Blind Deconvolution for Restoring Blurred and Distorted Images Based on the Pyramid Approach and Genetic Algorithms

Nikita Medvedkov [1] and Andrey Trubakov [1]

[1] *Bryansk State Technical University, 7, 50 let Oktyabrya blvd., Bryansk, 241035, Russia*

Abstract
Automatic restoration of blurred and distorted images is one of the most urgent tasks of image processing. There are two main classes of methods for solving this problem – deep learning and optimization methods. Optimization methods are a classic way to solve this problem and are aimed at selecting an unknown distorting function. This task is called blind deconvolution. The advantage of these methods is their accuracy and quality of the result obtained. However, a significant disadvantage of these methods is their operation time, which can reach more than a dozen minutes. In most methods of this class, classical gradient optimization methods and their modifications are used. However, this approach has a number of disadvantages. One of them is that the objective function must be differentiable. Another significant disadvantage of these methods is the probability of stopping at a local extremum and not finding the optimal solution. Thus, the disadvantages of the step-by-step optimization methods used are added to the long operation time of these methods. Optimization models devoid of these disadvantages (except for optimization time) are genetic algorithms. In this paper, we propose a method of blind deconvolution based on a pyramid approach and the use of a genetic algorithm with a modified mutation operator as an optimization model.

Keywords
Blind deconvolution, image restoration, genetic algorithms, artificial intelligence

1. Introduction

Availability and a large number of digital cameras for consumers has led to a sharp increase in the number of bitmaps. Along with the increase in the amount of graphic information, image processing methods have actively begun to develop.

One of the most interesting and complex tasks of image processing is the restoration of blurred and distorted images. This problem is especially relevant for the field of consumer photography, which uses relatively small and light cameras. Moving or shaking of the camera during exposure leads to a blurred image, and an incorrectly selected focal length leads to a distorted image. At the same time, consumer photography is just one of the many areas in which this problem is relevant.

Image restoration itself is of interest because of improving the visual quality of the image. However, it is much more important to restore the lost information on the image, which may be impossible to capture again, or very expensive for some reason.

Currently, there are two main classes of methods for solving this nontrivial problem. They are deep learning and optimization methods. The difference between optimization methods is that these methods are aimed at finding an unknown distorting function, which is used to restore a distorted image. This statement of the problem is called blind deconvolution [1]. Despite a very long operation time of these algorithms, they allow to obtain quite accurate results of reasonable quality unlike neural networks, which result directly depends on the trainable fetch [2]. For example, the result of restoration using a generative-adversarial neural network may be visually clear, but it may be unacceptable, since it may contain details that were not present in the original image, and in some cases the result is not the result

of restoration at all, but the result of stylization. The result of image restoration using pre-trained DeblurGAN model [3] can serve as an example (see Fig. 1, photo of Pablo Picasso).

Figure 1: An example of image restoration by means of DeblurGAN model

An important moment in the development of automatic image restoration can be considered the work of Rob Fergus [4], in which a method using a pyramid approach to search for a distorting function was presented. A large number of modern methods of blind deconvolution are based on this approach.

The feature of these approaches is gradient optimization methods [5, 6, 7], which use imposes a restriction on the objective function: it must be differentiable. In addition, these methods tend to get stuck in local extremum, which negatively affects the resulting optimization result.

These disadvantages can be avoided by using genetic algorithms by optimizing immediately for a certain set of points (population) of the search space and modeling crossing and mutation processes. The latter makes it especially effective to deal with situations of falling into a local extremum. In addition to these advantages, genetic algorithms do not use any other information about the objective function except for the objective function itself and its range of values, that is, there is no differentiation in the optimization process. These advantages make genetic algorithms very attractive for use in multidimensional optimization problems, in which the optimized function may not be differentiable or has many local extrema [8].

In this paper, we propose an algorithm for solving the problem of blind deconvolution, which uses a pyramid approach, but in which the optimization model is a genetic algorithm with a mutation operator specific to this problem.

2. Assumptions and constrains made in the study

In this paper it was decided for doing the research to focus on the design and development of a genetic algorithm and its recombination operators. At this stage, it was decided not to consider the problem of choosing and analyzing in detail the quality metric of the restored images, which is the objective optimization function. This issue requires additional study in the future. So far, instead of choosing a classical metric based on comparing the histogram of gradients with the "heavy-tailed" distribution, characteristic of clear images [9, 10], the study considers an idealized version in which a clear image is known, with which the restoration result is compared using one of the metrics that allow evaluating the similarity of two images. Such an image can be obtained in the future either by modeling edges, or using generative networks.

However, as experiments have shown, the similarity between the images alone is not enough. The objective function should contain information about how clear the image we get. Otherwise, the algorithm stops at the distortion function, which is a point of several pixels.

Also, for the purpose described above, artificially distorted images were used in the experiments without subsequent noise overlay. This is a rather great constrain of the proposed method at the current stage, but a preliminary analysis allows to conclude that this approach is also well suited for noisy images.

3. Distortion model used in the study

The paper considers a classical distortion model, which can be described by the following expression [4]:

$$g(x, y) = f(x, y) \times h(x, y) + n(x, y),$$

where f(x,y) is the original undistorted image; h(x,y) is a distorting function; n(x,y) is additive noise; $\times$ is a convolution operator; g(x,y) is the distortion result (blurred and distorted image).

The formula above shows the distorting function as a usual convolution kernel, which is unknown. It determines how the image is distorted and the restoration result depends on how accurately it can be collated. In the case of linear blur, the distortion function may look like in Fig. 2.
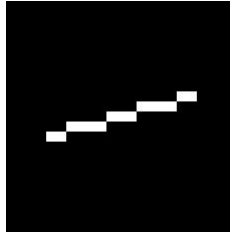

Figure 2: Linear distortion function

The goal of optimization is to find an unknown distortion function, which is subsequently transmitted to one of the inverse filtering algorithms, for example, Wiener filter or the Richardson-Lucy algorithm. After that, filtering result is evaluated using the objective function. That is, the essence of the task is to find such a convolution kernel, when using it in inverse filtering, the best result would be obtained considering quality. In other words, the objective function is maximized, which unknown variable is the distortion function.

4. The algorithm proposed

The algorithm structure repeats the structure of the algorithm described in the paper by Rob Fergus, with the exception that an assumption was made regarding the objective function: the restoration result is evaluated by a combination of two metrics: a metric that evaluates image sharpness and a metric that evaluates the similarity of the result obtained with the original, and its being known is also an assumption. In addition, the developed genetic algorithm is used as an optimization model, which will be described in this section.

4.1. Pyramid approach

The pyramid approach consists in gradual refining the kernel from the smallest size to the maximum size specified by the user. Searching for the convolution kernel begins, as a rule, with a resolution of 3x3 or 5x5. At the same time, in the process of optimizing the kernel of a certain size, a distorted image is used not of the original but of reduced dimensions. For this purpose, a pyramid of distorted images of different dimensions is built. Each of these images corresponds to a certain size of the convolution kernel. In the proposed algorithm, the kernel increases smoothly: the height and width of the kernel increases by 2 at each level of the pyramid. Accordingly, the maximum size of the kernel that the user sets should be odd.

For example, the maximum kernel size is 13, and let the image being restored has a resolution of 500x500, in this case, the pyramid in the proposed algorithm will look like in Table 1.

Table 1
Resolutions of distorted images in the pyramid depending on the kernel size

Kernel size	Image resolution in the pyramid
13	500x500
11	423x423
9	346x346
7	269x269
5	192x192
3	115x115

Accordingly, the algorithm starts from the lowest level of the pyramid (3x3 kernel resolution). 3x3 resolution kernel is optimized on it. After that, the algorithm increases the kernel to the size specified at the next level, and optimization is performed again, but with the resulting kernel size at the next level. Moving through the pyramid makes it possible to refine the kernel at each step, rather than collating it again, which significantly saves time and resources when the algorithm is running.

When designing, it was decided to make the increase in the kernel resolution smooth (by 2 in width and height for each level of the pyramid). The experiment showed that a sharp increase of the kernel adversely affects the optimization result. It should be noted that the kernels at the initial levels of the pyramid are initialized by horizontal lines of maximum brightness.

4.2.   Genetic algorithm

As it was mentioned above, at each level of the pyramid, the kernel of the appropriate size is optimized. Genetic algorithms operate with the concept of an individual - a point in the search space [11], which in this case is the kernel, that is, a square matrix. In the classical formulation, it is customary to work with each coordinate point of the search space in encoded form. In this case, the individual will be a chromosome - a sequence of genes. However, in the proposed algorithm, it was decided to abandon this idea and work with the individual phenotype that is, in an uncoded form. At the same time, it is proposed to work with the uncoded parameters of an individual as with genes. In this regard, it is worth noting that further, for the convenience of presentation, the individual will be called a kernel, and the uncoded parameter of the individual will be denoted by a pixel. It is also worth mentioning that the area of determining the brightness of the kernel pixels is a segment [0; 1]. All processed images are represented from 0 to 1.

4.2.1. Selection

We use tournament selection as the selection operator. This is done because this type of selection allows to adjust the diversity of the parent pool using the size of the tournament. The larger the size of the tournament, the more individuals with high fitness will be in the parent pool. The smaller the size of the tournament, the more chances for individuals with low fitness to get into the parent pool. This makes the tournament more diverse [11]. In the proposed algorithm, it is recommended to use a tournament size equal to two.

4.2.2. Crossover

The proposed algorithm uses a classical uniform crossover without any modifications. It was decided to use this type of crossover because of the peculiarities of individual representations. An individual is not a one-dimensional array, but a two-dimensional one. This is not just a set of variable values, but also an image of some continuous function. Therefore, we decided to abandon the operators of single-point or two-point crossover, since transferring an image section to another image section for this task will be extremely inefficient.

4.2.3. Mutation

The desired convolution kernel, as a rule, is an image with most of the pixels having brightness of 0. At the same time, only a small number of pixels in the center of the image have brightness greater than zero. The experiments showed that classical mutation operators, such as bit inversion, exchange mutation, or reversal mutation, do not produce a positive effect on the optimization process, since mutations in this case occur throughout the kernel, and the number of useful mutations is significantly reduced. This is manifested primarily in the change of pixels along the edges of the kernel, which in most cases should be equal to zero. This mutation takes the optimization process aside and increases the running time of the algorithm. In the course of the study, it was revealed that if mutations occur near the distortion function found in the previous iteration (pixels with high values), then the number of useful mutations becomes much larger. Based on the observational data, it was decided to produce mutations near pixels with high values. In the proposed algorithm, the high value is the value from the segment [0.1;1.0]. Thus, the number of pixels for mutation is greatly reduced.

It should also be noted that mutation occurs not with the pixel itself, but with the pixel located near it. In the proposed algorithm, this idea is implemented by selecting a random pixel from a square area, the center of which is the selected pixel with a high value. And the area has a size of 4x4. However, the value of this parameter is not fixed and we can experiment with it.

Let us consider the mutation using the example of an intermediate optimization result presented in Fig. 3. If the pixel marked with a red dot was selected among pixels with a high value, then only 1 of random pixels from the square area around the central pixel can mutate.
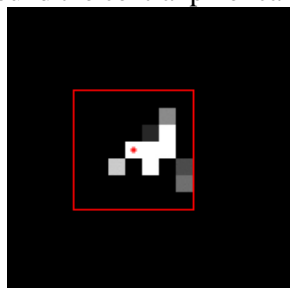

Figure 3: An example of mutation area

As it was mentioned above the work with individuals is carried out in an uncoded form, and inverting operation in this case loses its meaning. It is also should be stressed that changing the value to the value of the opposite segment edge of the detection area is not effective in this case, since most of the real distortion functions are not binary images and may have intermediate values of brightness. Therefore, the mutation operations in the algorithm are random positive or negative change of the pixel value to a random value from the definition area (a real number from 0 to 1). In this case, the probability of a positive mutation is set separately, and, as experiments have shown, the optimal value of this probability is the value from the segment [0.5, 0.9].

4.2.4. Elitism

In the proposed algorithm, it is recommended to use 1 elite individual to preserve the optimal solution. Thus, no time is wasted on re-searching for the solution already found [9]. A large number of elite individuals during the study did not have a positive effect on the results of the search for solutions.

Also, the author's approach without the use of elite individuals showed itself as quite positive. In this case, the best individual from each population was saved in a separate pool, and after the optimization for the current pyramid level was completed (before moving to a new level of the pyramid with a large kernel), the individuals were replaced with an average version of the best individuals from the pool above. But this option is worse than the option with 1 elite individual, although the probability of getting to the local maximum in this case is much lower.

4.3. Algorithm description

Let us consider a general optimization algorithm using genetic algorithms. The block diagram of the proposed algorithm is presented in Fig. 4.
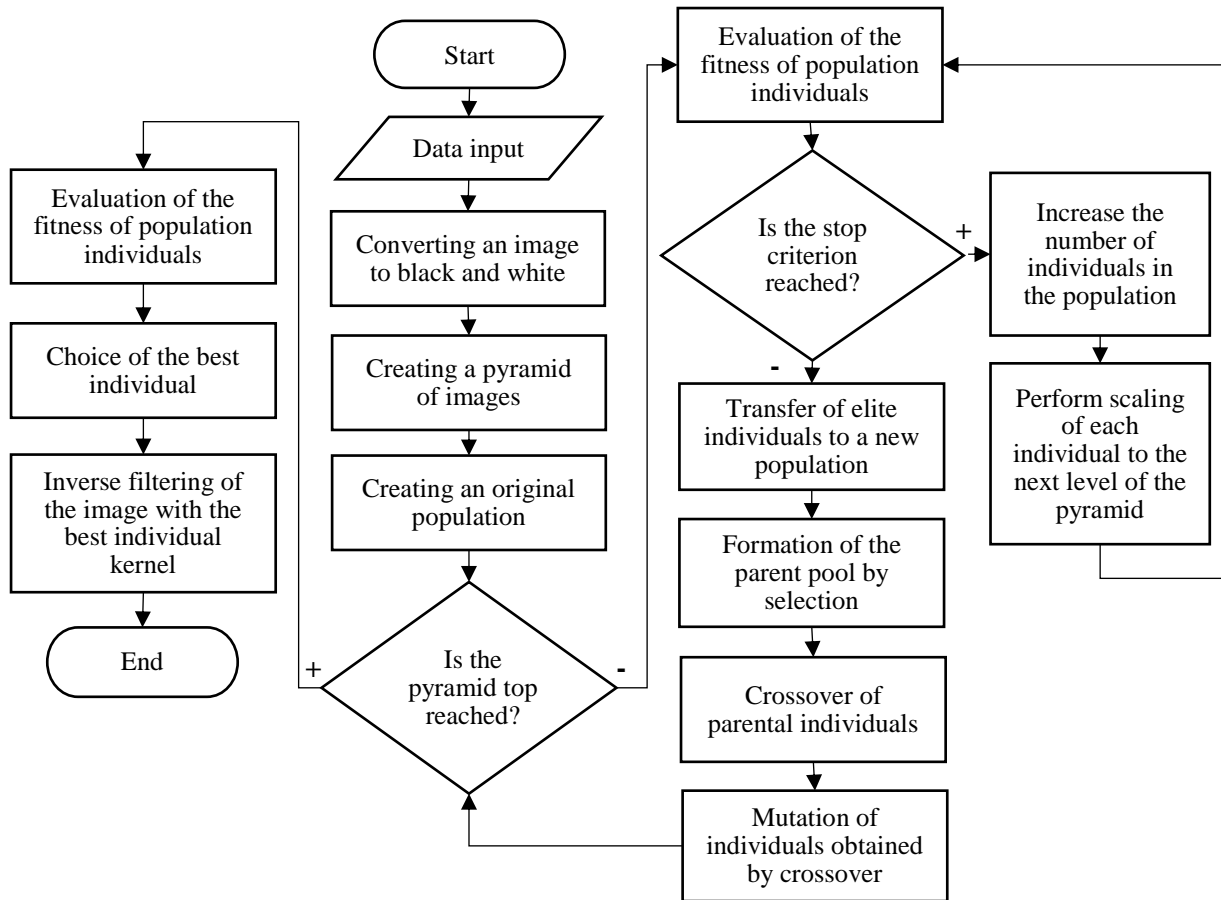

Figure 4: Algorithm block diagram

In the above block diagram, data input step has a special meaning. Input data mean:
- distorted image;
- probability of crossover;
- maximum kernel size;
- probability of mutation;
- probability of a positive change in the pixel value during mutation;
- a number of populations during which there is no increase in the objective function (the stop criterion).

We have made a decision to increase the number of individuals in the population when moving to a new level of the pyramid. The experiments showed that with an increase in the kernel size, this aspect significantly affects the optimization result. However, it also has a negative side – at the same time, it significantly reduces the speed of optimization.

5. Experimental verification of the method

For the experiments, a collection of small resolution images (not exceeding 1000x1000 pixels) was compiled. All the images in the collection were artificially distorted using pre-prepared distortion functions. As it was mentioned above, filtering result was not subjected to further noise overlay.

For inverse filtering, Wiener filter was used with the constant value describing the signal-to-noise ratio equal to 1. Despite the fact that the prepared image was not made noisy, it was decided to use this

filter, because it allows to adjust the sharpness of the result with the help of the above constant, changing which in some cases may be useful. So, the value of the constant 1 showed itself best when evaluating the sharpness of the filtering result using the metric, which will be discussed below. It should be noted that this value of the constant was used only when evaluating the fitness of individuals. The final result of restoration was obtained when using the value of this constant equal to 0.01.

As it was mentioned above, the objective function was represented by a function, which is a combination of two metrics: a metric that evaluates the similarity of the restoration result with the original and a metric that evaluates the sharpness of the result. The structural similarity metric SSIM [10] was used as a similarity metric, and the metric that evaluates how sharp the image is obtained was performed by a metric that represents the average value of Fourier transform:

$$Q_{Fourier}(I) = \frac{\sum_{j=1}^{n} \sum_{i=1}^{m} DFT(I)(i,j)}{m * n}$$

where I is the evaluated image resolution $m \times n$; i, j are pixel coordinates; $DFT(I)$ is the Fourier transform of I image; $\times$ is the convolution operator; g (x, y) is the result of distortion (blurred or distorted image).

Thus, the objective function can be described by the following expression:

$$Q(Sharp, Restored) = SSIM(Sharp, Restored) + \log(Q_{Fourier}(Restored)),$$

where $Sharp$ is a sharp image; $Restored$ is the result of restoration.

Let us consider 2 experiments conducted with images of different types (each image was distorted by different functions). All experiments were performed with the following parameter values:

- probability of crossover: 0.1;
- tournament size: 2;
- maximum kernel size: 23;
- probability of mutation: 0.1;
- probability of a positive change in the pixel value during mutation: 0.5;
- the number of populations during which there is no increase in the objective function (stop criterion): 15.

The original sharp image, distorting function, as well as distortion result are presented in Fig. 5.
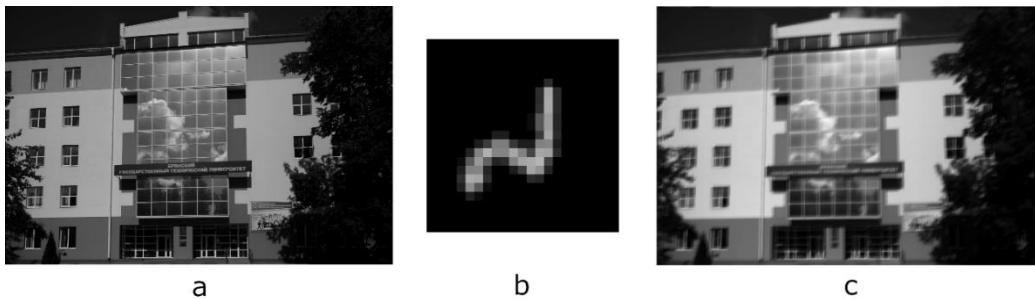

a   b   c

Figure 5: a – original image; b – distorting function; c – distortion result.

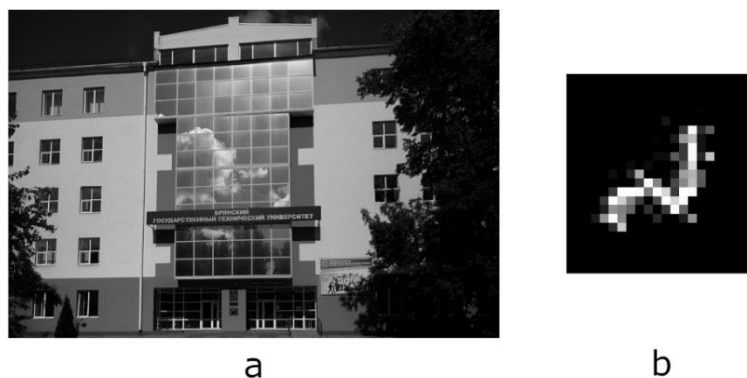Restoration result as well as distortion function are presented in Fig. 6.

a   b

Figure 6: a – restoration result; b – distortion function found during optimization.

As we can see, the distorting function obtained has a very great similarity in the form with the real distortion function used while preparing the initial data. However, it can also be noticed that the function obtained is much brighter than the actual one, and also has standalone pixels. These results can be avoided by further conducting additional experiments with the evaluation metric, which is the basis for the optimization function.

Let us consider another experiment in which the method gave not the best, but an acceptable result. Another distortion function was chosen for it, and the image itself contains a larger number of small details. The original sharp image, distorting function, as well as distortion result are presented in Fig. 7. Restoration result and distortion function obtained are presented in Fig. 8.
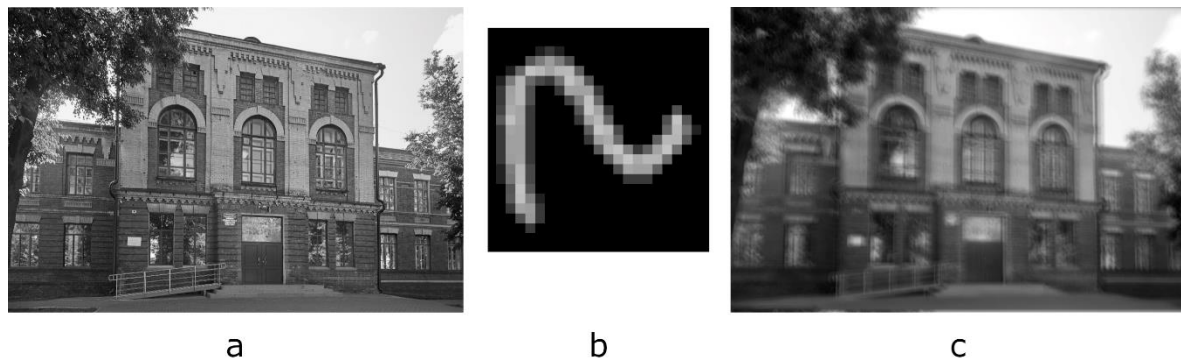

a   b   c

Figure 7: a – original image; b – distorting function; c – distortion result.

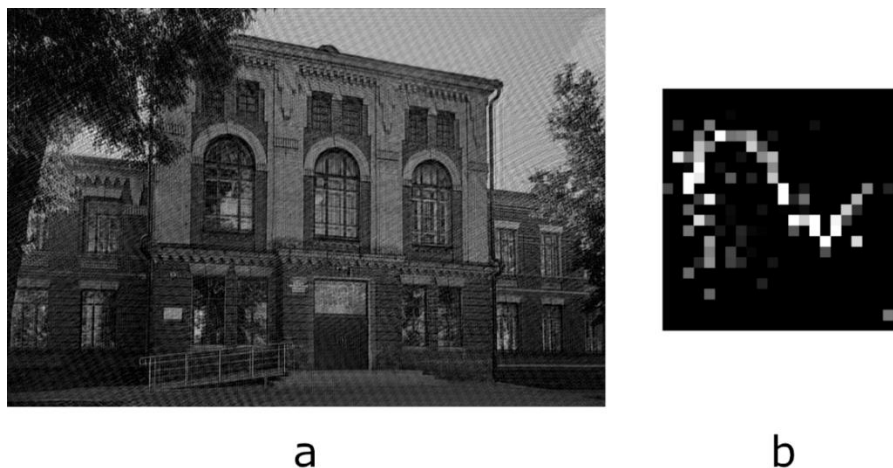a   b

Figure 8: a – restoration result; b – distorting function obtained during optimization.

It can be noted that the result is not perfect, but the distortion function obtained quite accurately describes the shape of the true distortion function. This result has all the disadvantages of the result of the first experiment: the brightness of the pixels is much higher, there are some bright pixels.

6. Conclusion

The paper presents a new approach to restoration of blurred and distorted images, based on the use of a pyramid of images and genetic algorithms for the optimization problem and kernel search. This approach still requires additional research and analysis (especially in the field of metrics for evaluating the quality of the restored result). However, even in the variant described above, the recovery result is quite good compared to other classical methods (in almost all the experiments conducted, the result was not worse, and in some cases much better, with fewer artifacts). In the future, there are plans to improve the metric for evaluating the quality of image reconstruction, which will allow for a better result.

7. References

[1] D. Krishnan, T. Tay, R. Fergus, Blind deconvolution using a normalized sparsity measure, in: CVPR 2011, 2011, pp. 233-240, doi: 10.1109/CVPR.2011.5995521.

[2] A.O. Trubakov, T.D. Prazdnikova. Restoration of distorted image areas. Proceedings of the 28th International Conference on Computer Graphics and Vision, Moscow, 2018, P.300-303.

[3] Kupyn O., Budzan V., Mykhailych M., Mishkin D., Matas J., DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8183-8192, doi: 10.1109/CVPR.2018.00854.

[4] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, W. T. Freeman, Removing camera shake from a single photograph, in: ACM SIGGRAPH 2006 papers, 2006, pp. 787-794, doi:10.1145/1179352.1141956.

[5] A. Levin, Y. Weiss, f. Durand, and W. T. Freeman. Efficient marginal likelihood optimization in blind deconvolution. CVPR, 2011, doi: 10.1109/CVPR.2011.5995308.

[6] L. Zhong, S. Cho, D. Metaxas, S. Paris and J. Wang, Handling Noise in Single Image Deblurring Using Directional Filters, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 612-619, doi: 10.1109/CVPR.2013.85.

[7] T. S. Cho, S. Paris, B. K. P. Horn and W. T. Freeman, "Blur kernel estimation using the radon transform," CVPR 2011, 2011, pp. 241-248, doi: 10.1109/CVPR.2011.5995479.

[8] A.O. Trubakov, A.A. Trubakova. Method for Improving an Image Segment in a Video Stream Using Median Filtering and Blind Deconvolution Based on Evolutionary Algorithms. CEUR Workshop Proceedings 2744, pp. 1-11, doi: 10.51130/graphicon-2020-2-3-80.

[9] J. Pan, Z. Hu, Z. Su and M. Yang, Deblurring Text Images via L0-Regularized Intensity and Gradient Prior, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2901-2908, doi: 10.1109/CVPR.2014.371.

[10] Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, in: IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April 2004, doi: 10.1109/TIP.2003.819861.

[11] W. Eyal, Hands-On Genetic Algorithms with Python, Packt Publishing Ltd., Birmingham, UK, 2020.