Building Detection on Aerial Images Using U-NET Neural Networks

Leonid Ivanovsky, Vladimir Khryashchev, Vladimir Pavlov,
P.G. Demidov Yaroslavl State University
Yaroslavl, Russian Federation
leon.ivanovsky@yahoo.com, vhr@yandex.ru, i@yajon.ru

Anna Ostrovskaya
People's Friendship University of Russia (RUDN University)
Moscow, Russian Federation,
ostrovskaya_aa@rudn.university

***Abstract*— this article presents research results of two convolutional neural networks for building detection on satellite images of Sovzond database. To analyze the quality of developed algorithms, there was used Sorensen-Dice coefficient of similarity which compares results of algorithms with tagged masks. The masks were generated from json files and sliced on smaller parts together with respective images before the training of algorithms. This approach allows to cope with the problem of segmentation for aerial high-resolution images efficiently and effectively. The problem of building detection on satellite images can be put into practice for urban planning, building control, etc.**

I. INTRODUCTION

Today, the problem of object detection on high-resolution satellite images is in the focus of researchers. Automatic image segmentation allows to extract areas of interest on aerial photos. Most approaches of solving this problem suggest the development of deep learning algorithms. However, for satellite image segmentation the usage of these algorithms instead of traditional solutions in the field of machine learning is nontrivial for some reasons [1]. Such methods should

- Take into account the small size of objects.
- On large aerial photos objects are tiny and grouped. The distance to the ground determines what is captured on one pixel of an image. This means that objects such as buildings take only few pixels on satellite images.
- Be invariant to rotation.
- Objects on aerial photos are located in different ways. For instance, vehicles or buildings can be rotated on any angle.
- Have enough training examples.
- For most available datasets, such as the Inria database [2], there is a shortage of tagged images. Nevertheless, the methods of data augmentation, such as a mirror flip, rotation or affine transformation allow to solve this problem.
- Have an ability to handle huge pictures.
- Satellite images are very large. Sometimes their size is more than 150 Mb and their resolution exceeds 16000 × 16000 pixels. For example, images from DigitalGlobe satellite [3] cover more than 64 km² and contain at least 250 million pixels.
- Cope with noise.
- For the problem of object detection on aerial photos there were collected special datasets. They contain satellite images which were shooted in fair weather. However, it possible that in some photos there is noise, such as little clouds in the sky or flare spots from roofs. Obviously, it is supposed that modern developed deep learning algorithms are noise resistant.

In machine learning the problem of image segmentation is usually reformulated as a classification on a pixel-wise level. The simplest and the slowest way of solving this problem is a manual segmentation of images by experts. However, it is a time-consuming process, which is subjected to human errors due to monotonous manual work. In this field of machine learning the great interest aims at automatic image segmentation. It makes possible to provide real-time image processing immediately after receiving it. Satellite image segmentation finds its application in urban planning, building control, forest management, meteorology and land-use agriculture.

This article presents developed convolutional neural networks (CNNs). The structure of these models is parallel and fit to the architecture of graphics processing units (GPUs) which consists of thousands of cores to perform several tasks simultaneously. Although CNNs have been known for decades, only recent achievements in the development of high-performance computers with GPUs have allowed researchers to launch CNNs, which have millions of parameters. Currently, in tasks of computer vision CNNs excel traditional machine learning algorithms and even some experts in a speed and a quality [1].

This article consists of six parts. The first part introduces readers to the problem of satellite image segmentation and also describes the preference of usage of CNNs together with GPU technology in comparison with classical machine learning algorithms. The second section contains an overview of papers related to the task of image segmentation. The third part is devoted to the available databases of satellite images. The forth section describes the developed architectures of CNNs for building detection on aerial photos and some peculiarities of training models. In addition in this part there is mentioned about Keras framework for the designing of deep machine learning algorithms. The fifth part presents the results of numerical experiments for the developed models. And finally, in the conclusion there is summarized the research.

II. RELATED WORKS

The problem of satellite images segmentation is challenging. The most approaches of solving this problem suppose the usage of deep learning algorithms. The features in these networks are formed automatically in the process of

training. In recent years there were developed some CNNs, which aim at satellite image segmentation.

One of the most successful algorithms is based on fully convolutional networks (FCNs). The basic idea of FCNs is the usage of fully connected layer with a convolution layer at end, while other layers extract necessary features from input data. It allows to launch this type of network for image segmentation [4].

FCNs have been improved and now they are known as Feature Pyramid neural networks (FPNs). FPNs uses pyramid architecture of CNNs to construct complicated features at all scales in a bottom-up pathway and extract needed ones after passing a top-down part of algorithm. This architecture showed significant improvement in several applications in particular for object detection on satellite images [5]. The usage of FPNs allows to get the value of Jaccard index is approximately equal to 0.49 for satellite images from DeepGlobe [6].

The method of using FCN was complicated to U-Net. For the first time this deep learning algorithm was presented in paper [7] for segmentation of biomedical images. Later this model was applied to pixel-wise classification of satellite images [8]. U-Net is a special type of FCNs, which merges low-level and higher-level feature maps for the better localization of objects. Using U-Net, the authors of paper [8] get the value of Sorensen-Dice coefficient is equal to 0.75 for building detection on satellite images.

In paper [9] authors present LinkNet. This is special architecture of CNNs, which consist of an encoder and a decoder as U-Net. It efficiently share the information learnt by the encoder with the decoder after each downsampling block. In some cases this approach is better than FCNs in a decoder. This technique of feature forwarding allows to get high accuracy values for object detection on the CamVid dataset [10].

III. DATABASES OF SATELLITE IMAGES

A dataset of images is the important part for training and estimation of quality for different machine learning algorithms. Now there exist some available databases of aerial photos.

The Inria database [2] contains 180 color satellite images, which cover a total area of 810 km². Every image of Inria dataset has a resolution of 1000 × 1000 pixels with a spatial resolution of 0.3 m / pixel. All pictures are divided into 2 classes: “buildings” and “not buildings”. Images of Inria database cover various cities, from megapolises to small towns: San Francisco (USA), Chicago (USA), Vienna (Austria), Innsburk (Austria), Bellingham (USA) and Tyrol (Austria). Examples of images from the Inria database are shown in Fig. 1.

The SpaceNet database [11] includes satellite images of 6 large urban agglomerations: Rio de Janeiro (Brazil), Las Vegas (USA), Paris (France), Shanghai (China), Khartoum (Sudan) and Atalanta (USA). Eight-channel photos are shooted by WorldView-2 and WorldView-3 satellites with a different spatial resolution. The database is divided into subsets, depending on the type of tagged objects. For instance, the SpaceNet database contains two subsets of satellite images, which cover areas of 3011 km² and 5555 km², for building detection. Examples of images from these subsets are shown in Fig. 2.

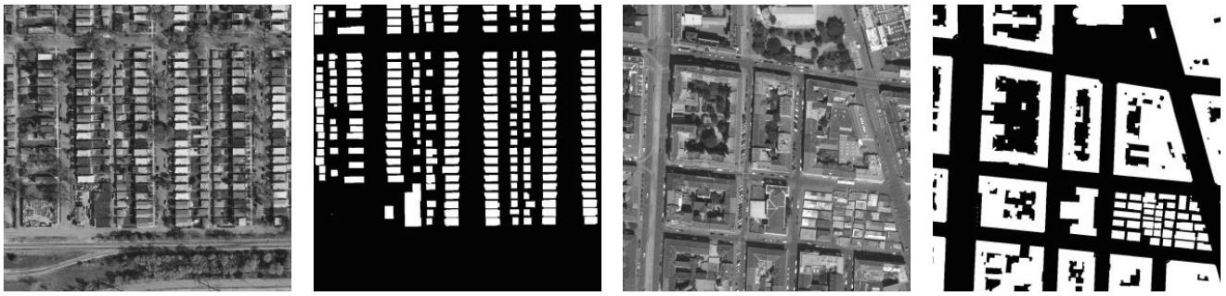


Fig. 1. Examples of images from the Inria database

Fig. 2. Examples of images from the SpaceNet database

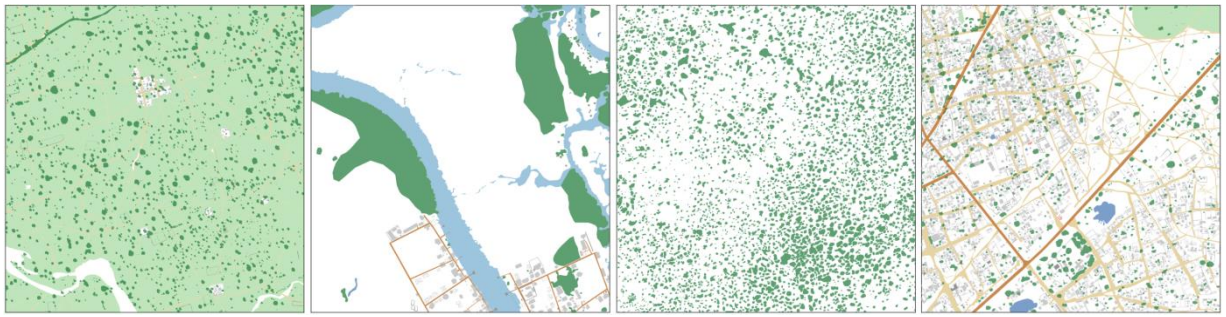


Fig. 3. Examples of images from the DSTL database

Fig. 4. Examples of images from the Sovzond database

The DSTL database contains 50 satellite images of 3300 × 3300 pixels in GEOTIFF formats. For the first time, this database was provided in Kaggle competition "DSTL Satellite Imagery Feature Detection" [12]. Images of the DSTL dataset cover the area of 1km × 1km. They are labeled on 10 different classes: "buildings", "manmade structures", "roads", "tracks", "trees", "crops", "waterway", "standing water", "large vehicles" (e.g. lorries, trucks or buses) and "small vehicles" (cars, vans or bikes). The examples of images from the DSTL database are shown on Fig. 3.

In our research to make a comparative analysis of developed algorithms for building detection, there were used 14 color aerial photos from the database provided by Sovzond [13]. Every image of the Sovzond dataset has a resolution of 16384 × 16384 pixels with a spatial resolution of 0.5 m / pixel. Images of Sovzond database cover the area of Yaroslavl (Russia) and Moscow (Russia) and their suburbs. Examples of images from the Sovzond database are shown in Fig. 4.

IV. DEEP LEARNING ALGORITHMS

This article presents developed CNNs. This type of neural networks has a special architecture, aimed at fast and high-quality detection and classification of various objects [1]. CNNs are related to algorithms of deep machine learning, which are popular now to solve most modern problems of computer vision, in particular satellite image segmentation. In this research there are created two models: U-Net [8] и LinkNet [9]. The research of work of developed models continues the research, which was provided in paper [13].

All developed networks were created using Keras library with Tensorflow framework as a backend. Keras is an open source library written in Python. This library contains many implementations of structural blocks of neural networks such as layers, activation functions, optimizers, and ready tools for preprocessing of images and text data [14]. Furthermore, this library allows to train and test networks on GPU.

As shown in Fig. 5, U-Net network consists of two parts: an encoder (left) and a decoder (right). The encoder is a neural network which has a typical CNN architecture including four blocks. Each of these blocks consists of two convolutional layers with 3 × 3 filter, ReLU activation function applied to each of them and a maxpooling operation with 2 × 2 filter and step 2. The decoder contains the same number of blocks. Each decoder block consists of an upsampling operation with 2 × 2 filter combining with a corresponding map of features from the encoder, two convolutional layers with 3 × 3 filter and ReLU activation function applied to each of them. The last layer of the network performs a convolution operation with 1 × 1 filter, which relates every pixel to a specific class. As a result, the network has 19 convolutional layers, 18 ReLU activation functions, 4 maxpooling operations, 4 upsampling operations, and 4 merge operations.

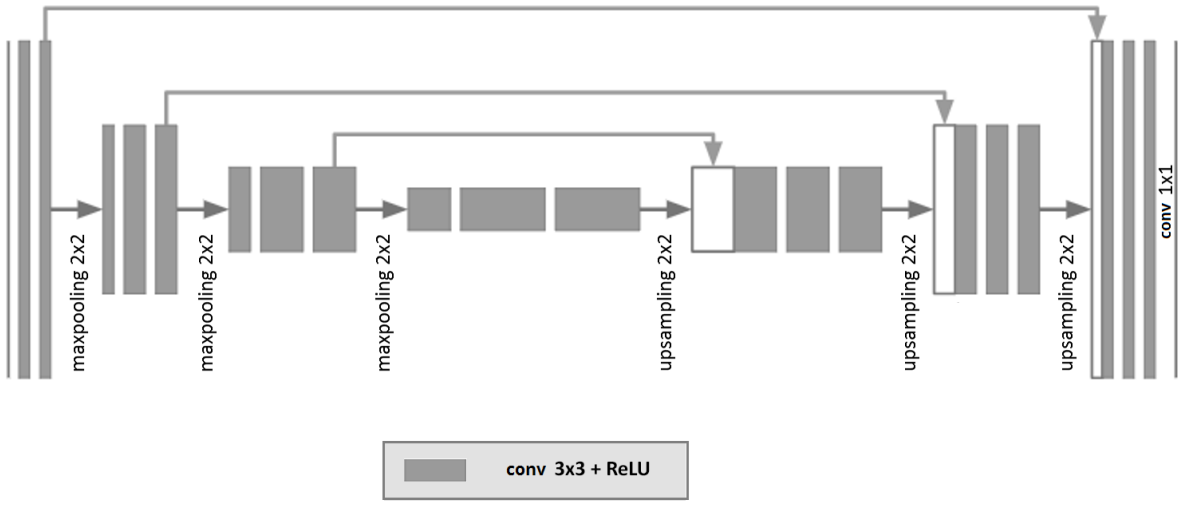


Fig. 5. U-Net architecture

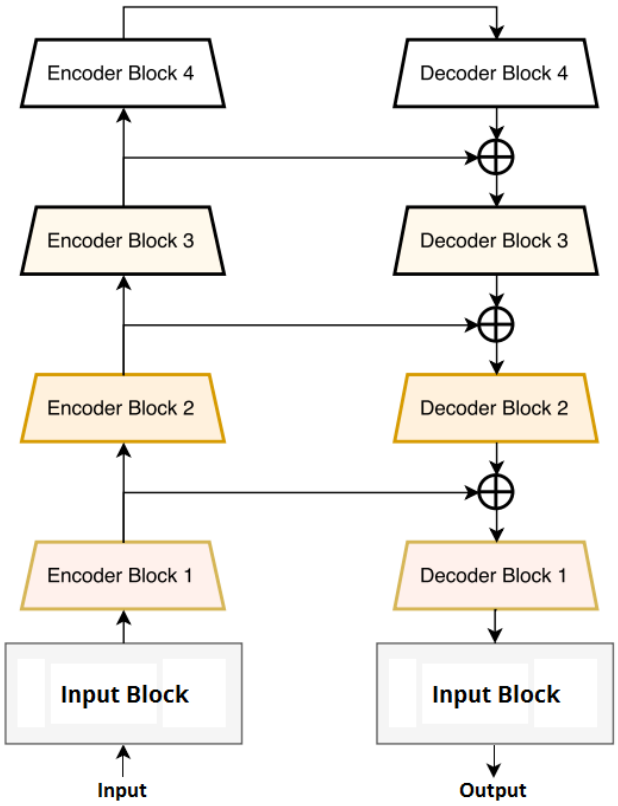
As in the case of U-Net architecture LinkNet has the encoder and the decoder. According to the network architecture, which is shown in Fig. 6, both subnets consist of 4 blocks. Each encoder block contains 4 convolutional layers, 2 merging operations and 1 maxpooling operation with 2 × 2 filter and step 2. In accordance with the structure of the encoder block, the decoder block has a similar architecture, except merging operations and a maxpooling operation, which was replaced with an upsampling operation with 2 × 2 filter. Moreover, before sending the map of features from an input data to the first encoder, there are sequentially implemented 2 operations of batch normalization, ReLU activation function, a convolution with 2 × 2 filter and a maxpooling operation with 2 × 2 filter. After the last decoder block was performed, a sequence of an upsampling operation with 2 × 2 filter, 2 operations of batch normalization, ReLU activation function and a convolutional layer with 2 × 2 filter. The LinkNet architecture and the encoder scheme of this network are shown in Fig. 5.

According to Table 1, the approach based on the usage of developed convolutional neural networks requires considerable computational resources. Therefore, training and test stages were implemented on a large number of independent streams of GPU using the parallel computing technology NVIDIA CUDA. This cross-platform technology is supported by all modern NVIDIA graphics cards [15].

TABLE 1. THE NUMBER OF TRAINING PARAMETERS IN DEVELOPED MODELS

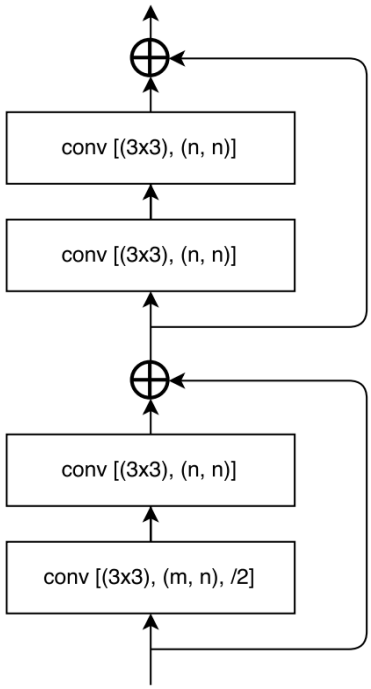
CNN	Approximate number of training parameters (mil)
U-Net	7.8
LinkNet	17,2

As the algorithm of numerical optimization Adam optimizer with a learning rate of 1e-3 was chosen. Binary cross-entropy was chosen as the loss function. On each training iteration, the model updated its weights after running through the network of a batch of 18 samples. The training finishes after completing 96 epochs.



a)

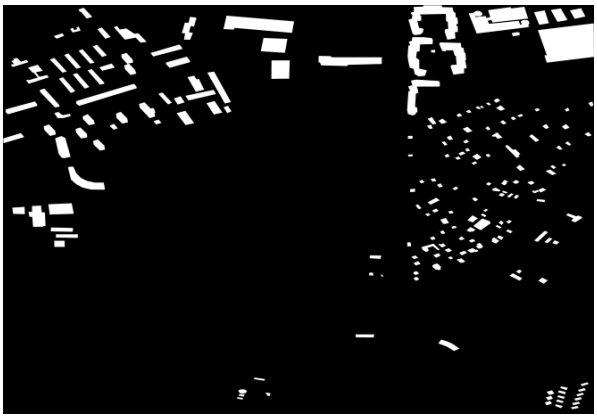
Fig. 6. LinkNet: a) architecture



b)

Fig. 6. LinkNet: b) endcoder scheme

a)



b)

Fig. 7. Sovzond data: a) images of dataset, b) generated masks

V. NUMERICAL RESULTS

Numerical experiments for developed algorithms were performed on satellite images of the Sovzond database. Information about the location of buildings was extracted from json files and generated as black-and-white masks, where a pixel is colored to white if it belongs to the building. Examples of images from the Sovzond database and generated masks are shown in Fig. 7.

The traditional approach for segmentation concerns the usage of parts of satellite images, which are fed to the input of CNN. Proposed detectors require images of 512 × 512 pixels, so before the launch of deep learning algorithms each image and mask of dataset have been cut on parts of appropriate size by data windowing. Examples of sliced samples are shown in Fig. 8. Every little part of sliced images corresponded to the needed little part of big generated mask.

As a result of such transformations, prepared dataset contained 3264 images of 512 × 512 size. For the modeling of numerical experiments the prepared dataset was shuffled and splitted on training and test set in ratio 80/20. Thus, the training set contained 2611 photos and the test set contained 653 photos. Train and test samples did not have same pictures. In our task, only 2 classes were taken into account: “buildings” and “not-buildings”.

Developed CNNs were launched on NVIDIA DGX-1 supercomputer, which was provided by Artificial Intelligence Center of P.G Demidov Yaroslavl State University.

As a result of numerical experiments, accuracy (A) of models was calculated with the following formula:

$$A = \frac{P}{N},$$

where P is a quantity of right classified objects and N is the count of objects for classification [16]. The results of numerical experiments on test set cite in Table 2.

TABLE 2. TESTING RESULTS OF CONVOLUTIONAL NEURAL NETWORKS

Model	**Accuracy (A)**
U-Net	96,31%
LinkNet	95,85%

The quality of the segmentation algorithms is usually evaluated by special metrics. To evaluate the quality of developed models, Sorensen-Dice coefficient (DSC) was used, which compares expert markup with predicted masks. This indicator takes values from the interval [0, 1] and shows the degree of similarity between two sets. The Sorensen coefficient is calculated with the following formula:

$$DSC = \frac{2I}{S},$$

where $I = |X \cap Y|$ is a power of intersection and $S = |X| + |Y|$ is a sum of powers for expert markup X and predicted masks Y [17].

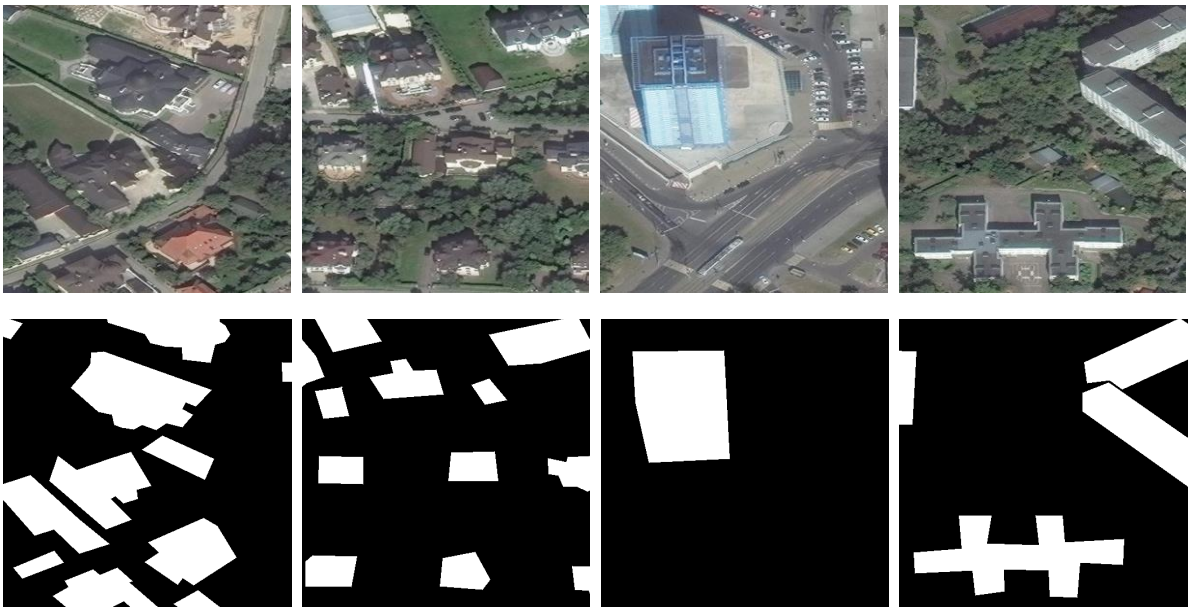


Fig. 8. Sliced samples: the first row – prepared images, the second row – masks.

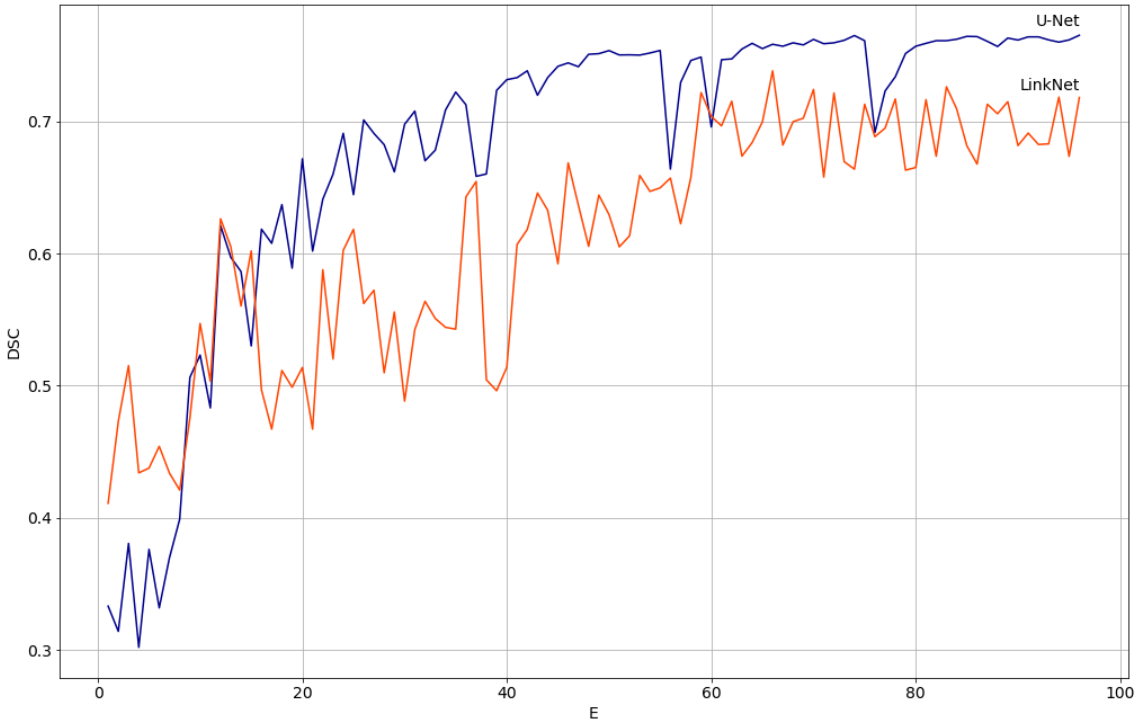


Fig.9. Dependencies of Dice coefficient on training epochs for developed convolutional neural networks

In our task, numerator I and denominator S can be calculated by following formulae

$$I = \sum_{\substack{x \in X \\ y \in Y}} xy, \qquad S = \sum_{\substack{x \in X \\ y \in Y}} (x + y),$$

where x, y are values of pixels from $[0, 1]$ for real mask X and predictions Y respectively. Dependencies of DSC values from the number of epochs (E) for each developed algorithm on the test set are shown in Fig. 9.

TABLE 3. TESTING RESULTS OF CONVOLUTIONAL NEURAL NETWORKS

Model	**Sorensen-Dice coefficient (DSC)**
U-Net	0,77
LinkNet	0,72

According to the results presented in Fig. 8 and Table 3, the worst results of satellite image segmentation were shown by LinkNet, while the best results were obtained using U-Net. This fact can be explained by the complexity of architectures of developed networks. However, it should be noticed, that the training of LinkNet lasted only 1.5 hours that is less than the time of learning of U-Net on about 1 hour in spite of the fact, that LinkNet has more parameters. Both architectures throw features from encoder to decoder, which allows models use more useful information from input data.

VI. CONCLUSION

The article shows CNNs which are implemented on GPU can be effectively used for building detection on aerial photos. Numerical experiments of evaluation for developed algorithms were performed for satellite images of Sovzond database. For an implementing of experiments there were extracted smaller parts from the images of dataset and corresponding masks which were generated from json files. Using the special metrics of similarity between expert markup and predicted masks there was shown that U-Net got better results compared with LinkNet. For U-Net the value of Sorensen-Dice coefficient (DSC) is equal to 0.77. Both created networks are simple for its implementation. For learning of CNNs there was used supercomputer NVIDIA DGX-1 of Artificial Intelligence Center of P.G Demidov Yaroslavl State University.

ACKNOWLEDGMENT

The article was prepared with the financial support of the Ministry of Education of the Russian Federation as part of the research project No. 14.575.21.0167 connected with the implementation of applied scientific research on the following topic: «Development of applied solutions for processing and integration of large volumes of diverse operational, retrospective and the thematic data of Earth's remote sensing in the unified geospace using smart digital technologies and artificial intelligence» (identifier RFMEFI57517X0167). The authors are also grateful to the AI-center of P.G. Demidov Yaroslavl State University for providing an access to NVIDIA DGX-1 supercomputer.

REFERENCES

[1] Y. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, The MIT Press, 2016, 800 p.
[2] Inria Aerial Image Labelling Database, Web: https://project.inria.fr/aerialimagelabeling/.
[3] DigitalGlobe satellite, Web: https://www.digitalglobe.com.
[4] E. Shelhamer, J. Long, T. Darrell, "Fully Convolutional Networks for Semantic Segmentation", Web: https://arxiv.org/pdf/1605.06211.pdf.
[5] S. Seferbekov, V. Iglovikov, A. Buslaev, A. Shvets, Web: https://arxiv.org/pdf/1806.03510.pdf.
[6] DeepGlobe. CVPR 2018 – Satellite Challenge, Web: http://deepglobe.org.
[7] O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", *Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer, LNCS*, vol. 9351, 2015, pp. 234–341.
[8] G. Chhor, C. Bartolome Aramburu, I. Bougdal-Lambert, "Satellite Image Segmentation for Building Detection using U-net", Web: http://cs229.stanford.edu/proj2017/final-reports/5243715.pdf.
[9] A. Chaurasia, E. Culurciello, "LinkNet: Exploiting Encoder Representations for Efficient Semantic Segmentation", Web: https://arxiv.org/pdf/1707.03718.pdf.
[10] The Cambridge-driving Labeled Video Database (CamVid), Web: http://mi.eng.cam.ac.uk/research/projects/VideoRec/CamVid/.
[11] SpaceNet Database, Web: http://explore.digitalglobe.com/spacenet.
[12] DSTL Satellite Imagery Feature Detection, Web: https://www.kaggle.com/c/dstl-satellite-imagery-feature-detection.
[13] Sovzond, Web: https://sovzond.ru/services/remote-sensing/.
[14] V. Khryashchev, L. Ivanovsky, V. Pavlov, A. Ostrovskaya and A. Rubtsov, "Comparison of Different Convolutional Neural Network Architectures for Satellite Image Segmentation," *2018 23rd Conference of Open Innovations Association (FRUCT), Bologna, 2018*, pp. 172-179.
[15] A. Gulli, S. Pal., *Deep Learning with Keras*, Packt Publishing, 2017, 320 p.
[16] J. Sanders, E. Kandrot, *CUDA by Example: An Introduction to General-Purpose GPU Programming*, Addison-Wesley Professional, 2010, 320 p.
[17] A. Muller, S. Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly, 2016, 400 p.
[18] T.S.K.M. Rabie, "Implementation of some similarity coefficients in conjunction with multiple upgma and neighbor-joining algorithms for enhancing phylogenetic trees", *Egypt. Poult. Sci*. Vol. 30 (II), pp.607-621.