IDENTIFICATION OF NATURAL OBJECTS USING DEEP LEARNING AND ADDITIONAL DATA PREPROCESSING

D. A. Kalashnikova[1,*], V. V. Buryachenko[1]

[1] Reshetnev Siberian State University of Science and Technology, Institute of Informatics and Telecommunications, 31, Krasnoyarsky Rabochy ave., Krasnoyarsk, 660037 Russian Federation – krutko.d00@gmail.com, buryachenkovv@gmail.com

Commission II, WG II/8

KEY WORDS: Classification of natural objects, Neural networks, Preprocessing methods, Deep learning, Stolby National Park.

ABSTRACT:

The classification of natural objects in the wild is a popular task in the field of tourism and remote sensing. The key problem is the requirement for system performance in the absence of Internet access and a small amount of available resources, such as a mobile phone. In this regard, to solve the classification problem, it is required to use fairly simple neural networks and rely on a small amount of training data. The paper presents an image preprocessing method for object recognition in the the “Stolby National Park” in Krasnoyarsk city using a neural network. The approach involves applying a set of methods to expand the original training set. To analyze the effectiveness, several different neural networks based on MobileNET V2 are used, which makes it possible to compare test results on the original and extended data sets. We also evaluate the quality of objects identification on open datasets, such as Animals-10 and Landscape Pictures. The results of the experiments show the efficiency of data preprocessing, as well as the high performance of the modified neural network structure for the task of classifying natural objects in the environment.

1. INTRODUCTION

Image classification is one of the most widely studied topics in the field of computer vision, for which many algorithms have been developed. The use of convolutional neural networks is one of the most modern approaches. In image classification, we classify an image according to one of the predefined classes or several classes at the same time. In multi-label image classification, an image can have multiple classes among the class set, whereas in simple image classification, an image only contains one class among the class set.

Object identification in the wild is the task where we need to recognize the natural object which may to belong to the same class with different objects, for example, in a national park, which increases the error probability by using classical methods. It is also necessary to consider the complexity of accessing the Internet and the small amount of resources that a tourist with a mobile phone has. Therefore, in the paper, we rely on relatively simple architectures of neural networks and a small amount of training set obtained manually by the authors.

The paper presents an image preprocessing method for identifying objects using a neural network in the the Stolby National Park in the Krasnoyarsk city. Due to the fact that there are not enough images on the Internet and collected manually to train a neural network, it was decided to implement a method for expanding the training dataset, which includes geometric transformations of rotation and scaling and preprocessing algorithms to improve image quality. We can analyze the performance of a convolutional neural network by training the model on different datasets before and after expansion. In the experiments, the effectiveness of the proposed preprocessing method in training the model was evaluated and various methods for expanding the data set, as well as the architecture of neural networks, were considered.

The rest of the paper is organized as follows. Related work is reviewed in Section 2. In Section 3, the preprocessing data methods are presented, while the neural network training process are considered in Section 4. Experimental results are reported in Section 5, and Section 6 concludes this paper.

2. RELATED WORK

The task of classifying natural objects in nature reserves is difficult to implement with sufficient quality, which is associated with the difficulty of obtaining a large sample of the training data. It is required to shoot natural objects in various natural conditions, which is not always possible. Therefore, to improve the quality of training of neural networks, methods of preprocessing and augmentation of the data sample are often used (Boekestijn et al., 2018).

There are two known approaches: the first is to create synthetic data by applying geometric transformations such as rotation, scaling and data cropping, and the second are more complex methods such as mix-up augmentation, cut- out augmentation (Devries et al., 2017), between class learning (Simard et al., 2003). In reference, the methods of preprocessing and expanding the training set are used in the problem of classifying plant images, which made it possible to increase the training accuracy by more than 3%. The authors show that the expansion of a data set using realistic transformations avoids overfitting and improves performance in the task of recognizing handwritten digits, for example, based on the well-known MNIST data set. (Wong et al., 2016). In the paper (Cisse, 2018) authors proposed a data extension method based on the idea of image style transfer. The authors showed an increase in efficiency using the example of processing medical images of skin diagnostics and MRI-analysis (Mikołajczyk et al., 2018).

Publisher's note: Copernicus Publications has not received any payments from Russian or Belarusian institutions for this paper.

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-2-W3-2023-95-2023 | © Author(s) 2023. CC BY 4.0 License.

One of the most popular tasks of neural networks is the recognition of visual images. Today, networks are being created in which machines are able to successfully recognize symbols on paper and bank cards, signatures of official documents, etc. (Zlobin et al., 2020, Finogeev et al., 2021). These functions make it possible to significantly facilitate human labor, as well as to increase the reliability and accuracy of various work processes due to the absence of the possibility of making mistakes due to the human factor (Mustaev et al., 2019). The work (Zhao et al., 2018) proposes an object-oriented deep learning method for accurate classification of high-resolution urban images without intensive human intervention to improve the quality of deep neural network feature building, which gives a classification accuracy above 90%. The authors (Supriya et al., 2020) consider using the well-known Alex-Net network to classify objects in natural scenes into 12 classes using preprocessing and achieve an accuracy of 90%.

3. DATA PROCESSING METHODS

In the process of preparing a data set for training a neural network, natural objects of different classes were photographed in the winter and autumn time of the day. The KrasnoyarskStolby dataset includes more than 40 images for each of the 9 object classes, and reflects the most memorable tourist places. Due to the small sample size, it is necessary to process and expand the data set to improve the quality of training. The general scheme of the proposed algorithm is shown in Figure 1.

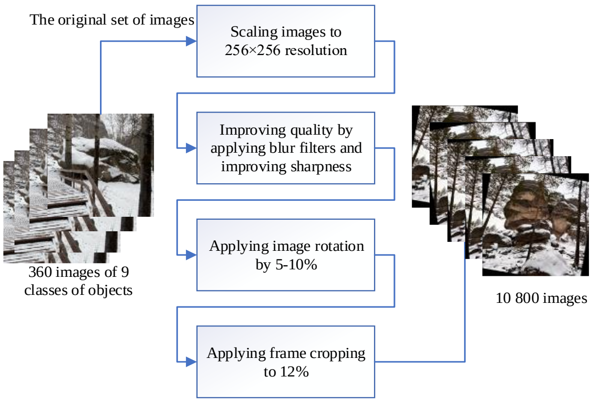

The original set of images
Scaling images to 256×256 resolution
Improving quality by applying blur filters and improving sharpness
Applying image rotation by 5-10%
Applying frame cropping to 12%
360 images of 9 classes of objects
10 800 images

Figure 1. The general scheme of the algorithm for preprocessing a data set of natural objects to improve the quality of neural network training.

Since it is planned to develop a mobile application that will use a neural network to classify images, a resolution of 256× 256 has been selected, at which the necessary speed of the system will be provided. When shooting tourist sites, photos were created in a square format to avoid distortion during network training. A function has been developed that allows us to resize all images in accordance with the resolution of the neural network.

Further, when using the *Pillow* library, a *Detail* filter is applied to the function that passes through all the pictures of each class, which makes the details of the image more obvious (Mustaev et al., 2019). Image blurring occurs using a sequence of extended rectangular filters that approach the Gaussian kernel, where everything comes down to convolution of the input signal with a kernel function with infinite support. Formula of the *m*-dimensional Gaussian kernel is presented below:

$$K_{\sigma}(x) = \frac{1}{\left(2\pi\sigma^{2}\right)^{\frac{m}{2}}} \exp\left(-\frac{|x|^{2}}{2\times\sigma^{2}}\right) \qquad (1)$$

where σ is the standard deviation of the Gaussian distribution, which has the shape of a bell-shaped curve that rapidly declines to the side ±∞ (Gwosdek et al., 2011).

In addition to the filter presented above, a *Smooth* filter has been applied, which allows you to create a smooth filter by gradually changing the brightness of the image, reducing the sudden gradient and improving the image better. Two simple moving averages are calculated on top of each other to give more weight to closer (neighboring) points. Thus, the triangular moving average is calculated by the formula:

$$TMA_{i} = \frac{SMA_{i} + \ldots SMA_{i+n}}{n} \qquad (2)$$

where $SMA_{i} = \frac{y_{i} + \ldots + y_{i+n}}{n}$ is a simple moving average
y = points with values
n = the window size (Smoothing in Python)

Thus, 1 data set was created, which has the increased number of images from 360 to 1800 images after transformations using *Detail* and *Smooth* filters.

By the next stage, the volume of the data set has already been doubled. Next, for each class, using the *rotate*() function, the image is rotated by 5° and 10° clockwise and counterclockwise. After that 2 data sets were created, the data increased from 1800 to 5400 images. The next preprocessing is scaling the images by 12% of the original. *Crop* (left, upper, right, lower) function was used to crop the image. In result, 3 data sets were created, the dataset increased from 5400 to 10800 images.

4. NATURAL OBJECTS IDENTIFICATION

Today there are many neural networks that recognize animals, objects and people. As a task to be solved, it was decided to develop an application for identifying tourist sites located on the territory of the Stolby National Park. For this task, a mobile application will be developed, so it was decided to consider MobileNetV2 as a basic model for identifying tourist objects. This model has a small weight, which allows us to store the trained model on a mobile device, and also has a very high accuracy and response time when working on 4 cores of a mobile device (Bahlmann et al., 2005).

Further classification of natural objects is carried out using two models of convolutional neural networks. The first model consists of three convolution layers, two fully connected and one output layer. For source images with a size of 256×256, layers 1-8 are used to highlight important features in the image, and layers 9-14 are used for classification.

The second model of a convolutional neural network includes three convolution layers, a fully connected and an output layer. For source images with a size of 256×256, layers 1-6 are used to highlight important features in the image, and layers 7-10 are used for classification.

Based on the results of the work, 4 different data sets were created for 9 classes of 360, 1800, 5400 and 10800 images. The size of the images is 256×256. Before the start of the training, it was decided to conduct training on 10 and 25 epochs to get the best result in time, accuracy and minimum error. The results of network training for each data set are presented in Table 1, where the total neural network training time is depicted.

Dataset description	Epoch	Model	Time, ms	Accuracy	Error
The original set (360 images)	10	1	26	0.4111	1.8603
	10	2	11	0.4778	1.5727
	25	1	**41**	**0.7000**	**1.2075**
	25	2	31	0.4333	1.7432
Data set 1 (1 800 images)	10	1	40	0.5000	**1.9780**
	10	2	23	0.5741	2.3228
	25	1	**81**	**0.7000**	2.6807
	25	2	57	0.6074	2.1436
Data set 2 (5400 images)	10	1	117	0.6756	3.0181
	10	2	104	0.6184	2.8068
	25	1	**293**	**0.7244**	2.8285
	25	2	261	0.6830	**1.8241**
Data set 3 (10 800 images)	10	1	**226**	**0.7604**	2.1762
	10	2	276	0.7263	**1.2362**
	25	1	532	0.7581	2.3874
	25	2	481	0.6396	2.8933

Table 1. Results of experiments with different data sets.

Analyzing the results obtained in Table 1, it can be seen that the second model trains about 30% faster than the first one. However, more accurate results are obtained with the first model. Also, when the model was trained on 25 epochs, the accuracy results are worse than on 10 epochs which indicates excessive training of the neural network. Summarizing Table 1, it can be seen that the best result for the highest accuracy and minimum error was shown by the first neural network training model at 10 training epochs.

Let’s consider a comparison of the accuracy and training error of data sets trained on the first model by 10 epochs.

The CNN shown the best accuracy when it was trained on the fourth dataset with the largest number of augmented images. During the training the error is not always decreased, which may occur due to insufficient training sample size and different quality of the images (Figure 2). Therefore, a further increase in the number of epochs leads to overfitting of the model.

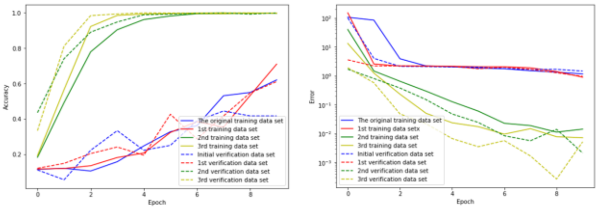


Figure 2. Comparison of accuracy and training error of different data sets on 10 epochs of the first training model.

5. EXPERIMANTAL RESULTS

The experiments were carried out using datasets of natural objects, both available in open sources and independently obtained. To assess the effectiveness, three different network architectures were considered, which were trained on 6 data sets. Let's look at each of the sets in more detail:

− Dataset Stolby (KrasnoyarskStolby Dataset) after scaling up consists of 10800 training images, 2700 validation images and 9 classes that describe natural objects, the images have a complex structure and are obtained in winter conditions;
− Landscape Picture (Landscape Pictures) consists of 3465 training pictures, 860 testing pictures and 7 classes, the set contains images of various quality with one or more key objects, as well as complex backgrounds;
− Intel Image Dataset (Intel Image Dataset) consists of 2406 training images, 598 testing images and 6 classes;
− Wildlife Animals Images (Wildlife Animals Images) consists of 1386 training images, 342 testing images and 6 classes. The dataset typically includes one object in the foreground. The classes are fairly well distributed manually;
− Animals-10 (Animals-10) consists of 20950 training images, 5230 testing images and 10 classes, contains more complex objects and a non-monotonous background, sometimes there are several different categories of objects in the frame;
− Original Stolby set consists of 360 training pictures, 90 testing pictures and 9 classes.

To train the MobileNetV2Base and the Mobile Net V2 Long all images were processed to size 224 by 224. The Stolby set consists of 9 classes such as: elephant, first_pillar, fourth_pillar, lion_rock_gate, pass, rock_feathers, rock_grandfather, rock_granfmother, third_pillar. The Landscape Picture set consists of 7 classes: beach, desert, island, japan. Landscapes, mountain, sea.

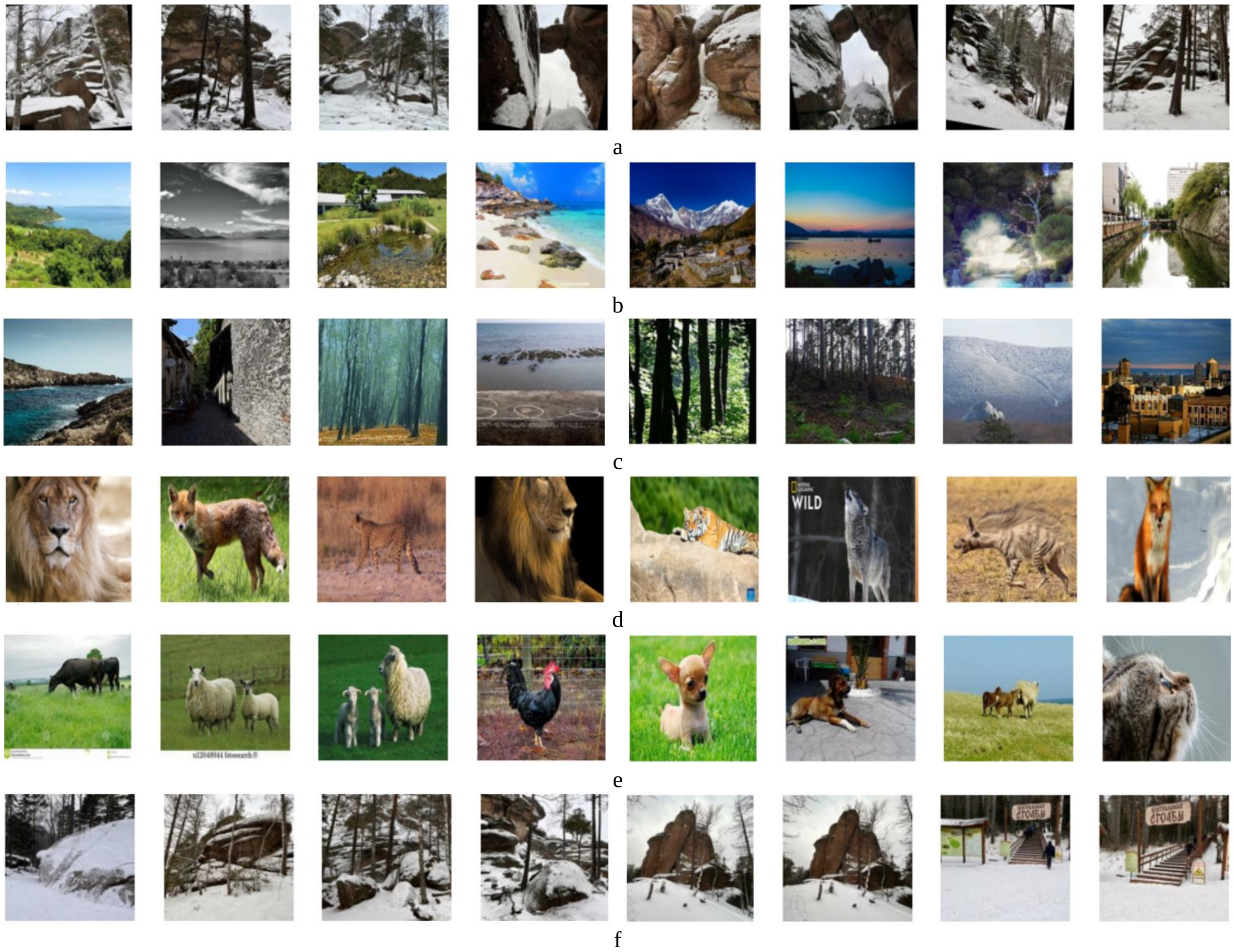


Figure 3. Datasets: a) dataset Stolby after scaling up, b) images from the Landscape Pictures dataset, c) frames from the Intel Image Dataset, d) images from the Wildlife Animals Images dataset, e) frames from the Animals-10 dataset, f) Original Stolby set

Intel Image Dataset consists of: buildings, forest, glacier, mountain, sea, street. Wildlife Animals Images consists of: cheetah, fox, hyena, lion, tiger, wolf. Animals-10 contains 10 classes: cane, cavallo, elefante, farfalle, galina, gatto, mucca, pecora, rango, scoiattolo.

The best of the above was chosen as the first network. The structure of this model is presented in Table 2.

Layer name	Parameters	Stride	Number of layer outputs
Conv 2D	5 × 5	2	16
MaxPooling2D	2 × 2	2	
Conv 2D	5 × 5	2	32
MaxPooling2D	2 × 2	2	
Conv 2D	5 × 5	2	64
MaxPooling2D	2 × 2	2	
Conv 2D	5 × 5	2	128
MaxPooling2D	2 × 2	2	
Flatten()			
Dense			1024
Dropout	0.2		
Dense			256
Dropout	0.2		
Dense			9

Table 2. BaseCNN neural network structure.

The following models were generated based on MobileNet V2. This model has a small weight, which allows us to store the trained model on a mobile device. Even with a small weight, the model has very high accuracy and response time when running on four cores. The Mobile Net V1 model features a deep light that significantly reduces the size of the network model, which is suitable for mobile devices or any devices with low computing power. MobileNet V2 introduces an improved module with an invested residual structure. Nonlinearities in narrow layers are eliminated in this version. In Figure 4, you can see the structure of the MobileNet V1 and MobileNet V2 light blocks.

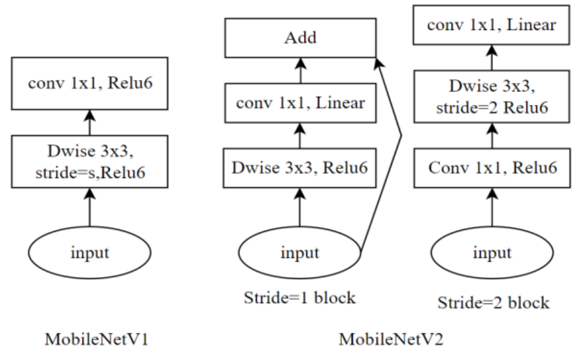


Figure 4. The structure of the MobileNetV1 and MobileNetV2 convolutional blocks.

The structure of MobileNet V2 is presented in table 3.

Input	Operator	*t*	*c*	*n*	*s*
224×224×3	Conv2d	-	32	1	2
122×122×32	Bottleneck	1	16	1	1
112×112×16	Bottleneck	6	24	2	2
56×56×24	Bottleneck	6	32	3	2
28×28×32	Bottleneck	6	64	4	2
14×14×64	Bottleneck	6	96	3	1
14×14×96	Bottleneck	6	160	3	2
7×7×160	Bottleneck	6	320	1	1
7×7×320	Conv2d 1x1	-	1280	1	1
7×7×1280	Avgpool 7x7	-	-	1	-
1×1×1280	Conv2d 1x1	-	k	-	

Table 3. General architecture of Mobile Net V2.

where *t* = the expansion coefficient
c = the number of channels at the output of the unit
n = convolution
s = the convolution step

The MobileNet V2 model from Keras is used as the base model, which is pre-trained on the ImageNet dataset (trained to recognize 1000 classes). This gives us an excellent feature extractor for image classification, and then we can train a new classification layer with the current datasets.

Model training is performed with frozen layers that are used to extract features. This effectively transforms the model into a feature extractor because all the pre-trained weights and offsets are stored in the lower layers when we start training for classification.

Next, we created a new Sequential model and pass the frozen MobileNet model as the main structure of the model, and also added new classification layers to set the final output dimension according to the number of classes in each dataset.

Thus, the MobileNetV2Base models were created, presented in Table 4 and MobileNetV2Long, presented in Table 5, which are able to transfer training only to the last fully connected layer. The results of training BaseCNN, MobileNetV2 and MobileNetV2Long are shown in Figure 4. After that, the model was trained additionally for 5 epochs using fine-tuning, which allows us to train more layers from a pre-trained model. That is, we have unfrozen some layers from the base model and adjusted these weights so that they are better adjusted for the features found in our datasets. The training results are shown in Figure 5.

Layer name	Parameters	Stride	Number of layer outputs
Mobilenetv2_1.00	7 × 7		1280
Conv 2D	5 × 5	2	32
Dropout	0.2		
Global Average Pooling 2D		2	32
Dense			9

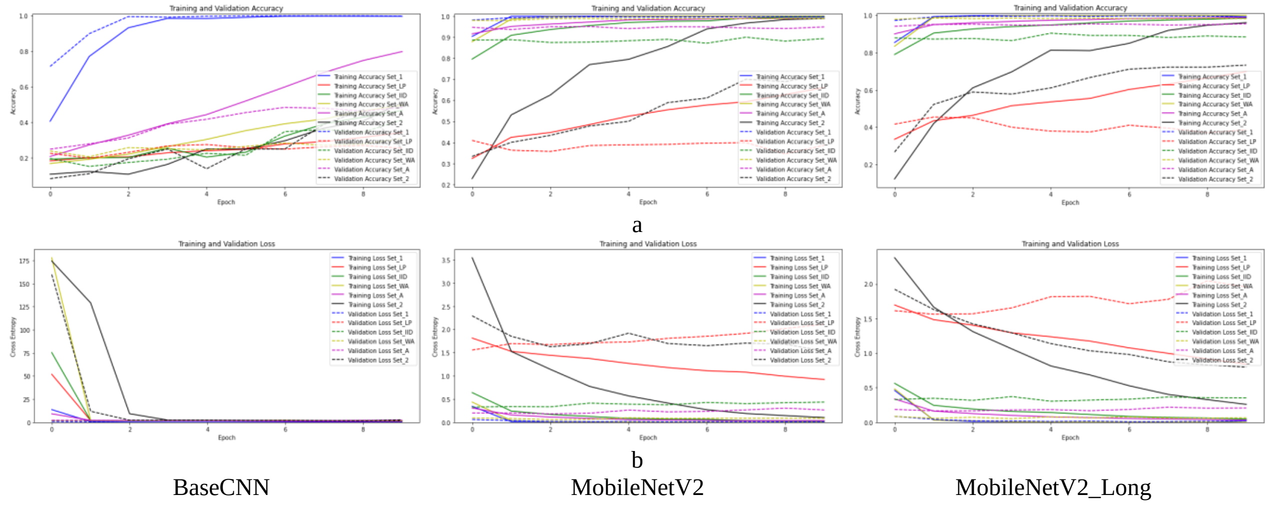
Table 4. MobileNetV2 structure for 9 classes.

Layer name	Parameters	Stride	Number of layer outputs
Mobilenetv2_1.00	7 × 7		1280
Conv 2D	6 × 6	2	32
Dropout	0.2		
Conv 2D	6 × 6	2	32
Dropout	0.2		
Global Average Pooling 2D		2	32
Dense			9

Table 5. MobileNetV2Long structure for 9 classes.

Verification losses are still higher than training losses, so there may be some overfitting during the training process. Retraining may also be due to the fact that the new training set is relatively small with less intra-class variance compared to the original ImageNet dataset that was used for MobileNet V2 training.

Based on the results presented in Table 6, it can be concluded that the best results are shown by the MobileNetV2Base model trained on 10 epochs. Comparing the Original Stolby set and Dataset Stolby after scaling up, high accuracy is observed in a set with data expansion, which is most likely due to the small volume of the original sample and the relatively simple structure of the images. On all training models, there is an increase in accuracy and a decrease in losses on the data set with data expansion, which indicates that we may need additional image preprocessing before training on the most datasets.

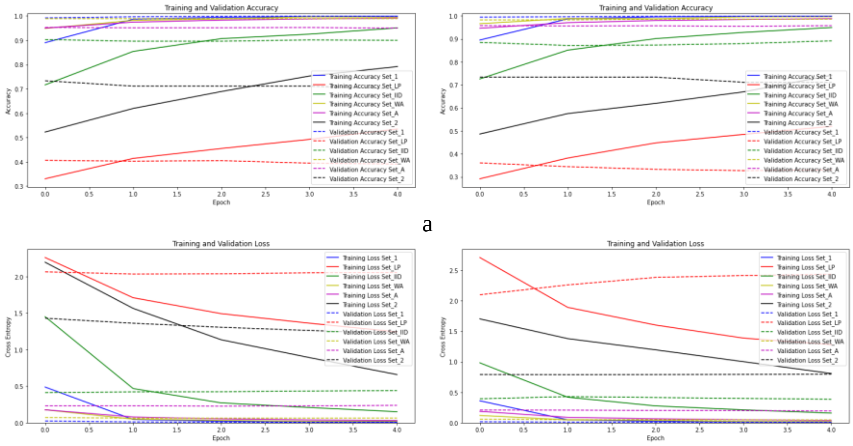


a

b

BaseCNN MobileNetV2 MobileNetV2_Long

Figure 4. Results of different neural networks architectures training on 6 datasets: a) Training and validation accuracy for 3 different models on 10 epochs, b) Training and validation loss for 3 different models on 10 epochs.



a

b

MobileNetV2 MobileNetV2_Long

Figure 5. Results of different neural networks fine-tuning on 6 datasets: a) Training and validation accuracy for 2 different models on 5 epochs, b) Training and validation loss for 2 different models on 5 epochs.

Dataset	Dataset Description	Model	Epoch	Training time, s	Accuracy	Error
Original Stolby set (Set_2)	Training images: 360 Validation images: 90 Classes: 9	BaseCNN	10	1.4	0.5123	1.4845
		MobileNetV2	10	1.6	**0.9889**	0.1012
		Base	5	1.6	0.7917	0.6585
		MobileNetV2	10	1.7	0.9611	**0.2585**
		Long	5	3.6	0.7414	0.8068
Dataset Stolby after scaling up (Set_1)	Training images: 10800 Validation images: 2700 Classes: 9	BaseCNN	10	19.6	0.9969	0.0095
		MobileNetV2	10	40.3	**1.0000**	**0.0002**
		Base	5	21.1	0.9989	0.0074
		MobileNetV2	10	39.9	0.9924	0.0221
		Long	5	239.4	0.9984	0.0081
Landscape Pictures (Set_LP)	Training images: 3465 Validation images: 860 Classes: 7	BaseCNN	10	10	0.3425	1.6240
		MobileNetV2	10	12.5	0.6603	0.9214
		Base	5	7.1	0.5345	1.2424
		MobileNetV2	10	14.1	**0.6985**	**0.8416**
		Long	5	63.4	0.5195	1.2799
Intel Images Dataset (Set_IID)	Training images: 2406 Validation images: 598 Classes: 6	BaseCNN	10	6.9	0.4665	1.3222
		MobileNetV2	10	8.7	**0.9958**	**0.0221**
		Base	5	5.4	0.9509	0.1500
		MobileNetV2	10	8.8	0.9863	0.0449
		Long	5	10.6	0.9492	0.1577
Wildlife Animals Images (Set_WA)	Training images: 1386 Validation images: 342 Classes: 6	BaseCNN	10	7.1	0.4964	1.2685
		MobileNetV2	10	8.9	**1.0000**	**0.0003**
		Base	5	4.5	0.9949	0.0187
		MobileNetV2	10	7.2	**1.0000**	0.0007
		Long	5	313.2	0.9964	0.0169
Animals-10 (Set_A)	Training images: 20950 Validation images: 5230 Classes: 10	BaseCNN	10	41.7	0.7986	0.6011
		MobileNetV2	10	73.1	0.9891	0.0340
		Base	5	37.7	**0.9909**	**0.0306**
		MobileNetV2	10	73.8	0.9893	0.0325
		Long	5	73.8	0.9879	0.0362

Table 6. Results of objects identification with three neural networks on the different natural objects datasets.

6. CONCLUSIONS

We propose a method for expanding the training sample when training a neural network for classifying natural objects of the Stolby National Park of the Krasnoyarsk Territory. Sharpening and noise reduction algorithms were applied to improve the quality of images, as well as geometric transformations to expand the data set. As a result of the expansion of the image database, the number of photos has been increased by 30 times, and the accuracy of object classification using a simple convolutional neural network model on a test sample including 9 classes of objects has increased by one and a half times from 41% to 76%.

The paper proposes an algorithm for expanding the data set for classifying natural objects using geometric transformations, as well as preprocessing filters to improve the quality of images. The efficiency of the algorithm was evaluated using several data sets. As a result, we can say that the proposed approach of expanding the training sample makes it possible to improve the quality of image classification by 10-20% with a slight increase in the time required for network training. Depending on the size of the training sample, the use of the modified MobileNet V2 network structure takes no more than 60 seconds and shows 95-99% accuracy on known datasets. Evaluation the classification algorithm accuracy on existing databases shows high identification speed and resistance to distortion, regardless of the complexity of the image.

REFERENCES

Animals-10. Animal pictures of 10 different categories taken from google images. Available at: https://www.kaggle.com/ datasets/alessiocorrado99/animals10 (12 February 2023).

Artificial neural network, URL: https://dic.academic.ru/dic.nsf /ruwiki/13889.

Bahlmann C., 2005: A system for traffic sign detection, tracking, and recognition using color, shape, and motion information. C. Bahlmann – *Intelligent Vehicles Symposium,* 2005. Proceedings. IEEE. - IEEE, 2005, 255-260.

Boekestijn, J., 2018: Deep learning with data augmentation. *University of Groningen faculty of science and engineerin*, 1-9.

Cisse, M., Dauphin, Y., Lopez-Paz, D., 2018: Mixup: Beyond Empirical Risk Minimization. arXiv:1710.09412v2 [cs.LG].

Devries, T., Taylor, G., 2017: Improved Regularization of Convolutional Neural Networks with Cutout. arXiv:1708. 04552v2 [cs.CV].

Finogeev E. L., Terekhova Yu.V., 2021: Investigation of the dependence of the accuracy of handwritten signature recognition on the language group using deep neural networks. *The second International Scientific and Practical Forum on Economic Security* "VII VSEB" Moscow, April 2021, 97-104.

Gwosdek P., Grewening S., Bruhn A., Weickert J., 2011: Theoretical Foundations of Gaussian Convolution by Extended Box Filtering. *International Conference on Scale Space and Variational Metgods in Computer Vision* 2011, 447-458.

Intel Image Dataset. Building, Forest, Glacier, Mountain, Sea, Street. Available at: https://www.kaggle.com/datasets/hamedetezadi/intel-image-dataset (8 February 2023).

KrasnoyarskStolby Dataset. Available at: https://github.com/ dikrutko/simple-neural-network/tree/main/ 224/Set_1 (08 March 2023).

Landscape Pictures. Datasets of pictures of natural landscapes. Available at: https://www.kaggle.com/datasets/arnaud58/landscape-pictures (8 February 2023).

Mikołajczyk, A., Grochowcki, M., 2018: Data augmentation for improving deep learning in image classification problem. *International Interdisciplinary PhD Workshop (IIPhDW)*, 1-6.

Mustaev A.F., 2019: Application of neural networks in image recognition. *International Scientific Journal "Bulletin of Science*" № 7, 2019, 53-57.

Pattern recognition by neural networks, URL: https://center2m.ru/ai-recognition.

Pillow ImageFilter, URL: https://pillow.readthedocs.io/en/ stable/reference/ImageFilter.html

Simard, P., Steinkraus, D., Platt, J., 2003: Best practices for convolutional neural networks applied to visual document analysis. *Proceedings of the Seventh International Conference on Document Analysis and Recognition, vol. 2. IEEE Computer Society,* 958–958.

Smoothing in Python, URL: https://plotly.com/python/ smoothing/

Supriya L., Rishidas S., 2020 Classification of Natural Scene using Convolution Neural Network. *International Journal of Engineering Research & Technology (IJERT)*, Vol. 9 Issue 07, 1-6.

Tokozume Y., Ushiku Y., Harada T., 2018: Between-class Learning for Image Classification. arXiv:1711.10284v2 [cs.LG].

Wildlife Animals Images. Dataset containing Images of Wildlife Animals. Available at: https://www.kaggle.com/datasets/ anshulmehtakaggl/wildlife-animals-images (12 February 2023).

Wong, S., Gatt, A., Stamatescu, V., 2016: Understanding data augmentation for classification: when to warp? arXiv:1609. 08764v2 [cs.CV].

Zhao W., Du S., Emery W., 2018: Object-Based Convolutional Neural Network for High-Resolution Imagery Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensin*g, 1-12.

Zlobin D. S., 2020: *Text recognition and processing system*. Innovative Science № 11, 26-27.