The investigation of the using the cyclic generative-competitive neural networks for image stylization

Dmitry Ulyanov
Samara NationalResearch University
Samara, Russia
dmitryulyanovhome@gmail.com

Dmitry Savelyev
Samara NationalResearch University;
Image Processing Systems Institute of RAS - Branch of the FSRC "Crystallography and Photonics" RAS
Samara, Russia
dmitrey.savelyev@yandex.ru

***Abstract*—The paper provides examples of convolutional neural network architectures, the corresponding activation functions, and the organization of their interaction in the learning process. Networks interact with each other according to the architecture of generative-adversarial networks. For the task, the NEXET 2017 data set was filtered and formatted. Studies of the architecture of neural networks and varying the volume of the training sample to solve the problem of image styling were carried out.**

Keywords—CycleGAN, NEXET 2017, loss function, convolutional layer

I. INTRODUCTION

The use of artificial neural networks is relevant for a wide variety of applications [1]. In particular, artificial neural networks are used to solve the problem of image recognition [2] and classification of detected objects [3-4].Object detection is successfully used in vehicle tracking, positioning, and surveillance [5].The use of neural networks for solving the problems of medical diagnostics is very promising [6-7].

There are special algorithmsto solve the problem of detecting objects in real time [8], which can be divided into two main families. The first family Region Proposes (the frame regions are alternately proposed and classified) and the second family Single Shot (all objects are immediately detected on the resulting image). The first family includes neural networks such as R-CNN, Fast R-CNN, Faster R-CNN [9-12]. The second family includes YOLO, SSD [5, 11, 13]. In particular, V.S. Gorbatsevich and al. propose original iterative proposal clustering (IPC) algorithm for aggregation of output face proposals formed by CNN and the 2-level “weak pyramid” providing better detection quality on the testing sets containing both small and huge images [14].

The image processing industry is one of the fastest growing in the world. Film studios hire hundreds of designers to meet the deadlines for creating CGI (Computer-Generated Imagery) videos. The work being done is extremely painstaking and of the same type. These properties characterize it as a potential candidate to replace it workers with artificial neural networks (ANNs). Particularly complex and monotonous is the change in the lighting conditions of the images, in particular the transformation of the daytime image into nighttime and vice versa. Until 2014, this problem did not have a solution allowing stylization in a resolution acceptable for the current generation of image formats (1280 × 720 or 1920 × 1080 pixels). In 2014, the architecture of the Generative-Adversarial Networks (GAN) was invented [15].

In 2016, a work was published that proposed a new architecture based on GAN — the cyclic generative-competitive network [16]. For GAN training, it is enough to have many images, generalized for a number of signs. At the same time, as a result of training, 2 stylists are obtained immediately. This is a method that can learn to identify the special characteristics of one set of images and determine how they can be transferred to another set, and all this in the absence of any pair of training examples.

This problem can be more broadly described as styling tasks — transforming an image from one representation of a given scene X to another, Y [17]. Years of research in the field of computer vision, image processing, computer photography and graphics have led to the creation of powerful translation systems in a controlled environment where, for example, pairs of images are available.

However, acquiring paired training data can be complicated and expensive. For example, there is only a pair of data sets for tasks such as semantic segmentation, and they are relatively small [18]. Obtaining input or output pairs for graphical tasks such as artistic styling can be even more difficult because it usually requires artistic development. Therefore, an algorithm is relevant that can learn how to translate between sets without paired examples of input-output. Suppose that there are some basic relationships between sets, for example, that they are two different drawing styles of the same base scene, and we aim to study these relationships. Although we lack control in the form of paired examples, we can use control at the set level: we are given one set of images in region X and another set in region Y. We can train the mapping of X to Y so that the output is indistinguishable from images from Y an adversary trained to distinguish fakes from the original [19].

In this paper, we considered the GAN and cycleGAN architectures. An artificial neural network based on the cycleGAN architecture was programmatically implemented, a set of input data was adapted for training and testing. Also, experiments were conducted to find the optimal network parameters for solving the stylization problem, in comparison with the reference result of one of the cycleGAN implementations.

The research has shown the possibility of using the architecture of a cyclic generative-adversarial network to solve the problem of styling images for day and night lighting conditions. The relevance of this work is due to the need of the film industry to simplify the process of shooting video by replacing the post-processing of a number of effects recreated on stage. In particular, one of the most problematic effects is shooting in low light conditions or night shooting. Equipment for night shooting is expensive, which makes the need for styling daytime images into nighttime images high.

II. ARCHITECTURE AND LOSS FUNCTIONS FOR NEURAL NETWORKS

According to the GAN architecture, two networks are involved in the learning process - the generator and the

discriminator [20]. The generator may be represented as 3 blocks: a feature extraction unit, a feature conversion unit and data recovery based on the features[21].In this work, the following layers are used: convolution (conv), residual block (ResNet) and transposed convolution block (TrConv). The convolutional layer selects features[22]. The ResNet block consists of two connected convolutional layers. The need for ResNet blocks arises when the task involves a large number of layers, and it, in turn, negatively affects the ability of the network to learn, and paradoxical cases are possible when a neural network with fewer parameters achieves a better result compared to its multilayer counterpart. This problem is called the degradation of the ANN. The addition of layers is possible due to the fact that the dimension of the output and input of the block must coincide. It is also optionally possible to add an activation function at the exit from the block. The transposed convolution layer on the basis of the characteristics applied to the input reproduces data possessing these characteristics[23].The architecture of the generators and discriminators are shown in Table 1 and Table 2.

TABLE I. GENERATOR ARCHITECTURE

Layer type	Input layer size	Kernel size	Distancebetweenconvolutions	Activation function
Conv	256×256×3	3×3	2	ReLU
Conv	128×128×32	3×3	2	ReLU
Conv	64×64×64	3×3	2	ReLU
Conv	32×32×128	3×3	2	ReLU
ResNet	32×32×128	3×3	1	ReLU
ResNet	32×32×128	3×3	1	ReLU
ResNet	32×32×128	3×3	1	ReLU
ResNet	32×32×128	3×3	1	ReLU
ResNet	32×32×128	3×3	1	ReLU
TrConv	32×32×128	3×3	2	ReLU
TrConv	64×64×64	3×3	2	ReLU
TrConv	128×128×32	3×3	2	ReLU
Conv	256×256×16	3×3	1	ReLU
Out	256×256×3	-	-	-

TABLE II. DISCRIMINATOR ARCHITECTURE

Layer type	Input layer size	Kernel size	Distancebetweenconvolutions	Activation function
Conv	256×256×3	4×4	4	ReLU
Conv	64×64×32	4×4	4	ReLU
Conv	16×16×64	4×4	4	ReLU
Conv	4×4×128	4×4	4	Softmax
Out	1×1	-	-	-

Due to the extreme computational complexity caused by the characteristics of the test system, the error functions take the following form:

$$D_A^{loss} = (D_A(a) - 1)^2 + D_A(G_{B\to A}(b))^2, \quad (1)$$

$$D_B^{loss} = (D_B(b) - 1)^2 + D_B(G_{A\to B}(b))^2, \quad (2)$$

where $D_A(a)$ and $D_B(b)$ are functions of discriminators of pictures of classes A and B, respectively.

Let's introduce the cyclic error function for transition from one class to another:

$$C_A^{loss} = \frac{1}{3\times L\times Z}\sum_{i=1}^{L}\sum_{j=1}^{Z}\sum_{k=1}^{3}\left|a_{ijk} - \tilde{a}_{ijk}\right|, \quad (3)$$

where a_{ijk} and $\tilde{a}_{ijk}$ are pixels from i-row, j-column, k-color channel of image a, that applied to generator $G_{A\to B}$ and $\tilde{a} = G_{B\to A}(G_{A\to B}(a))$ respectively.

The general function of the cyclic loss, taking into account (3), will take the following form:

$$C^{loss} = \frac{1}{2}(C_A^{loss} + C_B^{loss}) \quad (4)$$

The resulting loss functions of the generators, taking into account (4), take the following form:

$$G_{A\to B}^{loss} = (D_B(G_{A\to B}(a)) - 1)^2 + \lambda C^{loss} \quad (5)$$

$$G_{B\to A}^{loss} = (D_A(G_{B\to A}(b)) - 1)^2 + \lambda C^{loss} \quad (6)$$

III. TRAINING DATA FOR THE NEURAL NETWORK

For training the neural network, a part of the NEXET 2017 database was selected. This database consists of photographs from auto-registrars in the resolution of 1280×720. The dataset contains 17000 images. To divide into day and night sets, 200 images were manually selected (100 for each class), and the brightness of each of them was calculated using the following brightness formula from the HSP model [24]:

$$L = \sqrt{0.299\times r^2 + 0.587\times g^2 + 0.114\times b^2}, \quad (7)$$

where r, g and b are the discrete values of the RGB components of the pixel in the range [0; 255]. Thereafter all the values were normalized by dividing by the maximum value. The results are shown in Figure 1.

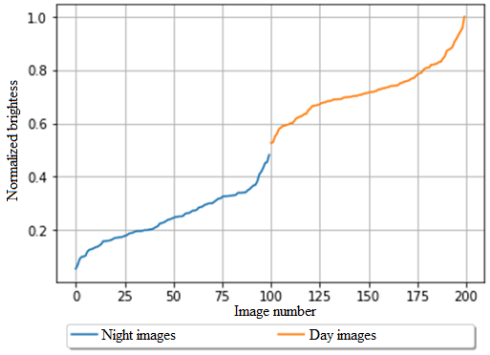

Fig. 1. Graph of normalized image brightness.

Values of normalized brightness were sorted so that the brightness jump was representable. The figure shows the gap between the graphs on the interval of normalized brightness [0.48; 0.52], from which it can be assumed that approximately in this interval there are images that cannot be classified as day or night. Due to the fact that with such a small separation interval, the probability of a false determination is high, we expand this interval to [0.4; 0.6]. We will divide it into day and night sets according to the following rule: if you can set the border of normalized brightness above which images can be classified as daylight, and below, respectively, as nighttime. After classifying the entire database according to this rule, 4,695 night and 11,442 day images were obtained, of which 500 random images per class were selected for the test sample. Due to the extremely limited resources for such tasks for neural

networks, each image was cropped to a 1:1 aspect ratio and scaled to 256×256 pixels.

IV. Conducting Experimental Research and Analysis of the Results

- CPU —intel core i5-2500, 3.3 GHz.
- RAM—8Gb, 1333 MHz.
- GPU —Nvidia GTX 1060, 6Gb GDDR5.
- OS —Windows 10 x64. Ver. 1903.
- Language—Python 3.7.1 x64.

A program was written that implements algorithms for training ANNs and styling images using the designed model using the following libraries:

- Pillow.
- Tensorflow-gpu 1.13.1.

A neural network has the following parameters:

- Learning rate α—2×10^{-4} for all networks.
- Received image size— 256×256×3.
- Neuron activation function—ReLU.
- Cyclic loss coefficient λ— 10.

The size of the training set is 1000 elements. The size of the test sample is 100 elements. The number of eras is 100. A graph of the errors of discriminators and generators is shown in Figure 2.

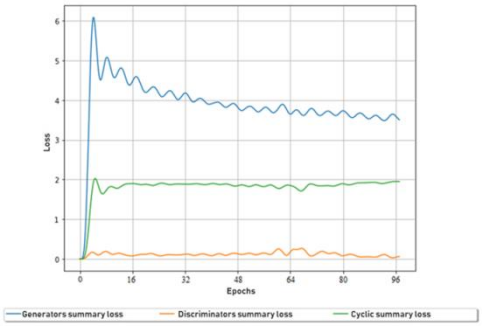

Fig. 2. Error graphs of an undersampled network.

The results of the conversion of test images are shown in Figure 3.

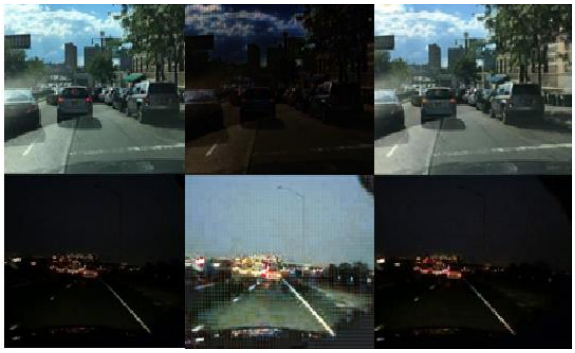

Fig. 3. Image styling result.

Experiments have shown that the conversion from day to night does not always work correctly - only the brightness of some elements of the scene increases, while the sky does not stylize. Increase the number of training examples to 10,000 daytime and 4,500 nighttime. 500 images from each class are used as test cases. Due to the sharply increased data volume, we reduce the number of epochs to 20. The graph of the errors of discriminators and generators is shown in Figure 4.

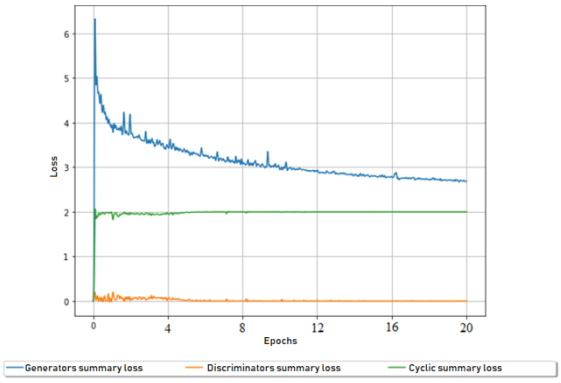

Fig. 4. Error graphs of a network trained on a full sample.

The result of the conversion of test images is shown in Figure 5.

Fig. 5. The result of styling the image while increasing the number of training examples.

We will change the learning speed by several orders of magnitude in order to find out whether the learning speed standard for most tasks is $\alpha = 2\times10^{-4}$acceptable for this task. The next experiment has been done on the same test image the result of processing neural networks trained on an incomplete (1000 images in each class) data set with different learning speed indicators: 2×10^{-4}, 2×10^{-3} and 2×10^{-2}. The number of epochs is 20. The result of the experiment is shown in Figure 6.

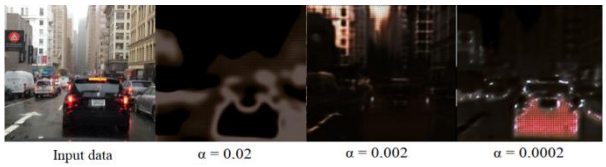

Fig. 6. The dependence of the result on the learning speed.

Comparing Figures 3 and 5, we can notice that the neural network obtained as a result of the first experiment either does not substitute the sign of light sources at night, or does not fully decode it, since the manifestations of this sign are noticeable in Figure 5, that indicates the need for a large number of eras for this styling task.

V. Conclusion

In this research, a software package was developed to demonstrate the operability of the cycleGAN architecture in

image styling tasks. Training and test samples from the NEXET 2017 data set were generated.The studies with the designed software package have shown the possibility of using the architecture of a cyclic generative-adversarial network to solve the problem of styling images for day and night lighting conditions. The solution to this problem is relevant in the film industry for creating CGI-video.

In the course of the work, the following tasks were solved: the cycleGAN architecture was implemented, a database for training and testing was formed, the ANN was trained on a complete and incomplete set of training data.

Studies have shown that to solve the problem of styling images for day and night styles using ANNs, one should maximize the number of unique elements of a training sample. This allows you to reduce the result of the sum of loss functions by 25% with fewer eras, which indicates an improvement in the quality of stylization. Also shown that for this problem in the selected ANN configuration, the optimal learning rate is 2×10^{-4}.

REFERENCES

[1] P.J. Lisboa and A.F.G. Taktak,"The use of artificial neural networks in decision support in cancer: a systematic review," Neural networks,vol. 19, pp. 408-415, 2006.

[2] J. Zhang, K. Shao and X. Luo, "Small sample image recognition using improved convolutional neural network," Journal of Visual Communication and Image Representation, vol. 55, pp. 640-647, 2018.

[3] R. Magdeev and A.G. Tashlinskii, "Efficiency of object identification for binary images," Computer Optics, vol.43, no. 2, pp. 277-281, 2019. DOI: 10.18287/2412-6179-2019-43-2-277-281.

[4] M.A. Isayev and D.A. Savelyev, "Investigation of optimal configurations of a convolutional neural network for the identification of objects in real-time," CEUR Workshop Proceedings, vol. 2416, pp. 417-423, 2019.

[5] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779-788, 2016.

[6] A. Qayyum, S.M. Anwar, M. Awais and M. Majid, "Medical image retrieval using deep convolutional neural network," Neurocomputing, vol. 266, pp. 8-20, 2017.

[7] Yu.D. Agafonova, A.V. Gaidel, P.M. Zelter and A.V. Kapishnikov, "Efficiency of machine learning algorithms and convolutional neural network for detection of pathological changes in MR images of the brain," Computer Optics,vol. 44, no. 2, pp. 266-273, 2020. DOI: 10.18287/2412-6179-CO-671.

[8] J. Dai, Y. Li, K. He and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," Advances in neural information processing systems, pp. 379-387, 2016.

[9] R. Girshick, "Fast r-cnn," Proceedings of the IEEE international conference on computer vision, pp. 1440-1448, 2015.

[10] S. Ren, K. He, R. Girshick and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," Advances in neural information processing systems, pp. 91-99, 2015.

[11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu and A.C. Berg, "Ssd: Single shot multibox detector," European conference on computer vision, pp. 21-37, 2016.

[12] Yu.V. Vizilter, V.S. Gorbatsevich and S.Y. Zheltov, "Structure-functional analysis and synthesis of deep convolutional neural networks," Computer Optics, vol. 43, no. 5, pp. 886-900, 2019. DOI: 10.18287/2412-6179-2019-43-5-886-900.

[13] R.P. Bohush and I.Y.Zakharava, "Person tracking algorithm based on convolutional neural network for indoor video surveillance," Computer Optics, vol. 44, no. 1, pp. 109-116, 2020. DOI: 10.18287/ 2412-6179-CO-565.

[14] V.S. Gorbatsevich, A.S. Moiseenko and Y.V. Vizilter, "FaceDetectNet: Face detection via fully-convolutional network," Computer Optics, vol. 43, no. 1, pp. 63-71, 2019. DOI: 10.18287/ 2412-6179-2019-43-1-63-71.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, "Generative adversarial nets", Advances in neural information processing systems, pp. 2672-2680, 2014.

[16] J.Y. Zhu,T. Park, P. Isola, A.A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," Proceedings of the IEEE international conference on computer vision, pp. 2223-2232, 2017.

[17] T. Rashid, "Make Your Own Neural NetworkMake Your Own Neural Network," CreateSpace Independent Publishing Platform, 2016.

[18] S. Naoki, "Up-sampling with Transposed Convolution," Medium, 2017.

[19] B. Li and Y. He, "An improved ResNet based on the adjustable shortcut connections," IEEE Access, vol. 6, pp. 18967-18974, 2018.

[20] X. Liu, Z. Deng and Y. Yang,"Recent progress in semantic image segmentation," Artificial Intelligence Review, vol. 52, no. 2, pp. 1089-1106, 2019.

[21] S. Saha, "A comprehensive guide to convolutional neural networks—the ELI5 way," Towards Data Science, vol. 15, 2018.

[22] K. Men, X. Chen, Y. Zhang, T. Zhang, J. Dai, J. Yi and Y. Li, "Deep deconvolutional neural network for target segmentation of nasopharyngeal cancer in planning computed tomography images," Frontiers in oncology, vol. 7, pp. 315-315, 2014.

[23] J. Fu, J. Liu, Y. Wang, J. Zhou, C. Wang and H. Lu, "Stacked Deconvolutional Network for Semantic Segmentation," IEEE Transactions on Image Processing, 2019.

[24] D.R. Finley, "HSP Color Model – Alternative to HSV (HSB) and HSL," 2006.