descent method [18].

To increase the network stability to various rotations and distortions and to increase the training set, the original images were rotated at a random angle.

For training and testing, the following network parameters were selected:

– learning coefficient 0,0005;
– the frequency of changes in the learning coefficient 104;
– the magnitude of the change in the learning coefficient of 0,1;
– attenuation for regularization L2 0,0005.

The experiments were carried out on a PC with the following characteristics: Intel Core i5-4690k (4GHz) processor, 8GB RAM, NVIDIA Quadro FX 4800 video adapter.

The training of the developed neural network was carried out using three training algorithms: Nesterov accelerated gradient, AdaGrad and Adam. The structure and parameters of the network remained unchanged. The number of training eras is 500. Table 1 presents the learning outcomes.

Table 1

CNN learningoutcomes

Algorithmname	Studyingtime		Accuracy (%)
	h.	min.	
Nesterovacceleratedgradient [16]	14	46	85,56
AdaGrad [17]		30	82,24
Adam [18]		28	89,2

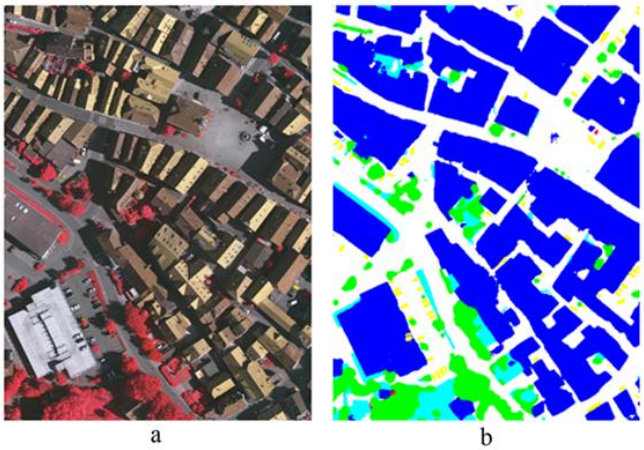

Fig. 4. The result of image segmentation:
a) the original image; b) the result obtained

The results presented in table 1 show that the Adam algorithm showed the best results relative to the rest: training time 14 hours 28 minutes, classification accuracy 89,2%. Accuracy for each class of objects: roads 90,89%, buildings 94,65%, trees 89,89%, bushes and grass 80,62%, cars 89,32%.

Fig. 4 shows the results of image segmentation developed by a neural network.

The results of the proposed neural network were compared with the results of the work of analogues (Table 2).

Table 2

Comparison with analogues

Name	Roads (%)	Buildings (%)	Grassandbushes (%)	Trees (%)	Cars (%)	Total (%)
EPSO [19]	91	93	81,3	88,3	83	87,32
PSO k-MEANS [19]	90,6	93,2	82	88,8	86,6	88,24
EDGEFLOW [20]	92,7	95,3	84,6	89,2	85,8	90,3
EDISON [20]	93,9	96	85,6	89,8	89,8	91
JSEG [20]	93,7	96	87,6	89,9	88	91
MULTISCALE [20]	91,5	93,7	81,5	88,4	84,9	88
Developedsystem	92,8	95,7	81,6	89,9	89,3	89,8 (+0,023;–0,04)

A comparative study of the accuracy of segmentation algorithms was carried out on a set of reference and test images subjected to noise distortion. To compare the results of segmentation, we used the boundaries of segmented objects, which is a set of points independent of the shading of the segments. To measure the results of segmentation, two metrics were used: average and Hausdorff distance [20]. A study of the quality of work of a number of popular segmentators showed that they behave unstably when noisy and blurry. Thus, we can conclude that it is advisable to clean the image from noise and increase its clarity before the segmentation procedure.

From table 2 it is seen that the developed system provides an accuracy of 89,8%, with a confidence interval: +0,023; –0,04. The results of the analogues are obtained from open sources. As you can see, the developed system is somewhat inferior in terms of accuracy to some analogs in such indicators as roads, buildings, grass. However, in terms of indicators such as trees and cars, she wins.

Conclusion

In the process of performing the work, a convolutional neural network was implemented, which performs segmentation of objects in the images of remote sensing of the Earth into the following classes: roads, buildings, trees, grass and cars. Experiments were conducted on the selection of a learning algorithm. The best results were obtained using the Adam algorithm.

The results of the neural network showed a classification accuracy of 89,9%. When compared with analog systems, the developed neural network is inferior in some respects, but generally shows good results and a competitive level.

This work was supported by the Russian Foundation for Basic Research (RFBR) in the framework of the scientific project No. 18-08-00977 A “Creation of an Intelligent System for the Detection, Recognition and Understanding of Distorted Printed Texts in Images and Videos” and within the framework of the Competitiveness Improvement Program of the Tomsk Polytechnic University.

REFERENCES

1. Bundzel, M. & Hashimoto, S. (2010) Object identification in dynamic images based on the memory-prediction theory of brain function. *Journal of Intelligent Learning Systems and Applications*. 2(4). pp. 212–220. DOI: 10.4236/jilsa.2010.24024
2. Park, S. & Yoo, J.H. (2013) Realtime face recognition with SIFT based local feature points for mobile devices. *The 1st Int. Conf. on Artificial Intelligence, Modelling and Simulation (AIMS 13)*. Malaysia. pp. 304–308. DOI: 10.1109/AIMS.2013.56
3. Tawfiq, A. & Ahmed, J. (2016) Object detection and recognition by using enhanced speeded up robust feature. *International Journal of Computer Science and Network Security*. 16(4). pp. 66–71.
4. Tore, V. & Chawan, P.M. (2016) FAST Clustering based feature subset selection algorithm for high dimensional data. *International Journal of Computer Science and Mobile Computing*. 5(7). pp. 234–238. DOI: 10.1109/TKDE.2011.181

5. Mammeri, A., Boukerche, A. & Khiari, E. (2016) MSER based text detection and communication algorithm for autonomous vehicles. *IEEE Symposium of Computers and Communication*. Messina, Italy. pp. 456–460. DOI: 10.1109/ISCC.2016.7543902
6. Dalal, N. & Triggs, B. (2005) Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. San Diego, USA. Vol. 1. pp. 886–893. DOI: 10.1109/CVPR.2005.177
7. Mohey el-Din, D. (2016) Enhancement bag-of-words model for solving the challenges of sentiment analysis. *International Journal of Advanced Computer Science and Applications*. 7(1). pp. 244–251. DOI: 10.14569/IJACSA.2016.070134
8. Kecman, V. & Melki, G. (2016) Fast online algorithms for support vector machines. *IEEE South East Conference (SoutheastCon 2016)*. Virginia, USA. pp. 26–31. DOI: 10.1109/SECON.2016.7506733
9. Le Cun, Y. & Bengio, Y. (1998) Convolutional networks for images, speech and time series. In: Arbib, M.A. (ed.) *The Handbook of Brain Theory and Neural Networks*. Vol. 7(1). Bradford Book. pp. 255–258.
10. Tensorflow.org. (n.d.) *TensorFlow: The Python deep learning library*. [Online] Available from: https://www.tensorflow.org (Accessed: 22nd October 2019).
11. Microway.com. (n.d.) *Deep Learning Frameworks: A Survey of TensorFlow, Torch, Theano, Caffe, Neon, and the IBM Machine Learning Stack*. [Online] Available from: https://www.microway.com/hpc-tech-tips/deep-learning-frameworks-survey-tensorflow-torch-theano-caffe-neon-ibm-machine-learning-stack (Accessed: 22nd October 2019).
12. Keras.io. (n.d.) *Keras: The Python deep learning library*. [Online] Available from: https://keras.io (Accessed: 22nd October 2019).
13. CS.toronto.edu. (n.d.) *CIFAR-10 dataset of images*. [Online] Available from: https://www.cs.toronto.edu/~kriz/cifar.html (Accessed: 22nd October 2019).
14. Abadi, M. (2016) TensorFlow: A System for Large-Scale Machine Learning. *12th USENIX Symposium on Operating Systems Design and Implementation*. 33. pp. 25–33.
15. Duchi, J., Hazan, E., & Singer, Y. (2011) Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*. 12. pp. 2121–2159.
16. Kingma, D.P. (2015) Adam: a Method for Stochastic Optimization. In: Kingma, D.P. & Ba, J.L. (eds) *Proc. Int. Conf. on Learning Representations*. San Diego, USA. pp. 1–13.
17. ISPRS. (n.d.) *2D Semantic Labeling - Vaihingen dataset*. [Online] Available from: http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-vaihingen.html
18. Nguyen, V., Kim, H. & Jun, S. (2018) A Study on Real-Time Detection Method of Lane and Vehicle for Lane Change Assistant System Using Vision System on Highway. *Engineering Science and Technology*. pp. 822–833. DOI: 10.1016/j.jestch.2018.06.006
19. El-Khatib, S.A. (2015) Image segmentation using a mixed and exponential particle-based algorithm. *Computer Science and Cybernetics*. 1. pp. 126–133.
20. Koltsov, P.P. (2011) The use of metrics in a comparative study of the quality of work of image segmentation algorithms. *Informatics and Its Applications*. 5. pp. 53–634.

Received: November 18, 2019

Druki A.A., Spitsyn V.G., Arkalykov E.U. (2020) SEMANTIC SEGMENTATION ALGORITHMS OF THE EARTH'S SURFACE PICTURESBASED ON NEURAL NETWORK METHODS. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie vychislitelnaja tehnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 51. pp. 72–78

DOI: 10.17223/19988605/51/8

Друки А.А., Спицын В.Г., Аркалыков Е.У. АЛГОРИТМЫ СЕМАНТИЧЕСКОЙ СЕГМЕНТАЦИИ СНИМКОВ ЗЕМНОЙ ПОВЕРХНОСТИ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ. *Вестник Томского государственного университета. Управление, вычислительная техника и информатика.* 2020. № 51. С. 72–78

Целью работы является разработка алгоритмов, решающих задачу семантической сегментации изображений. Была разработана сверточная нейронная сеть с оригинальной архитектурой. Выполнена программная реализация алгоритма, позволяющая строить карту из сегментированных объектов. Представлено сравнение результатов, предложенного алгоритма, с существующими аналогами.

DRUKI Alexey Alexeevich (Candidate of Technical sciences, Associate professor, National Research Tomsk Polytechnic University, Tomsk, Russian Federation).
E-mail: druki@tpu.ru

SPITSYN Vladimir Grigorievich (Doctor of Technical Sciences, Professor, National Research Tomsk Polytechnic University, Tomsk, Russian Federation).
E-mail: spvg@tpu.ru

ARKALYKOV Erbolat Usenovich (National Research Tomsk Polytechnic University, Tomsk, Russian Federation).
E-mail: arkalykov@tpu.ru