The using of data augmentation in machine learning in image processing tasks in the face of data scarcity

N A Andriyanov¹,², D A Andriyanov²

¹JSC "RPC "Istok" named after Shokin", 2, Vokzalnaya st., Fryazino, Moscow Region, 141190, Russia
²Ulyanovsk State Technical University, 32, Severny Venets st., Ulyanovsk, 432027, Russia

E-mail: nikita-and-nov@mail.ru

Abstract. The article presents the results of a study of the efficiency of various neural networks in the limited conditions of the source data and with a number of simple augmentations. In this case, the dependences were obtained for a serial neural network with back propagation of error. For data augmentation, the simplest transformations were used, including the letters tilting (italics), changing the color of letters (from black to red), as well as distortion of the reference images with white Gaussian noise at a signal-to-noise ratio q from 1 to 10. It is shown that the best results of recognition of letters of the Russian alphabet are provided by a network for which all the augmentation methods discussed in this work were used. A study of the dependence of recognition accuracy on the signal-to-noise ratio in all trained neural networkswas also conducted.

1. Introduction

Currently, image processing is becoming particularly popular in various systems of technical vision. In this case, image processing methods can be divided into 3 groups. The first group includes such processing algorithms that are based on the description of images using mathematical models of random fields [1-4]. The second group includes local processing operators that are well established for solving highly specialized problems [5-7]. Finally, methods of the 3rd group are associated with the use of machine learning. Furthermore in recent years deep learning [8–15] has become increasingly popular. The text considers in more detail the last group in terms of the influence of the amount of available data on the efficiency of the neural networks algorithms.

Today, there are many areas of research in which scientists hope to improve current results using deep convolutional networks for computer vision tasks. Improving the generalizing ability of such models is one of the most difficult tasks. Generalizability refers to the difference in model performance when evaluating using previously viewed data (training data) compared to data that has never been seen before (test data). Models with poor generalizability show significantly greater accuracy on the training sample than that on the test sample. This is the effect of retraining. Therefore usually in each epoch of training, it is necessary to correlate the accuracy (or efficiency) in the sample for training with the accuracy (or efficiency) in the test sample. To build useful deep learning models, the error in the verification data should decrease along with the error in the training sample. Increasing data is a very powerful method to achieve this. Extended data will represent a more complete set of
possible data points, thus minimizing the distance between the training and test set, and any future test sets.

To reduce the effect of retraining and increase generalizability, it is necessary to increase the amount of data. If an increase in the volume of real data is difficult, then various techniques for inflating data based on existing ones come into play [14, 15]. Many other generalization performance enhancement strategies focus on the model architecture itself. This led to a sequence of progressively more complex architectures from AlexNet [8] to VGG-16 [9], ResNet [10], Inception-V3 [11], and DenseNet [12]. Functional solutions such as dropout regularization, sample normalization, translation of training results, and pre-training have been developed to try to expand deep learning for use on small data sets. A complete review of regularization methods in deep learning was compiled by J. Kukacka et al. [13]. However, the goal of this paper is to show how beneficial augmentation of data can be even when solving the classical problem of recognizing letters of the alphabet.

2. Preparation of basic sample for training
The Russian alphabet containing 33 letters was chosen as the data for recognition. The basic selection was prepared in the Microsoft Paint editor and consisted of 33 “.png” files, each of which represented a specific letter from A(А) to IA(Я) (the same from A to Z in the English alphabet). All letters were in Times New Roman, font size - 26 pt, and the size of the entire image was 30x30 pixels. Figure 1 shows examples of some obtained images of the letters.

![Figure 1](image1.png)

**Figure 1. Basic data for training**

From Figure 1 it can be seen that the presented set of letters A(А), G(Г), Y(Й), T(Т), IA(Я) looks convenient for human perception, and the selected parameters make it easy to conduct a visual recognition procedure.

Let us imagine that a situation is possible when it would be necessary for us to recognize such handwritten texts, for example, from a teacher making notes with a red pen. Such assumption is made in order to make the meaning of color important for our neural network. Therefore, when training, we must use all the color channels of the image. Thus, one matrix will be described by 3 matrices with a size of 30x30 elements, which can be represented as a vector of length \( N = 3\times30\times30 = 2700 \).

3. Data augmentation
The proposed sample can be easily trained, but with a high degree of probability it will not be able to efficiently recognize colored letters, inclined letters, or distorted letters on a full set of letters. This is just a series of directions in which augmentation of the base sample is possible. Let's make the same selection, consisting of letters written in italics. Figure 2 shows examples for the same letters A(А), G(Г), Y(Й), T(Т), IA(Я).

![Figure 2](image2.png)

**Figure 2. Italic type alphabet**

The next option for data augmentation is to apply different colors to an existing base sample. For simplicity, let us assume that in addition to black letters, red ones may also appear. Figure 3 shows the application of color conversion to the base sample for letters A(А), G(Г), Y(Й), T(Т), IA(Я).

![Figure 3](image3.png)

**Figure 3. Alphabet in red color**

Finally, when transmitting images or during their photo registration, various interference may occur. In this regard, it is proposed to add the augmentation method, which is a direct mixture of the
available images from the base sample and white Gaussian noise. Figure 4 presents a variant of the obtained sample for the signal-to-noise ratio \( q = 1 \), and Figure 5 shows noisy data for \( q = 10 \). The examples show the letters A(А), G(Г), Y(Й), T(Т), IА(Я) again but with color inversion.

**Figure 4.** Alphabet distorted by noise comparable to signal (inversion)

**Figure 5.** Alphabet distorted by noise ten times smaller than signal (inversion)

It should be noted that this is only an insignificant part of the transformations that can be used for data augmentation. For example, it was possible to use many rotations, scaling and cropping data. But for further research, let us restrict ourselves to the augmentations described above.

### 4. The structure of neural networks and training samples

After the basic data were prepared and data augmentation was performed, 5 neural networks having the same structure were trained using different data. Figure 6 shows the structure of neural networks.

**Figure 6.** Neural network structure

As can be seen from Figure 6, 900 samples describing one letter are fed to the input, then there is one hidden layer consisting of 10 neurons. The output is a vector of 33 values, showing proximity to a particular letter.

So 5 neural networks were prepared as follows, depending on the training set. Table 1 presents the conventional name of the network and the corresponding training sample.
Table 1. Composition of the training set for various neural networks

| Neural Network        | Training dataset                                                                 |
|----------------------|-----------------------------------------------------------------------------------|
| Network Base (Network №1) | Basic alphabet only (33 letters)                                                   |
| Network Italic (Network №2) | Alphabet in italics only (33 letters)                                             |
| Network Color (Network №3) | Basic alphabet + alphabet in red color (66 letters)                              |
| Network BIC (Network №4) | Basic alphabet + alphabet in red color + italic alphabet (99 letters)             |
| Network BICN (Network №5) | Basic alphabet + alphabet in red color + italic alphabet + noisy alphabet (q = 1…10) (2089 letters) |

Figure 7 shows the learning processes of neural networks presented in Table 1. Figure 7a corresponds to Network No. 1, Figure 7b corresponds to Network No. 2, Figure 7c corresponds to Network No. 3, Figure 7d corresponds to Network No. 4, Figure 7e corresponds to Network No. 5. The blue line shows errors on the training sample, the green lines shows errors on the validation sample, and the red line shows errors on the test sample.

From the graphs of Figure 7 it can be seen that the retraining effect is absent only in Network No. 5, which is represented by the curves in Figure 7e. For other networks, there is significant retraining because the error on the test sample is much less than validation and test error.

5. Recognition results

All trained neural networks from Table 1 were tested using data including images from the base alphabet and each augmented alphabet. Figure 8 shows the recognition results in terms of accuracy. The abscissa axis represents the noise variance (taking into account the single variance of the signal), and the ordinate axis represents the number of precisely recognized alphabet characters in percent. Lines 1,2,3,4,5 correspond to Networks No. 1, No. 2, No. 3, No. 4, No. 5 respectively.
Figure 8. Comparison of recognition accuracy for different neural networks

The analysis of the obtained graphs shows that the recognition results of the test sample are significantly affected by the volume and structure of the training sample. The best results are provided by Networks No. 4 and No. 5, in which the training used the largest number of augmentations.

6. Conclusion
The simplest example of recognizing the symbols of the Russian alphabet shows how various data transformations can affect the efficiency of a neural network. In particular, several augmented alphabets were generated from 33 letters. A study for 3 augmentation options and their mixtures showed that it was possible to increase recognition accuracy up to 5 times compared to a network based only on a basic data set. Thus, it is possible to conclude that in conditions of insufficient data, one of the options for the improving accuracy may be the use of the data augmentation.

References
[1] Smith J, Best E, Sum E, Guzel Y, Saville M A, LoMonte L and Wicks M 2015 International Conference on Electromagnetics in Advanced Applications (ICEAA). DOI:10.1109/iceaa.2015.7297179
[2] Andriyanov N A, Vasiliev K K, Dementiev V E 2017 CEUR Workshop Proceedings 1901 10-15
[3] Vasiliev K, Dementiev V and Andriyanov N 2018 Procedia Computer Science 126 49-58 DOI: doi.org/10.1016/j.procs.2018.07.208
[4] Andriyanov N A, Vasiliev K K 2019 CEUR Workshop Proceedings 2391 72-78 DOI: 10.18287/1613-0073-2019-2391-72-78
[5] Arefyev E Yu, Proskurin A V 1990 Computer Optics 7 97-102
[6] Myasnikov V V 2007 Computer Optics 31(4) 86-94
[7] Myasnikov V V 2007 Computer Optics 31(2) 52-68
[8] Krizhevsky A, Sutskever I, Hinton G E 2012 Adv Neural Inf Process Syst. 25 1106–1114
[9] Karen S, Andrew Z 2014 arXiv e-prints 106-118
[10] Kaiming H, Xiangyu Z, Shaqing R, Jian S 2016 CVPR1-12
[11] Christian S, Vincent V, Sergey I, Jon S, Zbigniew W 2015 arXiv e-prints 1-10
[12] Gao H, Zhuang L, Laurens M, Kilian Q W 2016 arXiv preprint 1-7
[13] Jan K, Vladimir G, Daniel C. 2017 arXiv preprint 1-11
[14] Buslaev A, Parinov A, Khvedchenya E, Iglovikov V, Kalinin A 2018 arXiv:1809.06839v11-4
[15] Akimov A V, Sirota A A 2016 Computer Optics 40(6) 911-918