

## Classification of Hyperspectral Images Using Conventional Neural Networks

V. I. Kozik<sup>1</sup> and E. S. Nezhevenko<sup>1\*</sup>

<sup>1</sup>*Institute of Automation and Electrometry, Siberian Branch, Russian Academy of Sciences,  
Novosibirsk, 630090 Russia*

Received October 12, 2020; revised January 28, 2021; accepted February 4, 2021

**Abstract**—We show that for the classification of fragments of a hyperspectral image, it is very effective to first transform its spectral features into principal components and then to recognize it using a convolutional neural network trained on a sample composed of fragments of this image. High percentage of correct classification was obtained when working with a large-format hyperspectral image while some of the classes of the hyperspectral image are very close to each other and, accordingly, are difficult to distinguish by hyperspectra. We investigate the dependence of the correct classification on the change in the size of the fragments from which the training and validation samples are composed and on the parameters of the convolutional neural network.

**DOI:** 10.3103/S8756699021020102

*Keywords:* classification, hyperspectral imaging, principal components, convolutional neural networks, learning

### INTRODUCTION

In recent years, the problem of classification of terrain images by their hyperspectral measurements has become more popular. This problem has been investigated in some detail in works [1–4]. Most importantly, it was shown that a significant increase in the percentage of correct classification of hyperspectral images (HSIs) takes into account not only the spectral characteristics, but also the spatial structure of the HSI. Note that the overwhelming majority of works in this area use classic classification algorithms. However, at the moment it can be considered proven that the best (if not unique) results in the field of image recognition, including classification, were obtained by using convolutional neural networks and deep learning. Thus, applying these approaches to HSIs is more than justified. Work in this direction has already been carried out [5–7]. The aim of [7] is a thorough study of the dependence of the HSI classification accuracy on various parameters of convolutional networks used for classification. We shall use the methods proposed in this work with caution: some of them seem not entirely justified to us.

### CHARACTERISTICS OF THE CLASSIFIED OBJECT

The investigated hyperspectral image is a terrain area obtained within the AVIRIS (Airborne Visible Infrared Imaging Spectrometer) program at the Indian Pine test site (Indiana, USA). Image size is  $614 \times 1408$  pixels, the definition is 20 m/pixel, and the number of channels is 220 in the range of 0.4–2.5  $\mu\text{m}$ . The RGB representation of the HSI is shown in Fig. 1.

Figure 2 presents the splitting of this HSI into classes in pseudocolors. There are 57 classes in total. However, the specific nature of the spatial processing method we have chosen is such that in some areas classification objects cannot be formed due to the small size of the areas. Therefore, the names of the classes will be given after training the network (when selecting the existing classes).

---

\*E-mail: [nejevenko@iae.nsk.su](mailto:nejevenko@iae.nsk.su)



Fig. 1.

## STRUCTURE OF THE CONVOLUTIONAL NETWORK

We are not going to describe the principles of operation of convolutional networks, since they are well enough discussed in literature. Let us present the scheme of a convolutional network (Fig. 3) and consider its features.

Let us use a three-dimensional convolutional neural network. The input layer is a three-dimensional cube of size  $M \times N \times F$ ;  $M \times N$  is the fragment size of a region belonging to the same class describing the spatial characteristics of the region and  $F$  is the number of features representing the spectral characteristics of the area. The fragment size  $M \times N$  is of utmost importance. Too small size of a fragment would not reveal its spatial features. In the case of large fragments, their number in the class decreases, since the areas belonging to the classes have an arbitrary shape, and the fragments are rectangular, so some classes might end up not containing any fragments. Let us now discuss the third dimension  $F$ . The work [7] states that in this dimension it is most efficient to use all the spectral components without transformation, since the latter is used only to reduce the computational procedures by reducing the layers. In reality, this is not entirely true. The spectral components of the third layer are highly correlated. And we know from recognition theory that the use of correlated features reduces the correctness of recognition, therefore, for effective recognition features are usually decorrelated. Therefore, the spectral information was pre-processed by transforming it into principal components. The number of principal components and, accordingly, the number of layers in the input plane are determined, for example, by calculating the scree plot, i.e., the graph of decreasing eigenvalues. Figure 4 presents the scree plot for the considered HSI.

The  $x$  axis is the numbers of the principal components and the  $y$  axis is their normalized eigenvalues. Since the eigenvalues decrease very quickly, the graph depicts the first 10 numerical eigenvalues. The graph shows that the fifth eigenvalue is already  $1/500$  of the first value, which means that it accounts for 0.2% of the variance of the spectral components, therefore, most of the experiments are carried out



Fig. 2.

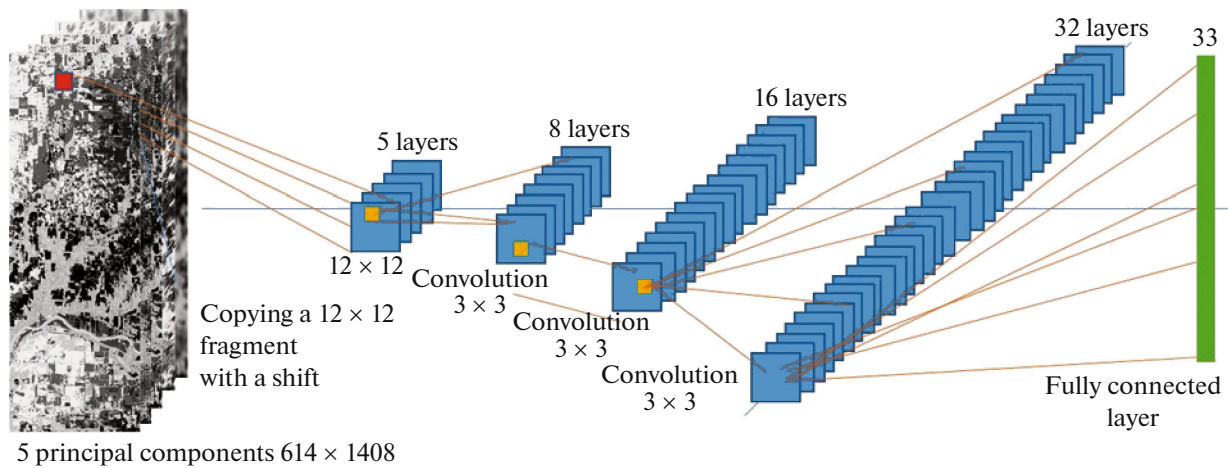


Fig. 3.

with the number of principal components equal to 5. The presented convolutional neural network has five layers in total, the kernel size is  $3 \times 3$ . Subsampling is not used in our network, because the images to be classified are already small enough. The dimension of the output layer is equal to the number of classes identified on the HSI, taking into account the size of the fragments.

The most important step when using convolutional neural networks and deep learning for classification is to create a training set. In our case, the objects of this set are HSI fragments.

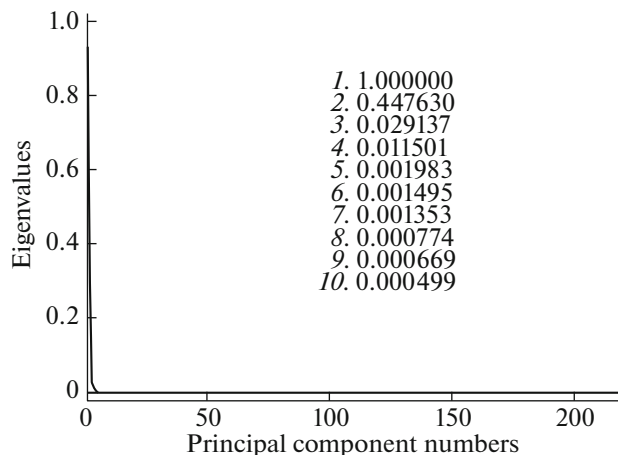


Fig. 4.

### RESULTS OF EXPERIMENTS ON HSI CLASSIFICATION

Let us list all the stages of the HSI classification (note that the classification was carried out in MATLAB except for finding the principal components, which were calculated using ENVI):

1. The principal components of the HSI are calculated .
2. Class directories from 1 to 57 are formed.
3. Fragments all elements of which belong to the same class are selected from the file containing the HSI markup into classes (see. Fig. 2) using a floating window of size  $M \times N$  and shifts  $\text{shift}_M$  and  $\text{shift}_N$ .
4. A window all elements of which belong to the same class is identified as an object belonging to this class, and its coordinates are determined on the image. Using these coordinates, a fragment of size  $M \times N \times F$  is taken from the file containing the selected principal components and written to the corresponding class directory. Files are registered in each directory according to the number of found fragments of this class. Based on the results of forming the directories, the number of classes is determined and the training function is corrected.
5. The network parameters are adjusted: the number of layers, the kernel size, and the number of feature maps.
6. The parameters of the training procedure are adjusted: the number of classes and the number of training epochs; objects of each nonempty class are divided into training and validation sets.
7. The training procedure starts.

Let us proceed to the experimental results. Note that we consider the classification accuracy as the only criterion for the effectiveness of a particular procedure—generation of fragments, training, classification—defined as the ratio of the number of correctly classified objects to the total number of objects (the term “accuracy” is used along with the term “probability of correct classification”).

We use cross-validation (hold-out validation) to assess the classification accuracy in the formation of training and validation (test) sets [8].

The set is randomly divided into training and validation ones in a 7 : 3 ratio and they do not overlap. The hold-out method is used for large datasets, which fits our case (the total number of objects is 34 596 and there are at least 50 objects in each class).

Let us consider how the classification accuracy will change with a change in some parameters of the convolutional network (the size of the fragment that determines the dimension of the input layer, the number of layers of the neural network, the number of training epochs, and the number of principal components). Fragments are square and, to ensure the maximum number of fragments, the shifts are  $\text{shift}_M = \text{shift}_N = 1$ . Figure 5 presents the classification accuracy depending on the fragment size for five network layers and 50 training epochs and for five and ten principal components.

Interestingly, at ten principal components the classification accuracy is much less dependent on the fragment size. An important parameter of any neural networks, including convolutional networks, is the

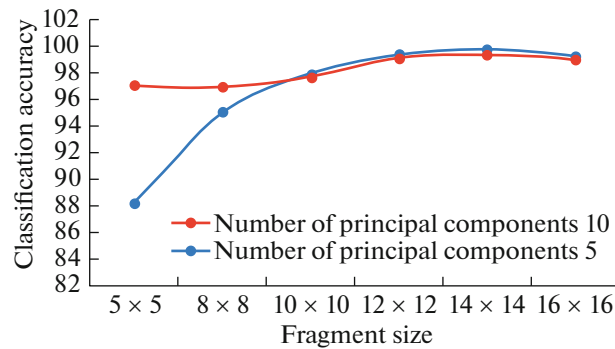


Fig. 5.

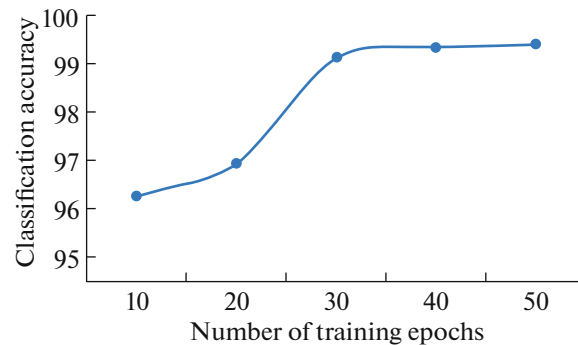


Fig. 6.

initial learning rate. In our case it is 0.01. The rate was kept constant, because fast learning was not an objective. Another important learning parameter is the number of training epochs, which determines not only the rate, but also the final classification accuracy. Figure 6 shows the dependence of the final classification accuracy on the number of training epochs at  $12 \times 12$  fragment size and five principal components. It can be seen that the classification accuracy monotonically increases with the number of epochs, and there is a sharp increase from 20 to 30 epochs. However, this dependence is largely determined by the fragment size. Figure 7 (top) shows changing classification accuracy during training for fragments of  $12 \times 12$  elements in size, and Fig. 8 shows that for a  $5 \times 5$  fragment. For  $12 \times 12$  fragments at 30 epochs, the classification accuracy virtually saturates while for  $5 \times 5$  fragments the accuracy continues to grow at 50 epochs.

Figure 9 shows the dependence of the classification accuracy on the number of layers of a neural network. The optimal number of layers is five.

The parameter of the most successful classification experiment is the fragment size  $M \times N = 12 \times 12$ . At the same time, it should be noted that the larger the fragments, the fewer of them are in the class and the lesser the number of classes themselves. Table 1 shows class names, number of fragments in a class, and classification accuracy for two fragments of sizes  $5 \times 5$  and  $12 \times 12$  with shifts  $\text{shift}_M = \text{shift}_N = 1$ , five layers and, 50 epochs. The results are: 45 classes were obtained for a  $5 \times 5$  fragment, and 33 classes were obtained for a  $12 \times 12$  fragment. Names of the classes suggest that we did not combine closely related classes (for example, crops of corn or crops of soybeans) into one, as in [4]. It is clear that it is much easier to distinguish corn crops from forest than to distinguish between different crops of the same corn or soybeans. The differences between close classes are shown in [9], so we compare the results of this work with them. Note that for  $12 \times 12$  fragments almost all objects hard to distinguish (crops of corn and soybeans) are classified with a very high (often 100%) probability. It should also be noted that the results obtained in this work significantly surpass the results of [9] with a restriction: the latter does not have the problem of covering an area belonging to the class with rectangular windows; therefore, it can classify areas with a complex configuration. As for comparison with [5–7], the presented work has significantly more classes, including those difficult to distinguish.

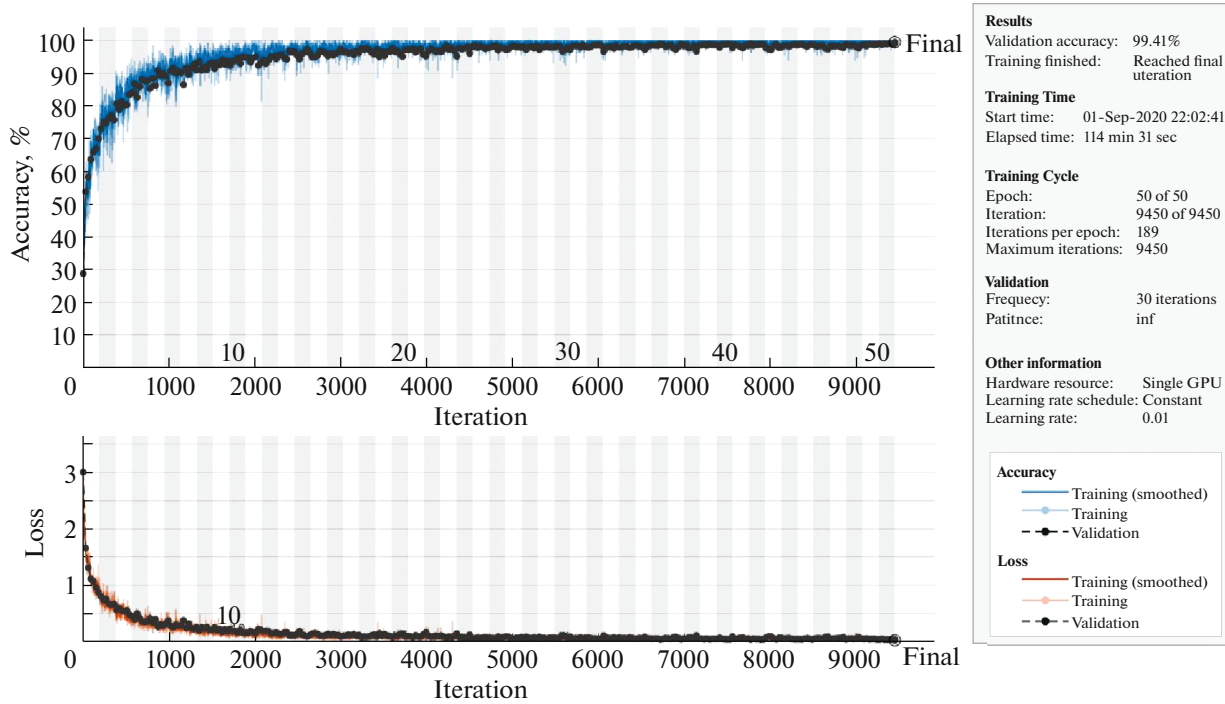


Fig. 7.

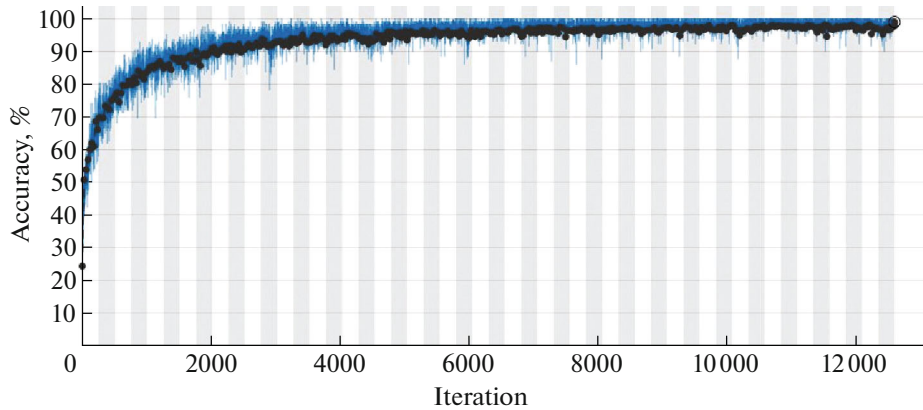


Fig. 8.

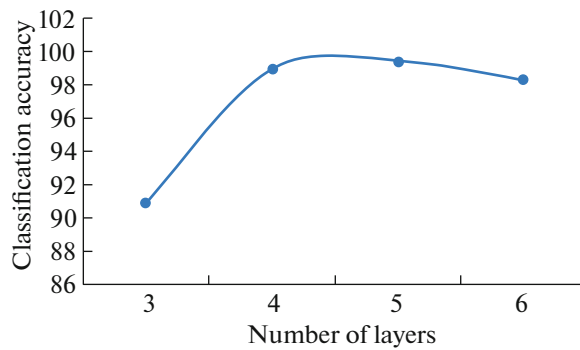


Fig. 9.

**Table 1**

Class number	Class name	Fragment size 5 × 5		Fragment size 12 × 12	
		Number of fragments	Classification accuracy	Number of fragments	Classification accuracy
1	Bare soil	168	0.7200	0	—
2	Buildings	5541	0.8454	2621	0.9987
3	Corn	6005	0.8440	2269	0.9927
4	Corn, west-east	60	1.0000	0	—
5	Corn, north-south	648	0.8247	169	1.0000
6	Corn, conventional tillage	2541	0.7507	368	1.0000
7	Corn, conventional tillage, west-east	8571	0.7841	2481	0.9970
8	Corn, conventional tillage, north-south	12803	0.8578	4241	0.9914
9	Corn, conventional tillage, north-south, irrigated	116	0.8286	0	—
10	Corn, conventional tillage — ?	307	0.4783	45	1.0000
11	Corn, low-destructive tillage	96	0.7586	0	—
12	Corn, low-destructive tillage, west-east	2006	0.8605	896	0.9963
13	Corn, low-destructive tillage, north-south	3255	0.9037	1099	0.9970
14	Corn without tillage	879	0.8598	93	1.0000
15	Grain without tillage, west-east	200	0.7833	30	1.0000
16	Corn without tillage, north-south	2705	0.8633	1304	1.000
17	Grass	32	0.8000	0	—
18	Grass/Trees	561	0.9881	91	0.9630
19	Hay	48	0.7857	0	—
20	Hay?	964	0.9343	443	0.9925
21	Hay-alfalfa	628	0.9894	191	1.0000
22	Not cropped	180	0.9630	0	—
23	Oats	324	1.0000	70	1.0000
24	Pasture	3377	0.9704	1996	1.0000
25	Soybeans	1324	0.8363	326	1.0000
26	Soybeans?	152	0.9130	0	—
27	Soybeans, north-south	72	0.9091	0	—
28	Soybeans, conventional tillage	957	0.7526	227	1.0000
29	Soybeans, conventional tillage?	792	0.6933	239	0.9722
30	Soybeans, conventional tillage, west-east	4715	0.8112	2000	0.9900
31	Soybeans, conventional tillage, north-south	3830	0.6762	1057	0.9905
32	Soybeans, conventional tillage, furrows	384	0.8174	40	0.6667
33	Soybeans, conventional tillage, weeds	116	0.8000	0	—
34	Soybeans planted in rows	4680	0.8832	1046	0.9777



**Table 1.** Continuation

Class number	Class name	Fragment size $5 \times 5$		Fragment size $12 \times 12$	
		Number of fragments	Classification accuracy	Number of fragments	Classification accuracy
35	Soybeans, low-destructive tillage	424	0.9449	50	1.0000
36	Soybeans, low-destructive tillage, west-east	512	0.9221	65	1.0000
37	Soybeans, low-destructive tillage, ridge	2507	0.9109	689	1.0000
38	Soybeans, low-destructive tillage, north-south	2212	0.6717	721	1.0000
39	Soybeans, west-east	673	0.9356	185	1.0000
40	Soybeans without tillage, west-east	1054	0.9747	356	1.0000
41	Soybeans without tillage, north-south	180	0.7222	0	–
42	Soybeans without tillage planted in rows	2324	0.9813	436	1.0000
43	Trees?	48	1.0000	0	–
44	Wheat	1664	0.9880	636	1.0000
45	Forest	20324	0.9405	8115	0.9988

High classification accuracy raises doubts that the neural network is being overtrained. According to [10], overtraining or overfitting is an undesirable phenomenon that occurs in problems of instance-based learning, when the probability of an error of the trained algorithm on the objects of the test set turns out to be significantly higher than the average error on the training set. From Fig. 7 (bottom) that describes the behavior of an error during training, it follows that the error on the test set exceeds the error on the training set insignificantly (by fraction of a percent); therefore, there is no overtraining in this case and no measures should be taken to eliminate it.

## CONCLUSIONS

Thus, the proposed work experimentally shows that in the classification of hyperspectral images, the transformation to the spectral principal components and further spatial transformation by dividing the principal components into small fragments, training the convolutional neural network on parts of these fragments and classification of HSI using that network provides a very high percentage of correct classification (99.43%). Additionally, the number of classes is quite large (33) and among them there are very close classes (8 classes of corn crops and 13 classes of soybeans crops). As the fragment size decreases, the classification accuracy decreases somewhat, but the number of recognized classes increases. For a  $5 \times 5$  fragment the classification accuracy for the number of principal components of 5 is about 88%, and for 10 principal components, 97%; the number of recognized classes in this case is 45. The influence of changing the parameters of the convolutional network and the number of principal components on the classification accuracy has been investigated.

It should be noted that such a high classification accuracy is largely due to the way the training and validation sets are formed, characterized by their very close mixing. At the same time, it is clear that this classification method can also be applied in some cases. In particular, this method of formation (random division into training and validation sets) works well when classes occupy both large and small areas. Our research, not included in this publication, shows that in the case of small areas, a high classification accuracy is also obtained with a spatial separation of the training and validation sets.

## FUNDING

The research was supported by Ministry of Science and Higher Education of the Russian Federation (project no. AAAA-A17-117052410034-6).



## REFERENCES

1. M. Borhani and H. Ghassemian, "Hyperspectral image classification based on non-uniform spatialspectral kernels," in *Proc. of the Iranian Conf. on Intelligent Systems, Bam, Iran, 2014*  
<https://doi.org/10.1109/IranianCIS.2014.6802579>.
2. S. M. Borzov and O. I. Potaturkin, "Efficiency of the spectral-spatial classification of hyperspectral imaging data," *Optoelectron., Instrum. Data Process.* **53**, 26–34 (2017).  
<https://doi.org/10.3103/S8756699017010058>
3. S. M. Borzov and O. I. Potaturkin, "Classification of hyperspectral images with different methods of training set formation," *Optoelectron., Instrum. Data Process.* **54**, 76–82 (2018).  
<https://doi.org/10.3103/S8756699018010120>
4. S. M. Borzov and O. I. Potaturkin, "Spectral-spatial methods for hyperspectral image classification. Review," *Optoelectron., Instrum. Data Process.* **54**, 582–599 (2018). <https://doi.org/10.3103/S8756699018060079>
5. B. Fang, Y. Li, H. Zhang, and J. Ch.-W. Chan, "Hyperspectral images classification based on dense convolutional networks with spectral-wise attention mechanism," *Remote Sens.* **11**, 159 (2019).  
<https://doi.org/10.3390/rs11020159>
6. N. Audebert, B. Saux, and S. Lefèvre, "Deep learning for classification of hyperspectral data: A comparative review," *IEEE Geosci. Remote Sens. Mag.* **7** (2), 159–173 (2019).  
<https://doi.org/10.1109/MGRS.2019.2912563>
7. Y. Li, H. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.* **9**, 67 (2017).  
<https://doi.org/10.3390/rs9010067>
8. Cross-validation. ITMO University. <http://neerc.ifmo.ru/wiki/index.php?title=Cross-validation> Cited October 12, 2020.
9. E. S. Nezhevenko, "Neural network classification of difficult-to-distinguish types of vegetation on the basis of hyperspectral features," *Optoelectron., Instrum. Data Process.* **55**, 263–270 (2019).  
<https://doi.org/10.3103/S8756699019030087>
10. Overtraining. <http://www.machinelearning.ru/wiki/index.php?title=Retraining>. Cited October 12, 2020.

*Translated by L. Trubitsyna*