

# The solution of the problem of simplifying the images for the subsequent minimization of the image bit depth

Evgenii Semenishchev

Don State Technical University  
344000, Russia, Rostov-on-Don  
sea.sea@mail.ru

Viacheslav Voronin

Don State Technical University  
344000, Russia, Rostov-on-Don  
Voronin\_sl@mail.ru

Igor Shraifel

Don State Technical University  
344000, Russia, Rostov-on-Don  
shraifel17@mail.ru

Irina Tolstova

Don State Technical University  
344000, Russia, Rostov-on-Don  
irinamolstova@yandex.ru

Dmitrii Chernishchov

Don State Technical University  
344000, Russia, Rostov-on-Don  
dimcher@inbox.ru

## ABSTRACT

In this paper, the approach of changing bit depth of images is considered. This type of operation is required when performing primary processing operations, identifying parameters and stitching images. The process of changing bits depth of images is performed in three stages. At each stage, the error minimization criterion is tested. Result of applying the approach allows obtaining numerical region characteristics including the number of clusters, the number of minimum and maximum cluster sizes. To perform the process of minimizing some of the criteria, it is necessary to divide the image into areas. The paper presents a mathematical description of the approach, as well as flowcharts for performing operations of data processing steps. The article gives recommendations for choosing coefficients to obtain optimal minimizing parameters. The test images give an example of performing bit changes on image areas.

## Keywords

Images, bit, pixel brightness, areas detection, changes brightness.

## 1. INTRODUCTION

Analysis of streaming video, as well as data, requires preliminary preparation. Analyzing preliminary image, and searching objects on them, will be faster if the frame bit size is reduced. To reduce the data (bits) it is necessary to change the bits depth. The bits depth cannot be changed by combining the nearest values, but it is possible to do it by applying the block generation procedure and averaging the values in them. The application of this approach allows performing maximum changes to stationary areas and leave unchanged some areas the users are interested in. The use of bits changing operation is applicable: to the image area, to the block next to a certain part of the histogram or the whole image. The application of this approach is important for combining data obtained in different electromagnetic ranges. Also it is significant for a boundary layer formation on a low-level image. An example is the integration of IR and optical image data. In this task, it is possible to perform bit depth reduction to the level of the IR sensor.

This direction is relevant for solving the problem of combining data obtained by sensors operating in different electromagnetic ranges [1]. An example is a problem of obtaining an image with the boundaries of objects obtained with a two video camera [2]. Obtaining the boundaries of objects is carried out by analyzing the data obtained with an optical system. When obtaining the boundaries of objects obtained by IR and optical sensors, in conditions of poor visibility [3]. When solving the problem of obtaining depth maps for a set of data obtained from a group of cameras [4]. For preliminary simplification of data, the main task is to improve the performance of systems working with a great amount of data [5]. For preliminary analysis in case of solving the problem of stitching images into one composition [6] and many other problems.

## 2. STATEMENT OF THE PROBLEM

Let some (first) image be divided into small fragments of the  $K_1, K_2, \dots, K_n$ , and the  $i$ -th fragment consisting of  $l_i$  pixels. Next, the operation of inducing the fragments brightness by the first image

is to be performed. In this paper they are called–large pixels. For the case of  $l_i = 1$ , the large pixel  $K_i$  is an ordinary pixel. All pixels are known to be equal in the image . The brightness of a large pixel is  $K_i$ ,so it is advisable to calculate the arithmetic mean of  $a_i$  the brightness of usual pixels is  $i = 1, 2, \dots, n$ . The image thus obtained will be called the second image, and the set of larger pixels will be denoted by the symbol  $\mathfrak{R}$ . The problem is to divide the set into clusters means to give all large pixels of each cluster  $T_j$  new luminance value. Averaging of their brightness is performed in accordance to the following two formulas:

$$\beta_j = (\max_{K_i \in T_j} a_i + \min_{K_i \in T_j} a_i) / 2, \quad (1)$$

$$\gamma_j = \frac{\sum_{K_i \in T_j} a_i l_i}{\sum_{K_i \in T_j} l_i} \quad (2)$$

In future, the word combination "splitting into clusters" is reduced to one word "splitting". In case of using the method, all  $n$  of large pixels are replaced by brightness of  $\beta_j$  in the first case or  $\gamma_j$  in the second, two more images are obtained, they are called the third and fourth, respectively. Obviously, the necessary condition for quality of the above pointed division is the possibility of enumerating clusters as inequalities

$$\min_{K_i \in T_j} a_i \geq \max_{K_i \in T_{j+1}} a_i, \quad j = 1, 2, \dots, m-1; \quad (3)$$

here:  $m$  – is the number of clusters,  $T_j$  – is the cluster with the number  $j$ . Further dividing the set  $\mathfrak{R}$  into clusters, decompositions are pointed out corresponding to the requirement (3). Any decomposition needs implication

$$1 \leq i < j \leq n, \quad a_i = a_j, \quad K_j \in T_p, K_j \in T_q \Rightarrow p \leq q \quad (4)$$

In paper we propose an algorithm of partitioning. Its use provides a predetermined degree of proximity of the second and third or (and) second and fourth images. Replacing the first image with the subsequent one allows to reduce the processing time. The problem of choosing between bit depth and image quality is solved.

### 3. NOTATION AND DEFINITIONS

Let  $x = (x_1, x_2, \dots, x_m)$  – be any vector with real coordinates. the vector values are arranged in decreasing order to obtain a series of numbers

$\tilde{x}_1 \geq \tilde{x}_2 \geq \dots \geq \tilde{x}_m$  and, respectively, the vector  $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_m)$  of the same dimension  $m$ .

The calculation procedure of the algorithm presented in the work begins with the construction of the brightness vector of all large pixels  $\tilde{a} = (\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n)$ , situated in descending order. For any pair of large pixels  $K_i, K_j (i < j)$  with the same luminosity, it is assumed that in the coordinate recording the vector  $\tilde{a}$ , the brightness of  $K_i$  precedes the brightness of  $K_j$ . This requirement, together with the monotonicity of the sequence  $\tilde{a}$ , will ensure, (3), (4), the one-to-one correspondence between all partitions of the set  $\mathfrak{R}$  and all possible separations of the vector  $a$  on sub vectors (vectors of the form  $(\tilde{a}_i, \tilde{a}_{i+1}, \dots, \tilde{a}_j)$  when  $1 \leq i \leq j \leq n$ ). To solve the problem, it is possible to use formulas containing the function  $sign(x)$ . The notation is used:  $\tau_j$  – is the number of coordinates of the vector  $a = (a_1, a_2, \dots, a_n)$ , large  $j$ -th coordinates or equal to it;  $\eta_j$  – is the number of the  $j$ -th coordinate of the vector  $a$ , after placing the coordinates  $a$  in descending order. The relations are used

$$\tau_j = \sum_{i=1}^n sign(1 + sign(a_i - a_j)), \quad j = 1, 2, \dots, n.$$

Due to the requirement the numbers  $\eta_1, \eta_2, \dots, \eta_n$  are determined. They can be found according to the formulas

$$\eta_j = 1 + \tau_j - \sum_{i=1}^n (1 - (sign(\tau_i - \tau_j))^2) sign(1 + sign(i - j)), \quad j = 1, 2, \dots, n$$

admit the expression

$$\tilde{a}_j = \sum_{i=1}^n a_i (1 - (sign(\eta_i - j))^2), \quad j = 1, \dots, n.$$

The desired vector is obtained  $\tilde{a} = (\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n)$ . Next, the relocation  $\Theta_1, \Theta_2, \dots, \Theta_n$  of the elements of the set  $\{1, 2, \dots, n\}$  is carried out as they are corresponding to changing relocation of  $n_1, n_2, \dots, n_n$ . This relocation is done by taking  $\Theta_{\eta_i} = i$  for each  $1 \leq i \leq n$ . Any partition of the set  $\mathfrak{R}$  into  $m$  clusters has the form  $\{T_j = \{K_{\Theta_{n_{j-1}+1}}, K_{\Theta_{n_{j-1}+2}}, \dots, K_{\Theta_{n_j}}\} | 1 \leq j \leq m\}$  which allows to specify it  $n_1, n_2, \dots, n_{m-1}$  (when  $n_0 = 0 < n_1 < n_2 < \dots < n_m = n$ ). The cluster size is  $T_j$  the difference is  $\Delta_j$  the between distance of the largest and the smallest brightness of large pixels are

$\Delta_j = \tilde{a}_{n_{j-1}+1} - \tilde{a}_{n_j}, j=1,2,\dots,m$ . The maximum  $\Delta$  distances of the clusters will be called the minimal step of the partition:  $\Delta = \max_{1 \leq j \leq m} \Delta_j$ .

Fixing an arbitrary non negative number  $\varepsilon$ .  $m(\varepsilon)$  is possible if the smallest number of clusters of the set  $\mathfrak{R}$  can be divided under the condition that the smallest dimension of the result partition does not exceed  $\varepsilon$ . The value  $\Delta(\varepsilon)$  is the smallest possible size of partitioning the set  $\mathfrak{R}$  into  $m(\varepsilon)$  clusters (obviously, it is  $\Delta(\varepsilon) \leq \varepsilon$ ). Let us introduce  $\sigma(\varepsilon)$  as the smallest possible sum of cluster sizes among all partitions of  $\mathfrak{R}$  into  $m(\varepsilon)$  clusters with minimum size of  $\Delta(\varepsilon)$ . This is true, since there exists a partition of the set  $\mathfrak{R}$  with the dimension which does not exceed  $\varepsilon$ . Under the condition that the set of all partitions of  $\mathfrak{R}$  is finite.

In this paper, the problem of finding a partition of the set  $\mathfrak{R}$  into  $m(\varepsilon)$  of minimal clusters  $\Delta(\varepsilon)$  with the sum of the ranges of the clusters  $\sigma(\varepsilon)$  is solved. The triple minimizing algorithm can be mathematically described and justified. It consists of three stages, each of which minimizes a certain numerical characteristic - the number of clusters, the minimum and the sum of cluster dimensions. At least any of them has an acceptable result. All three minimization are performed for  $\leq \varepsilon$  and each subsequent minimization preserves the result of the previous minimization.

#### 4. MATHEMATICAL JUSTIFICATION OF THE ALGORITHM

For any non negative number  $\delta$ , we define  $\delta$  as the partition of the set  $\mathfrak{R}$  as follows. Since the sequences are increasing  $\tilde{a}_1 - \tilde{a}_i, i=1,2,\dots,n$ . thus a unique number is  $1 \leq n_1 \leq n$  so  $\tilde{a}_1 - \tilde{a}_{n_1} \leq \delta < \tilde{a}_1 - \tilde{a}_{n_1+1}$  (for convenience will be assumed  $\tilde{a}_{n+1} = -\infty$ ). The next number  $n_j$  is to be found, where  $j \geq 1$ . If  $n_j = n$  the required  $\delta$  partition  $(n_1, n_2, \dots, n_{m-1})$  is already constructed; so

$m := j$ . In the case of  $n_j < n$ , it is possible to use  $n_{j+1}$  the only number corresponding to the condition  $\tilde{a}_{n_{j+1}} - \tilde{a}_{n_{j+i}} \leq \delta < \tilde{a}_{n_{j+1}} - \tilde{a}_{n_{j+i}+1}$ . After a finite number of steps, it is possible to obtain a  $\delta$  breakdown of  $(n_1, n_2, \dots, n_{m-1})$ , and the number of its clusters  $m$ .

The algorithm for partitioning into local areas can be realized in three steps.

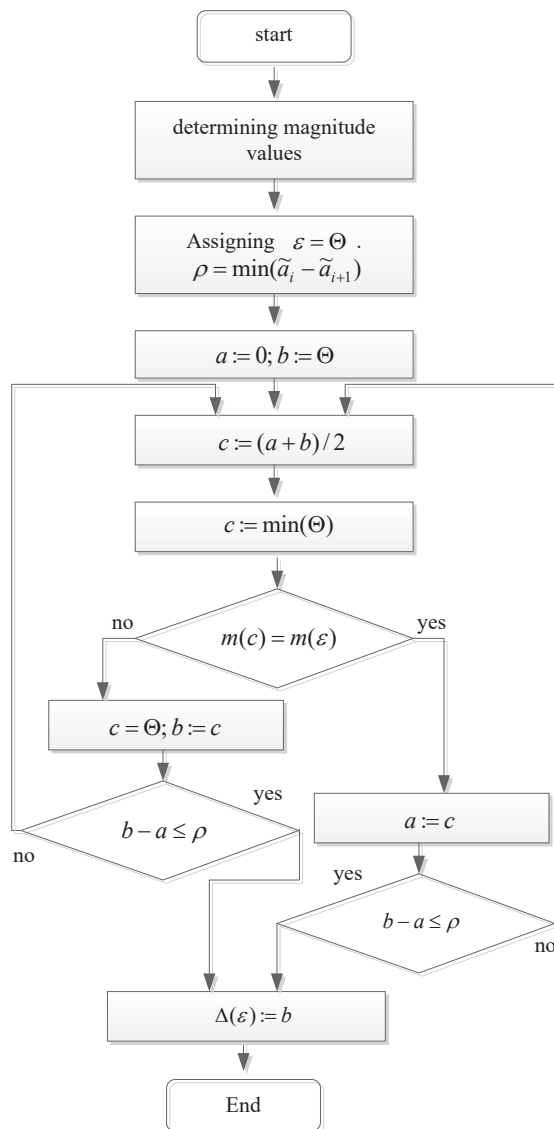


Fig. 1. The algorithm of determining the magnitude of the  $\Delta(\varepsilon)$ .

**I. Calculating the value  $m(\varepsilon)$ .** It is possible to construct an  $\varepsilon$ -partition of the set  $\mathfrak{R}$  and fix the number of its clusters. This value coincides with the value of  $m(\varepsilon)$ , at the first stage of the algorithm.

**II. Calculating the value  $\Delta(\varepsilon)$ .** The algorithm is derived from the construction of the 0-partition, the  $\varepsilon$ -partition of the set  $\mathfrak{R}$ , and the calculation of each of the cluster numbers. The values of  $m(0)$  and  $m(\varepsilon)$  are equal. The minimum of the 0-partition and the size of all its clusters are equal to zero, i.e they assume the values that are minimal for these values. Hence, for  $m(0) = m(\varepsilon)$ , the equalities  $\Delta(\varepsilon) = \sigma(\varepsilon) = 0$ , are true, i.e., 0-partition is a finite result of applying the algorithm. For condition of

$m(0) \neq m(\varepsilon)$ ,  $\varepsilon > 0$  and  $m(0) = m(\varepsilon)$ . Having found the minimum of the  $\varepsilon$ -partition of  $\mathfrak{R}$ , its value is assigned to the variable  $\Theta$ .

It is denoted by  $\rho$  the smallest of the numbers  $(\tilde{a}_i - \tilde{a}_{i+1})$ , where  $1 \leq i \leq n-1$ ,  $\tilde{a}_i > \tilde{a}_{i+1}$  (the inequality  $\Theta > 0$  implies the existence of a number of the specified type). Obviously it is  $\rho > 0$ . The algorithm of determining the magnitude of the  $\Delta(\varepsilon)$  is shown in Figure 1.

The algorithm shown in Figure 1 is implemented by means of the following steps:

- 1) Assigning is  $a := 0; b := \Theta$ ;
- 2) Assigning is  $c := (a + b) / 2$ ;
- 3) Constructing a  $c$ -partition and the assignment of the variable  $\Theta$  to its minimum value;
- 4) When  $m(c) = m(\varepsilon)$ , assigning is  $c = \Theta; b = c$ . If  $m(c) = m(\varepsilon)$  assigning is  $a = c$ .
- 5) If the inequality is true then  $b - a \leq \rho$ .

Assigning is  $\Delta(\varepsilon) := b$ , otherwise it is necessary to return to step 2.

**III. The construction of a partition of the set  $\mathfrak{R}$  into  $m(\varepsilon)$  clusters of minimal dimension  $\Delta(\varepsilon)$ , with the sum of the cluster sizes  $\sigma(\varepsilon)$ .** Let  $m = m(\varepsilon)$  and  $\Delta = \Delta(\varepsilon)$  – the values can be found in the first two stages of the algorithm;  $(n_1, n_2, \dots, n_{m-1})$   $\Delta$ -partition of the set  $\mathfrak{R}$  into  $m$  clusters. The third minimization is realized by the recurrence relation method. It is possible to fix arbitrary numbers  $1 \leq p \leq n$ ,  $1 \leq M \leq m$  and take the notation  $\mathfrak{R}_\rho = \{K_{\Theta_1}, K_{\Theta_2}, \dots, K_{\Theta_\rho}\}$ . The  $\Delta$ -partition clusters of the set  $\mathfrak{R}_\rho$  coincide with the  $\Delta$ -partition clusters of the set  $\mathfrak{R}$ . Only the last cluster of the partition  $\mathfrak{R}_\rho$  can be part of a cluster partition of  $\mathfrak{R}$  with the same number. There is a unique number  $1 \leq j \leq m$  so as  $n_{j-1} + 1 \leq \rho \leq n_j$ .  $\mathfrak{R}, \delta, m$  can be replaced by  $\mathfrak{R}, \Delta, M$  respectively. In this case, it is possible to replace  $m(\Delta)$  by  $j$ . As a result, it is pointed out that it is possible to divide the set  $\mathfrak{R}_\rho$  into  $M$  clusters of size  $\leq \Delta$  if and only if  $j \leq M \leq \rho$ . The equivalence of this condition for double inequality  $M \leq \rho \leq n_M$  is satisfied. The number  $1 \leq q \leq \rho$  is said to be admissible if there exists a decomposition of  $\mathfrak{R}_\rho$  into  $M$  clusters of minimal dimension not exceeding  $\Delta$ , with  $M$ -the cluster  $\{K_{\Theta_q}, \dots, K_{\Theta_\rho}\}$ . For  $q$  to be

admissible, it is necessary and sufficient that: 1) the set  $\mathfrak{R}_{\rho-1}$  can be partitioned into  $M-1$  clusters  $\leq \Delta$ , and 2) the range  $\Delta_M$  of the  $M$  cluster does not exceed  $\Delta$ . Similarly, it is possible to prove the equivalence of condition (1) to the double inequality of  $M-1 \leq q-1 \leq n_{M-1} \Leftrightarrow M \leq q \leq n_{M-1} + 1$ . As  $q$  increases from  $M$  to  $\rho$ , the quantity of  $\Delta_M = \tilde{a}_q - \tilde{a}_\rho$  decreases. In the case if  $q = \rho$ ,  $\Delta_M = 0 < \Delta$ . Consequently, among the values of  $q$  under consideration there is the smallest  $Q$  for which the inequality is  $\Delta_M \leq \Delta$ . The condition (2) will correspond only to numbers of  $Q \leq q \leq \rho$ . In the case of  $\tilde{a}_M - \tilde{a}_\rho \leq \Delta$ , inequality of  $\Delta_M \leq \Delta$  is satisfied by all values  $M \leq q \leq \rho$  and  $Q = M$ .

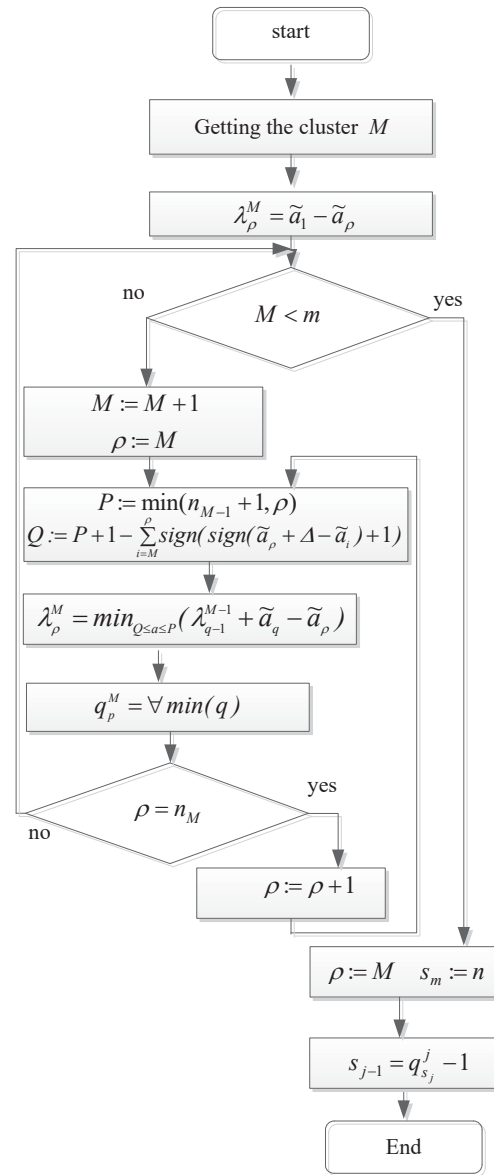


Fig.2. The algorithm of performing step 3.



If  $\tilde{a}_M - \tilde{a}_\rho > \Delta$  number  $Q(> M)$  is the unique solution of such a system of inequalities with unknown  $q$ :  $\tilde{a}_q - \tilde{a}_\rho \leq \Delta$ ,  $\tilde{a}_{q-1} - \tilde{a}_\rho > \Delta \Leftrightarrow \tilde{a}_q - \Delta \leq \tilde{a}_\rho <_{q-1} - \Delta$ . Hence the range of allowed values  $q$ :  $Q \leq q \leq P$ , and  $P := \min(n_{M-1} + 1, \rho)$ . The algorithm for performing step 3 is shown in Figure 2.

The algorithm shown in Figure 2 is implemented in the following steps:

1) Assigning  $M := 1$ . Calculating quantities  $\lambda_\rho^M = \tilde{a}_1 - \tilde{a}_\rho$  at  $1 \leq \rho \leq n_1$ .

2) In case of  $M := m$  go to step 7). At  $M < m$  assigning  $M := M + 1$ .

3) If  $M = m$  assigning  $\rho := n$ . In case of  $M < m$  assigning  $\rho := M$ .

4) Assigning  $P := \min(n_{M-1} + 1, \rho)$ ,  $Q := P + 1 - \sum_{i=M}^{\rho} \text{sign}(\text{sign}(\tilde{a}_\rho + \Delta - \tilde{a}_i) + 1)$

5) Calculating value  $\lambda_\rho^M = \min_{Q \leq a \leq P} (\lambda_{a-1}^{M-1} + \tilde{a}_a - \tilde{a}_\rho)$  and assigning variable  $q_\rho^M$  of one (any) of the values of  $q$  at which the specified minimum is reached.

6) At  $\rho = n_M$  go to step 2). Otherwise – assigning  $\rho := \rho + 1$  and go to step 4).

7) Assigning  $s_m := n$  and then perform a sequential computation of the numbers  $s_{m-1}, s_{m-2}, \dots, s_1$  by the recurrence formula  $s_{j-1} = q_{s_j}^j - 1$ . The resulting partition  $(s_1, s_2, \dots, s_{m-1})$  is optimal, that is, it has the minimum value of  $\Delta$  and the sum of the cluster sizes  $\sigma(\varepsilon)$ .

## 5. RECOMMENDATIONS FOR THE CHOICE OF ALGORITHM PARAMETERS

Recommendations for choosing the value of  $\varepsilon$  for calculation. For each of the numbers  $\rho = 3, \rho = 4$ , denoted by  $Q_\rho(m)$  the maximum modulus of brightness increment of a large pixel are appeared in the transition from the second image to the  $\rho$ -th. We get

$$Q_3(m) = \max_{1 \leq j \leq m} \max_{s_{j-1}+1 \leq i \leq s_j} |\rho_j - a_i| = 0.5\Delta(\varepsilon)$$

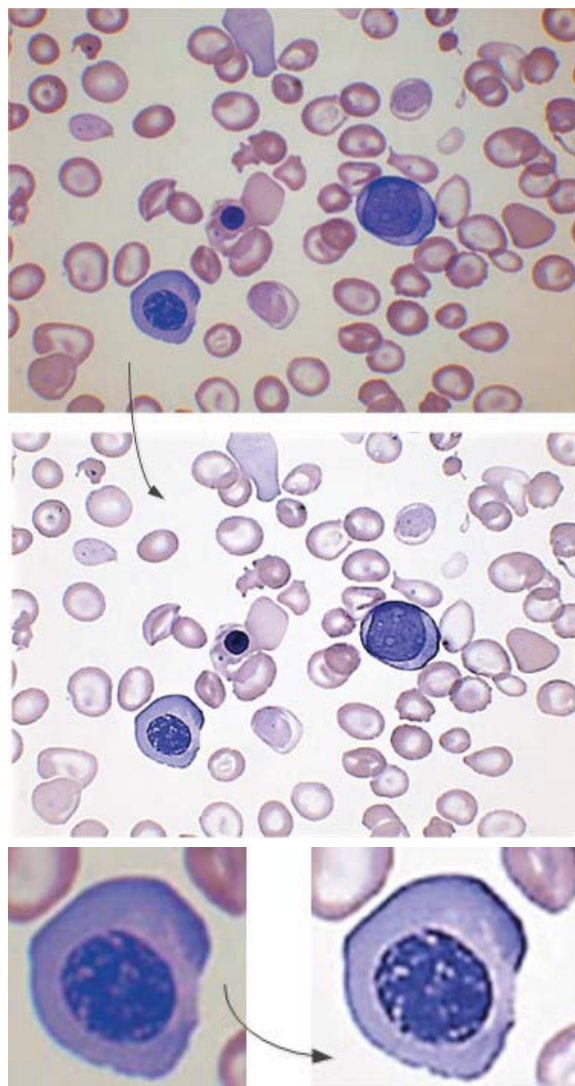
(due to the optimal partition  $(s_1, s_2, \dots, s_{m-1})$ );

$$Q_4(m) = \max_{1 \leq j \leq m} \max_{s_{j-1}+1 \leq i \leq s_j} |\gamma_i - a_i|. \quad \text{It is}$$

necessary to define the third image with the value  $Q_3(m) \leq \delta$ , it is enough to apply the algorithm for  $\varepsilon = 2\delta$ . This will ensure a given degree of proximity of the second and third images. Constructing the fourth image, corresponding to the condition of  $Q_4(m) \leq \delta$  is possible by means of the choice  $\varepsilon = \delta$  to guarantee its observance for limiting the brightness of small pixels to 8 bits (range of values 0, 1, 2, ..., 255). The quantities  $\beta_j, \gamma_j$  will belong to the interval  $[0, 255]$ , however, they can take fractional values. In this case, they should be rounded according to the formula  $x := [x + 0.5]$  ( $[t]$  – is the integer part of the number  $t$ ). As a result, the shift of brightness  $\beta_j, \gamma_j$  to the right or left along the numerical axis can reach 0.5, which requires decreasing of  $\varepsilon = 2\delta$  by one, and  $\varepsilon = \delta$  by 0.5.

## 6. EXAMPLES OF PROCESSING

In Figure 3, an example of changing the bit depth of the areas in the image is presented.





**Fig.3. The result of the approach**

As it can be seen from the results presented in Figure 3, this approach allows fulfilling the bit change operation for local domains on images.

## 7. CONCLUSION

As a conclusion, I want to note the following shortcomings and generalize the advantages. The proposed algorithm allows changing the bit depth in the selected area of the image, however, the selection of this section is performed by the operator and is not automated. When processing the entire image, if the threshold is not chosen correctly, artifacts appear in the form of noise. In the continuation of the work, we see the introduction of automation in the process of searching and defining boundaries. It is also possible to modify the algorithm to use an adaptive threshold setting that allows for partitioning into local areas.

## 8. ACKNOWLEDGMENTS

This work was supported by Russian Ministry of Education and Science in frame of the Federal Program "Research and development on priority directions of scientific-technological complex of Russia Federation in 2014-2020" (contract № 14.577.21.0212 (RFMEFI57716X0212)).

## 9. REFERENCES

- [1] Everitt, J. H., et al. "A three-camera multispectral digital video imaging system." *Remote sensing of environment* 54.3 (1995): 333-337.
- [2] Lloyd, J. Michael. *Thermal imaging systems*. Springer Science & Business Media, 2013.
- [3] Jack, Michael D., Michael Ray, and Richard H. Wyles. "Integrated IR, visible and NIR sensor and methods of fabricating same." U.S. Patent No. 5,808,350. 15 Sep. 1998.
- [4] Kanade, Takeo, et al. "A stereo machine for video-rate dense depth mapping and its new applications." *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*. IEEE, 1996.
- [5] Chen, Min, Shiwen Mao, and Yunhao Liu. "Big data: A survey." *Mobile Networks and Applications* 19.2 (2014): 171-209.
- [6] Semenishchev, Evgeny A., et al. "Stitching algorithm of the images acquired from different points of fixation." *Digital Photography XI*. Vol. 9404. International Society for Optics and