Neural network technology to search for targets in remote sensing images of the Earth

N S Abramov[1], A A Talalayev[1], V P Fralenko[1], O G Shishkin[1] and V M Khachumov[1,2]

[1]Aylamazyan Program Systems Institute of Russian Academy of Sciences, Peter the First Street, 4 "a", Veskovo Village, Yaroslavl Region, Russia
[2]The Peoples' Friendship University of Russia, Miklukho-Maklaya Street, 6, Moscow, Russia

e-mail: shishkinog@mail.ru, nikolay.s.abramov@gmail.com

Abstract. The paper introduces how multi-class and single-class problems of searching and classifying target objects in remote sensing images of the Earth are solved. To improve the recognition efficiency, the preparation tools for training samples, optimal configuration and use of deep learning neural networks using high-performance computing technologies have been developed. Two types of CNN were used to process ERS images: a convolutional neural network from the nnForge library and a network of the Darknet type. A comparative analysis of the results is obtained. The research showed that the capabilities of convolutional neural networks allow solving simultaneously the problems of searching (localizing) and recognizing objects in ERS images with high accuracy and completeness.

CEUR-WS.org/Vol-2391/paper25.pdf

1. Introduction

Today, there is an upsurge of activity in the field of Earth remote sensing (ERS) data processing: new software systems are being created, high-resolution image processing methods are being modernized. The current situation is characterized by the improvement of the equipment of spacecraft (SC) and ground control stations, the expansion of the functionality and spectrum of the image processing tasks performed. The scope of application of these spacecraft includes monitoring of forest, agricultural and arctic zones, analysis of natural disasters, environmental protection, public safety, etc. The growing volumes of evolving ERS data have significantly increased the requirements for speed and quality of information processing. Recently, artificial neural networks (ANN) and high-performance computing technologies have been increasingly used.

The analysis of modern work on the application of ANN has shown that neural networks are mainly used for searching and recognizing targets that are related to the category of nonrigid [1,2]. The authors of this paper created a scientific and practical groundwork in solving various problems based on intelligent processing of ERS images (multispectral, panchromatic, color) search for rigid objects and zones of interest using the developed spectrographic approach and the generalized metric (fires, inundations, ice conditions assessment, etc.) [3-8]. The proposed paper presents the results of new in-depth studies related to the use of modern convolutional neural networks (CNN) for processing panoramic full-color ERS images obtained from unmanned aerial vehicles (UAVs); some methods and

tools to improve their efficiency and performance during the search and recognition of the objects of military equipment with the necessary completeness and accuracy, which still remains unresolved even with an abundance of software, are proposed. The modern formulation of the task of finding and recognizing an object by a neural network includes the steps of selecting the type, setting the parameters of the ANN and preparing the input data. The multi-class and single-class problems were considered as part of the study. The first task is thought of as the search and recognition of objects of several classes simultaneously. The second task involves the search by a neural network of objects of a single class.

2. Methods, software tools and results of image processing using ANN

Two types of CNN were used to process ERS images: a convolutional neural network from the nnForge library [9] and a network of the Darknet type [10]. Both implementations are distinguished by the support of various types of layers; therewith a flexible configuration is provided and it is possible to change the structure to suit own needs. In addition, the network of the second type not only classifies the target objects but also reports on their positions in the shot. A distinctive feature of the considered CNN is the support of computational speedup using graphics processing units (GPU) both during training and operation. Copies of trained ANN are distributed between the existing GPUs where data are processed independently and asynchronously. A special software complex for designing neural network application systems was used to implement the computational process [11, 12].

A special tool for the automated preparation of training samples has been implemented to improve the quality of the classification. A human expert prepares preliminarily some images with a transparency marker set, where background pixels are set invisible using alpha channel controls. Figure 1 shows the original fragment of the ERS image, on the right: the same fragment is shown after the alpha channel change. For convenience, images with objects of different classes are sorted to different directories.

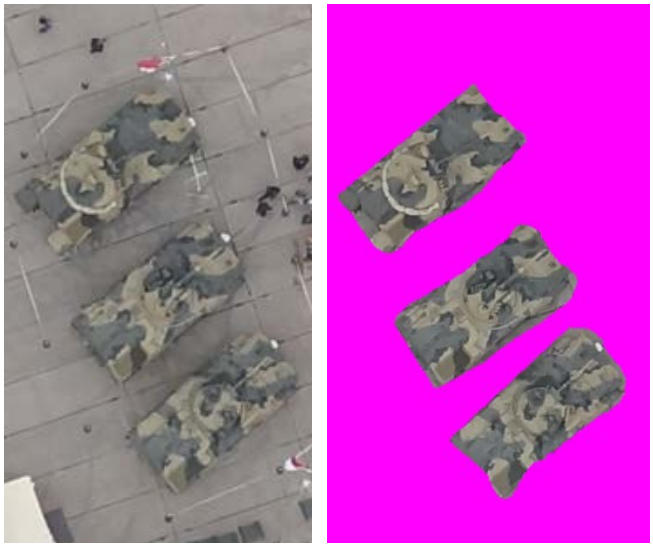

Figure 1. Images: original and with the alpha channel mask.

Various settings of the ANN from the nnForge library (configurations and characteristics of the layers) can be customized with scaled copies of these images. The scaling factor is chosen so that each target object is placed in a separate scanning window, the size of which coincides with the size of the input ANN window. The experiments were conducted on military equipment images of 6000x4000 pixels, made from a height of 300 meters at the Russia Arms Expo – 2015 (RAE-2015) international exhibition.

The following CNN architecture from the nnForge library was experimentally chosen:

– contrast extraction layer with a 9x9 pixel Gaussian window [13]; the original size of the data window: 39x39;

– convolution layer with the 6x6 feature maps (total 136 maps), the hyperbolic tangent module is used for normalization, the window size after processing is 34x34;

– average subsampling layer with a 2x2 mask, the window size after processing is 17x17;

– convolution layer with the 6x6 feature maps (total 272 maps), the window size after processing is 12x12;

– average subsampling layer with a 2x2 mask, the window size after processing is 6x6;

– convolution layer with the 6x6 feature maps (total 544 maps), the window size after processing is 1x1;

– dropout layer with an adjustable probability of disabling connections between neurons (experimentally set to 0.05);

– convolution layer with the 1x1 feature maps, the number of feature maps corresponds to the number of distinguished classes, the hyperbolic tangent type activation function is used.

The scanning window during the recognition moves through the image in increments of one pixel and is processed by the neural network. The sequential processing of the entire image results in a colored map where the target objects are separated from the background. Figure 2 shows an example of the original image and the result of its processing.

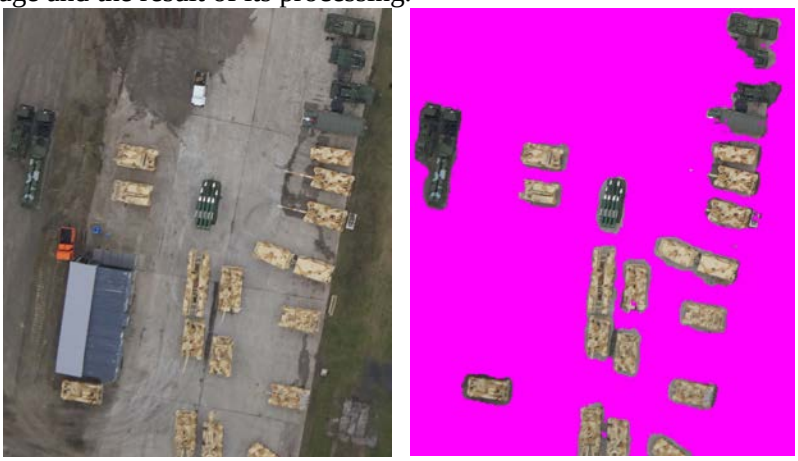

Figure 2. The original image and the result of its processing.

3.626 million objects automatically extracted from 73 images were used for training. The following results have been achieved: Classification completeness: background – 0.9976, materiel – 0.9354. Classification accuracy: background – 0.9392, materiel – 0.9974. Training time: 24 hours on one Nvidia Geforce GTX 1060 and using a single CPU core Intel Core i7 6850K (of the existing 6 cores, 3.6 @ 4.0 GHz). Processing time of ten panoramic images on one GPU: 3051 s; on two GPUs: 1640 s.

As a result of numerous experiments, the developers of the Darknet network have selected a very successful architecture, such that it works on different training / test samples [14, 15]. The network, within certain limits, is resistant to the fact that images of different sizes and subjected to geometric distortions can be input. The main adjustable parameter is the size of the CNN input layer.

The software tool developed as part of this study includes the programs YOLORotate, YOLOAnchors and YOLOGetObjects.

The YOLORotate program is designed to convert images into a format suitable for training the YOLO v2 type ANNs. The data needed for the preparation of a series of our experiments include many panoramic images taken from the UAV. They have four classes of target objects: IFV (infantry fighting vehicles), Military vehicles, SPG (self-propelled gun mounts) and Tanks; information about the coordinates and sizes of each of the objects is pre-assembled and stored in text files. Each such image has a size of not less than 832x832 pixels, where all target objects occupy a relatively small part of the image. YOLORotate rotates images from a training sample with a given step, for example, 15 degrees. At the same time, the maximum possible number of fragment images with target objects is cut out. A total of 4361 fragments for the training sample and 1173 fragments for the test sample were automatically created. It is guaranteed that on each such fragment there is at least one target object. In

order to ensure that the target objects are located in random positions of the received fragment images, a pseudorandom number generator is used. The analysis showed that with the selected size of the source data of 832x832 pixels, up to 20 targets fall into the frame. The YOLOAnchors program uses the k-means method [16] to detect the width and height of typical targets on the output window of a neural network. The program finds several such pairs of sizes, which are used later in the training of the ANN.

The program YOLOGetObjects segments the images into fragments of 832x832 pixels while providing a partial intersection, and each next window captures a quarter of the previous one (both horizontally and vertically). In total, there are 70 fragments per panoramic picture. Further, all fragments are independently processed using a GPU and a general-purpose processor. The next step is to combine information about all the target objects found. Figure 3 shows an example of the result of using a trained ANN.

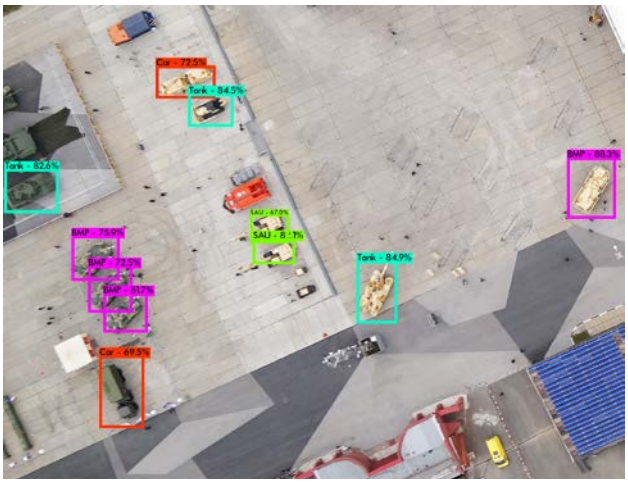

Figure 3. The result of shot processing by a multi-class ANN.

Table 1. ANN test results with an input window of 416x416 pixels.

		6 images per pack	14 images per pack
Share of target objects found	IFV	0.9346	0.9731
	Military cars	0.9880	0.9983
	SPG	1.0000	1.0000
	Tanks	0.9967	0.9934
Average share of target objects found		0.9798	0.9912
Completeness	IFV	0.7500	0.6885
	Military cars	0.9811	0.9949
	SPG	0.9006	0.9655
	Tanks	0.9058	0.9107
Normalized accuracy	IFV	0.9118	0.8947
	Military cars	0.8627	0.9541
	SPG	0.9990	1.0000
	Tanks	0.8561	0.7672
F1-measure	IFV	0.8230	0.7782
	Military cars	0.9181	0.9741
	SPG	0.9473	0.9825
	Tanks	0.8802	0.8328
Average *F1*-measure		0.8922	0.8919
The ratio of the number of found targets to their total number		0.9869	0.9948
Training time, hours		1.18	2.44
6000x4000 pixels image processing time, seconds		2.13	

Table 2. ANN test results with an input window of 832x832 pixels.

		6 images per pack	14 images per pack
Share of target objects found	IFV	0.9615	0.9538
	Military cars	0.9983	0.9974
	SPG	1.0000	1.0000
	Tanks	0.9967	0.9934
Average share of target objects found		0.9891	0.9862
Completeness	IFV	0.5846	0.6077
	Military cars	0.9657	0.9657
	SPG	0.9290	0.9432
	Tanks	0.9421	0.9322
Normalized accuracy	IFV	0.8630	0.9004
	Military cars	0.9462	0.9160
	SPG	0.9927	0.9724
	Tanks	0.7123	0.7484
F1-measure	IFV	0.6970	0.7256
	Military cars	0.9558	0.9402
	SPG	0.9598	0.9576
	Tanks	0.8113	0.8303
Average *F1*-measure		0.8560	0.8634
The ratio of the number of found targets to their total number		0.9945	0.9925
Training time, hours		4.78	11.11
6000x4000 pixels image processing time, seconds		5.50	

Table 3. ANN test results with an input window of 416x416 pixels and increased number of feature maps.

		6 images per pack	14 images per pack
Share of target objects found	IFV	0.9615	0.9500
	Military cars	1.0000	0.9983
	SPG	1.0000	1.0000
	Tanks	0.9934	0.9983
Average share of target objects found		0.9887	0.9867
Completeness	IFV	0.7846	0.7538
	Military cars	0.9923	0.9966
	SPG	0.9432	0.9473
	Tanks	0.9223	0.9438
Normalized accuracy	IFV	0.8970	0.9230
	Military cars	0.9421	0.9166
	SPG	1.0000	1.0000
	Tanks	0.8510	0.8616
F1-measure	IFV	0.8371	0.8299
	Military cars	0.9665	0.9549
	SPG	0.9708	0.9729
	Tanks	0.8852	0.9008
Average *F1*-measure		0.9149	0.9146
The ratio of the number of found targets to their total number		0.9945	0.9937
Training time, hours		4.69	10.37
6000x4000 pixels image processing time, seconds		3.98	

The batch training where the next step of adjusting the weighting factors is based on information about the results of processing a limited group of images of the training sample was used in all experiments. Each group of images on the new training period is formed randomly; preference is given to the groups with representatives of all classes of target objects. Using batch training allows improve the quality of the neural network and abandon the resource-intensive dropout layer [17]. The best package size is chosen experimentally for each problem to be solved.

Tables 1-3 show the refinement characteristics and the results of the experiments performed in solving a multi-class problem — simultaneous search and recognition of objects of four classes.

Table 4 shows the comparative results of processing the test sample when solving single-class and multi-class problems. The training time of the selected network configuration on each of the four classes of military equipment was 6.84, 13.6 and 26.8 hours when working with groups of 28, 56 and 112 images. In the single-class case, the ANN works with only one class; the user has the opportunity to choose the best option – the network trained using a group of images of the optimal size. In the last column of table 4, the best coefficients for mixed mode are collected, when a network trained for individual classes is used. For instance, a network trained on a package of 56 images is used for IFV, and for the Tanks class – a network trained on a package of 112 images. The average processing time of a 6000x4000 pixels image in one separate single-class neural network is the same as that of a multi-class neural network, that is, four seconds.

Table 4. Experimental data, multi-class and single-class problems.

Class of objects	Share of founded objects and the size of the group of images					
	Multi -class problem		Single-class problem			
	6	14	28	56	112	/
IFV	0.9615	0.9500	0.8577	0.9115	0.8538	0.9115
Military cars	1.0000	0.9983	0.9057	0.8885	0.8954	0.9057
SPG	1.0000	1.0000	0.8276	0.9635	0.9473	0.9635
Tanks	0.9934	0.9983	0.9174	0.9455	0.9521	0.9521
Average share of target objects found	0.9887	0.9867	0.8771	0.9273	0.9121	0.9332
Training time of one separate neural network, hours	4.69	10.37	6.84	13.6	26.8	–

The results of the experiments confirmed the effectiveness of the use of single-class neural networks. However, training a complex of such networks requires more computing resources than those used for training of one multi-class network. An increase in the complexity of the task with the same number of feature maps leads to a decrease in the average share of the found target objects for a single-class ANN.

3. Conclusion

The article presents the results of research related to the use of modern convolutional neural networks for processing panoramic full-color aerial ERS images. Using the nnForge and Darknet CNN, multi-class and single-class problems of searching and classifying targets are solved. Some methods for preparing training samples, optimal configuration, and the use of high-performance computing have been developed to improve the recognition efficiency. A comparative analysis showed that the one-class approach has an advantage in recognition quality but loses in operation time. In general, it should be noted that the capabilities of convolutional neural networks allow solving simultaneously the problems of searching (localizing) and recognizing objects in ERS images with high accuracy and completeness.

4. References

[1] Vizilter Yu V, Gorbatsevich V S, Vorotnikov A V and Kostromov N A 2017 Real-time face identification via CNN and boosted hashing forest *Computer Optics* **41(2)** 254-265 DOI: 10.18287/2412-6179-2017-41-2-254-265
[2] Ivanov A I, Lozhnikov P S and Sulavko A E 2017 Evaluation of signature verification reliability based on artificial neural networks, Bayesian multivariate functional and quadratic forms *Computer Optics* **41(5)** 765-774 DOI: 10.18287/2412-6179-2017-41-5-765-774
[3] Fralenko V P 2010 Spectrographic texture analysis for earth remote sensing data *Artificial Intelligence and Decision Making* **2** 11-15
[4] Fralenko V P 2018 Intelligent analysis of aerospace images using high-performance computing devices *Proceedings of the conference "Artificial Intelligence: Problems and Solutions"* (Moscow region, Patriot Park)
[5] Abramov N S, Agronik A Yu, Emelyanova Yu G, Latyshev A V, Talalaev A A, Fralenko V P and Khachumov M V 2017 Methods, models and software for processing data for space monitoring of the Arctic zone *Aerospace Instrument-Making* **7** 38-51
[6] Fralenko V P 2017 Localization and classification of military equipment in the stream of images from UAVs *Materials of the conference "Fundamental Science for Army" within of the Third International Military-Technical Forum "ARMY-2017"* (Moscow region, Patriot Park)
[7] URL: https://www.science-education.ru/ru/article/view?id=18607
[8] Khachumov V M, Fralenko V P, Chen Guo Xian and Zhang Guo Liang 2015 Construction perspectives of the remote sensing data high-performance processing system *Program Systems: Theory and Applications* **1** 121-133
[9] URL: http://milakov.github.io/nnForge
[10] URL: https://arxiv.org/abs/1612.08242
[11] Talalaev A A and Fralenko V P 2013 The complex of tools for the design of neural network application systems *Scientific and Technical Volga region Bulletin* **4** 237-243
[12] Talalaev A A and Fralenko V P 2013 The architecture of a parallel-pipeline data processing complex for heterogeneous computing environment *Bulletin of Peoples' Friendship University of Russia. Mathematics series. Computer science. Physics* **3** 113-117
[13] URL: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.0040027
[14] Everingham M, Van Gool L, Williams C K, Winn J and Zisserman A 2010 The pascal visual object classes (voc) challenge *International journal of computer vision* **88** 303-338
[15] Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollar P and Zitnick C L 2014 Microsoft coco: Common objects in context *In European Conference on Computer Vision*
[16] Celebi M E, Kingravi H A and Vela P A 2013 A comparative study of efficient initialization methods for the k-means clustering algorithm *Expert Systems with Applications* **40** 200-210
[17] Srivastava N, Hinton G E, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 Dropout: a simple way to prevent neural networks from overfitting *Journal of Machine Learning Research* **15** 1929-1958

Acknowledgements

This work was supported by the Russian Foundation for Basic Research (projects No. 18-29-03011-mk Research and Development of New Methods and Technologies for the Tasks of Intellectual Analysis and Optimization of Processing Large Data Streams of the Earth Remote Sensing and No. 17-29-07003-ofi_m Development of Methods and Models of Dynamic Behavior Planning and Hierarchical Intellectual Motion Control of Unmanned Aerial Vehicles in an Uncertain Environment with Computing Resources Constraints).