Using the variational principle of leveling illumination in images

Anastasiia Bekesheva[1], Vladyslav Khaidurov[1,2], Vladyslav Romanenko[2], Roman Yarovoy[3†]

[1] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 03056, Prospect Beresteiskyi (former Peremohy) 37, Kyiv, Ukraine
[2] General Energy Institute of NAS of Ukraine, Antonovycha 172, 03150, Kyiv, Ukraine
[3] European University, Academician Vernadsky 16 V, 03115, Kyiv, Ukraine

Abstract

This work describes and practically implements the method of leveling illumination in images of various origins. Insufficient or uneven lighting can result in loss of detail, low contrast, or incorrect color reproduction. Lighting that is too bright can result in areas where detail is lost due to excessive brightness. Also, during the formation and analysis of various medical images, the problem of insufficient illumination of the image arises, that is, the image looks dark, details are difficult to see, especially in the shadows. The practical part of the work contains a description of the mathematical model of the transition from the variational problem (in the optimization form) to the differential form. A comparative analysis of existing computational methods of mathematical physics was also conducted. The working time and the number of iterations for obtaining results that form images with uniform illumination for various images were experimentally determined.

Keywords

Poisson's equation, variational problem formulation, computational methods, elliptic second order equation, equalization of illumination

1. Introduction

The problem of image illumination refers to how light affects the quality and perception of an image. If the lighting is insufficient or uneven, it can result in loss of detail, low contrast, or incorrect color reproduction. Lighting that is too bright can result in areas where detail is lost due to excessive brightness. Also, during the formation and analysis of various medical images, the problem of insufficient illumination of the image arises, that is, the image looks dark, details are difficult to see, especially in the shadows. One of the special problems is the uneven

[1] *ITTAP'2024: 4th International Workshop on Information Technologies: Theoretical and Applied Problems, October 23–25, 2024, Ternopil, Ukraine, Opole, Poland*

* Corresponding author.
† These authors contributed equally.
nastya.bekesheva@gmail.com (A. Bekesheva); allif0111@gmail.com (V. Khaidurov); vlad.romanenko.24@gmail.com (V. Romanenko); roman.yaroviy@e-u.edu.ua (R. Yarovoy)
0009-0007-0479-5161 (A. Bekesheva); 0000-0002-4805-8880 (V. Khaidurov); 0000-0002-3227-4183 (V. Romanenko); 0000-0001-8978-8137 (R. Yarovoy)

illumination of the object under study. Such a problem complicates the correct perception of this object.
Mathematical apparatus of varying complexity is used to analyze unevenly lit images and correct the above-mentioned lighting problems. The methods of leveling lighting in images use a matrix filtering apparatus (based on the principles of convolution), which is known from the course of linear algebra [1], [2]. This device is simple from the point of view of software implementation, and the processor time required to implement the implemented approach is relatively small. The Laplace and Fourier transform apparatus is more complicated than the previous one, but it gives a better visual result of correcting uneven lighting in the image at the output [1], [3]. The processor time for image processing with this mathematical apparatus is significantly greater than the time for image processing with matrix filters. The most well-known mathematical apparatus for solving the problem of processing unevenly lit images is the variational principle, which involves the transition from the optimization formulation of the minimization of a quadratic functional of a special form to finding the solution of the Poisson equation with the appropriate boundary conditions that correspond to the given problem. To date, there are many methods of solving the main classes of problems of mathematical physics, as well as optimization problems of mathematical physics, where functional constraints are equations describing a specific process [3]–[5]. The main computational methods for finding the numerical solution of the Poisson equation are the finite difference method and the finite element method [2], [5], [6]. It should be noted that the finite element method has an advantage over the finite element method for solving the problem of equalizing uneven illumination in the case of examining an image area that has a very complex geometric shape. This paper will describe the image analysis method based on the variational principle and the finite difference method.

2. Problem formulation

2.1. Technical formulation of the problem

The technical task of the work consists in conducting an analysis of unevenly lit images based on the field of the given field of gradients of these images, to obtain a visually improved image. To conduct an analysis of classical computational methods of elliptic equations of mathematical physics of the second order, which are the basis of the variational approach of Poisson image processing.

2.2. Mathematical formulation of the problem: transition from the variational formulation of the problem to the differential formulation

To correct unevenly distributed lighting in the image, you need to preserve all boundary elements of the objects that are in the image under study. In order to distinguish one object from another, it is necessary to have data about the gradient of the entire image, which is presented in the form of two matrices for each channel, if the image is full-color and is presented in digital form in the RGB palette [7]. The Poisson approach involves saving this information on the basis of such a quadratic functional $I_1(u)$, which must be minimized:

$$I_1(u)=\iint\limits_G \left(u_x^{'}-v_x^{'}\right)^2 ds+\iint\limits_G \left(u_y^{'}-v_y^{'}\right)^2 ds \to min,$$

where u – the searched image, which is the result of editing by saving, moreover $u_x^{'}=\partial u/\partial x, u_y^{'}=\partial u/\partial y$, v – input image for which field gradients are known, presented as matrices based on formulas $v_x^{'}=\partial v/\partial x, v_y^{'}=\partial v/\partial y$, G – the computational domain that defines the image itself.

It is obvious that in order to equalize the unevenly distributed illumination in the image, it is necessary to minimize another functional, which has the form:

$$I_2(u)=\iint\limits_G (u-\bar{u})^2 ds \to min,$$

where $\bar{u}$ – average value of illumination. The functional $I_2(u)$ consists in minimizing the variance. A reduction in dispersion implies a much more uniform distribution of the brightness of the pixels of the entire image. In this case, we have the functional of the variational formulation of the problem of leveling uneven illumination in the image given by and:

$$I(u)=\alpha_1 I_1(u)+\alpha_2 I_2(u)=¿$$

$$¿\alpha_1\left(\iint\limits_G \left(u_x^{'}-v_x^{'}\right)^2 ds+\iint\limits_G \left(u_y^{'}-v_y^{'}\right)^2 ds\right)+\alpha_2\iint\limits_G (u-\bar{u})^2 ds \to min,$$

$$\alpha_1+\alpha_2=1.$$

In functionality $I(u)$ parameters α_1 and α_2 є weighting factors that determine the strength of influence of functionals $I_1(u)$ and $I_2(u)$ respectively. It is obvious that the last functional $I(u)$ can also be written with one parameter:

$$I(u)=\alpha_1\left(1\cdot I_1(u)+\frac{\alpha_2}{\alpha_1}\cdot I_2(u)\right)\ I_1(u)+\lambda\cdot I_2(u)=¿$$

$$¿\iint\limits_G \left(u_x^{'}-v_x^{'}\right)^2 ds+\iint\limits_G \left(u_y^{'}-v_y^{'}\right)^2 ds+\lambda\iint\limits_G (u-\bar{u})^2 ds \to min,$$

The last equivalence transformation "~" is true given that the extremal of the initial functional and the resulting functional are the same function. It should also be noted that $\lambda=\alpha_2/\alpha_1, \alpha_1\neq 0$. It is obvious that $\alpha_1\neq 0$, because at $\alpha_1=0$ all information about the initial image (field of image gradients) is lost. This means that at $\alpha_1=0$ the task loses any meaning.

In order to close the mathematical formulation of the variational problem, it is also necessary to mention the boundary conditions. Taking into account the fact that the boundary of the image objects is not tracked at the boundary of the entire image, and also that the minimization of variance for I_2 is better achieved with unfixed boundaries, then

$$\left.\frac{\partial u}{\partial x}\right|_{¿}=\left.\frac{\partial u}{\partial x}\right|_{¿=\left.\frac{\partial u}{\partial y}\right|_{Up}=\left.\frac{\partial u}{\partial y}\right|_{Down}=\left.\frac{\partial u}{\partial \bar{n}}\right|_{\Gamma}=0,¿}$$

where Γ – border of the investigated image.

Such a variational problem can be easily reduced to the differential formulation of the same problem by applying the Euler-Lagrange equation, which has the form:

$$\frac{\partial S(u,u_x^{'},u_y^{'})}{\partial u}-\frac{\partial}{\partial x}\left(\frac{\partial S(u,u_x^{'},u_y^{'})}{\partial u_x^{'}}\right)-\frac{\partial}{\partial y}\left(\frac{\partial S(u,u_x^{'},u_y^{'})}{\partial u_y^{'}}\right)=0,$$

$$S(u,u_x^{'},u_y^{'})=(u_x^{'}-v_x^{'})^2+(u_y^{'}-v_y^{'})^2+\lambda(u-\bar{u})^2.$$

In this case, we will get:

$$S_u^{'}(u,u_x^{'},u_y^{'})=\frac{\partial S(u,u_x^{'},u_y^{'})}{\partial u}=\left((u_x^{'}-v_x^{'})^2+(u_y^{'}-v_y^{'})^2+\lambda(u-\bar{u})^2\right)_u^{'}=2\lambda(u-\bar{u}).$$

$$S_{u_x}^{'}(u,u_x^{'},u_y^{'})=\frac{\partial S(u,u_x^{'},u_y^{'})}{\partial u_x^{'}}=\left((u_x^{'}-v_x^{'})^2+(u_y^{'}-v_y^{'})^2+\lambda(u-\bar{u})^2\right)_{u_x^{'}}^{'}=\text{¿}$$

$$\text{¿}\left((u_x^{'}-v_x^{'})^2\right)_{u_x^{'}}^{'}+0+0=2(u_x^{'}-v_x^{'}).$$

$$S_{u_y}^{'}(u,u_x^{'},u_y^{'})=\frac{\partial S(u,u_x^{'},u_y^{'})}{\partial u_y^{'}}=\left((u_x^{'}-v_x^{'})^2+(u_y^{'}-v_y^{'})^2+\lambda(u-\bar{u})^2\right)_{u_y^{'}}^{'}=\text{¿}$$

$$\text{¿}0+\left((u_y^{'}-v_y^{'})^2\right)_{u_y^{'}}^{'}+0=2(u_y^{'}-v_y^{'}).$$

Let's substitute the obtained intermediate results into the Euler-Lagrange equation. We will have:

$$2\lambda(u-\bar{u})-\frac{\partial}{\partial x}\left(2(u_x^{'}-v_x^{'})\right)-\frac{\partial}{\partial y}\left(2(u_y^{'}-v_y^{'})\right)=\text{¿}$$

$$\text{¿}-2\left(\frac{\partial}{\partial x}(u_x^{'}-v_x^{'})+\frac{\partial}{\partial y}(u_y^{'}-v_y^{'})-\lambda(u-\bar{u})\right).$$

After simplification, we get the equation:

$$u_{xx}^{''}+u_{yy}^{''}-\lambda u=v_{xx}^{''}+v_{yy}^{''}-\lambda\bar{u}.$$

The last equation is Poisson's equation. Taking into account the boundary conditions (derivatives along the normal are equal to zero), it is obvious that if the solution of the equation is some function $u^{\text{¿}}(x,y)$, then the function will also be a solution $u^{\text{¿}}(x,y)+C$, where C – an arbitrary constant. That is, we will prove:

$$(u^{\text{¿}}+C)_{xx}^{''}+(u^{\text{¿}}+C)_{yy}^{''}-\lambda(u^{\text{¿}}+C)=v_{xx}+v_{yy}-\lambda\overline{(u^{\text{¿}}+C)}$$

based on

$$(u^{\text{¿}})_{xx}^{''}+(u^{\text{¿}})_{yy}^{''}-\lambda u^{\text{¿}}=v_{xx}+v_{yy}-\lambda\overline{u^{\text{¿}}}.$$

It is obvious that

$$(u^{\text{¿}}+C)_{xx}^{''}+(u^{\text{¿}}+C)_{yy}^{''}-\lambda(u^{\text{¿}}+C)-\left(v_{xx}+v_{yy}-\lambda\overline{(u^{\text{¿}}+C)}\right)=\text{¿}$$

$$\text{¿}(u^{\text{¿}})_{xx}^{''}+(u^{\text{¿}})_{yy}^{''}-\lambda u^{\text{¿}}-\lambda C-v_{xx}-v_{yy}+\lambda\overline{u^{\text{¿}}}+\lambda C=\text{¿}$$

$$(u^{\text{¿}})_{xx}^{''}+(u^{\text{¿}})_{yy}^{''}-\lambda u^{\text{¿}}+\lambda\overline{u^{\text{¿}}}-v_{xx}-v_{yy}=0$$

That is, this means that to solve the problem, you can put $\bar{u}=0$ to fix the only solution of the problem, which will be easily pulled under the limits of the permissible pixel intensity values (for real numbers - from 0 to 1, or for integers numbers from 0 to 255), for example, according to the classic formula:

$$u=\frac{u-u_{min}}{u_{max}-u_{min}} \text{ або } u=\left[255\cdot\frac{u-u_{min}}{u_{max}-u_{min}}\right].$$

Next, computational experiments will be conducted for the boundary value problem for the Poisson equation, which are known in classical mathematical physics.

3. Computational experiments

3.1. A numerical method for solving the problem

We have a mathematical model that contains a well-known linear equation of mathematical physics of the 2nd order [8], [9]:

$$\lambda u-\Delta u=-\Delta f, \lambda>0 \quad (1)$$

where

$$\Delta u=\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}, \Delta f=\frac{\partial^2 f}{\partial x^2}+\frac{\partial^2 f}{\partial y^2},$$

In (1), we assume that we have a known function $f(x,y)$. This means that Δf is easily found by classical well-known difference schemes. Boundary conditions for equation (1) are equality of zero derivatives.

Suppose we need to write the difference equation for (1) [8]–[10]. Then we will use central difference schemes for any node $(x_i, y_j), i=\overline{1,N_x+2}, j=\overline{1,N_y+2}$. Steps in spatial coordinates x and y let's put equal h_x, h_y. Then:

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_{(x_i,y_j)} \approx \frac{u(x_{i-1},y_j)-2u(x_i,y_j)+u(x_{i+1},y_j)}{(h_x)^2},$$
$$\left.\frac{\partial^2 u}{\partial y^2}\right|_{(x_i,y_j)} \approx \frac{u(x_i,y_{j-1})-2u(x_i,y_j)+u(x_i,y_{j+1})}{(h_y)^2}. \quad (2)$$

Similarly, we write the second derivatives in (1) for the known function $f(x,y)$ in the right-hand side of equation (1).

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{(x_i,y_j)} \approx \frac{f(x_{i-1},y_j)-2f(x_i,y_j)+f(x_{i+1},y_j)}{(h_x)^2},$$
$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{(x_i,y_j)} \approx \frac{f(x_i,y_{j-1})-2f(x_i,y_j)+f(x_i,y_{j+1})}{(h_y)^2}. \quad (3)$$

Let's enter the notation $u_{i,j}=u(x_i,y_j), f_{i,j}=f(x_i,y_j), i=\overline{1,N_x+1}, j=\overline{1,N_y+1}$. Then (2) and (3) will have the form:

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_{(x_i,y_j)} \approx \frac{u_{i-1,j}-2u_{i,j}+u_{i+1,j}}{h_x^2}, \left.\frac{\partial^2 u}{\partial y^2}\right|_{(x_i,y_j)} \approx \frac{u_{i,j-1}-2u_{i,j}+u_{i,j+1}}{h_y^2},$$
$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{(x_i,y_j)} \approx \frac{f_{i-1,j}-2f_{i,j}+f_{i+1,j}}{h_x^2}, \left.\frac{\partial^2 f}{\partial y^2}\right|_{(x_i,y_j)} \approx \frac{f_{i,j-1}-2f_{i,j}+f_{i,j+1}}{h_y^2}, \quad (4)$$
$$i=\overline{2,N_x+1}, j=\overline{2,N_y+1}.$$

Using (4), we finally write down) the difference scheme for (1):

$$\lambda u_{i,j}-\left(\frac{u_{i-1,j}-2u_{i,j}+u_{i+1,j}}{h_x^2}+\frac{u_{i,j-1}-2u_{i,j}+u_{i,j+1}}{h_y^2}\right)=¿$$
$$¿-\left(\frac{f_{i-1,j}-2f_{i,j}+f_{i+1,j}}{h_x^2}+\frac{f_{i,j-1}-2f_{i,j}+f_{i,j+1}}{h_y^2}\right), \quad (5)$$
$$i=\overline{2,N_x+1}, j=\overline{2,N_y+1}.$$

Let's transform equation (5), collecting all the coefficients near the points $(x_i,y_j),(x_{i-1},y_j),(x_{i+1},y_j)$, (x_i,y_{j-1}) and (x_i,y_{j+1}), and we will see the 5-point difference scheme for equation (1) in the new notation.

$$\frac{u_{i-1,j}-2u_{i,j}+u_{i+1,j}}{h_x^2}+\frac{u_{i,j-1}-2u_{i,j}+u_{i,j+1}}{h_y^2}-\lambda u_{i,j}=¿$$
$$¿\frac{f_{i-1,j}-2f_{i,j}+f_{i+1,j}}{h_x^2}+\frac{f_{i,j-1}-2f_{i,j}+f_{i,j+1}}{h_y^2}, i=\overline{2,N_x+1}, j=\overline{2,N_y+1}.$$

$$\frac{1}{h_x^2}u_{i-1,j}+\frac{1}{h_x^2}u_{i+1,j}+\frac{1}{h_y^2}u_{i,j-1}+\frac{1}{h_y^2}u_{i,j+1}-\left(\frac{2}{h_x^2}+\frac{2}{h_y^2}+\lambda\right)u_{i,j}=¿$$
$$¿\frac{f_{i-1,j}-2f_{i,j}+f_{i+1,j}}{h_x^2}+\frac{f_{i,j-1}-2f_{i,j}+f_{i,j+1}}{h_y^2}, i=\overline{2,N_x+1}, j=\overline{2,N_y+1}. \quad (6)$$

We will introduce new notations in the difference equation (6):

- coefficient near $u_{i-1,j}$: $AX_{i,j}=1/h_x^2$;
- coefficient near $u_{i+1,j}$: $CX_{i,j}=1/h_x^2$;
- coefficient near $u_{i,j-1}$: $AY_{i,j}=1/h_y^2$;
- coefficient near $u_{i,j+1}$: $CY_{i,j}=1/h_y^2$;
- coefficient near $u_{i,j}$: $B_{i,j}=2/h_x^2+2/h_y^2+\lambda$;
- coefficient of the right side of the equation $D_{i,j}$:

$$D_{i,j}=\frac{f_{i-1,j}-2f_{i,j}+f_{i+1,j}}{h_x^2}+\frac{f_{i,j-1}-2f_{i,j}+f_{i,j+1}}{h_y^2}.$$

In this case, (6) will turn into the following differential equation:

$$AX_{i,j}u_{i-1,j}+CX_{i,j}u_{i+1,j}+AY_{i,j}u_{i,j-1}+CX_{i,j}u_{i,j+1}-B_{i,j}u_{i,j}=D_{i,j},\quad i=\overline{2,N_x+1},\ j=\overline{2,N_y+1}. \tag{7}$$

Equation (7) contains the unknown function $u(u_{i,j})$ at each point of the calculation area $(x_i, y_j), i=\overline{2,N_x+1}, j=\overline{2,N_y+1}$.

For (7), you can apply any known computational method of linear algebra, or use multi-grid methods, which will significantly speed up the calculations, given that the number of unknowns in the calculation area is large (pixels of the image). Also, it should be noted that taking into account that $\lambda>0$, then the difference equation (7) has a diagonal advantage, which consists in the fact that in each row the module of each diagonal element of the SLAR is not less than the sum of the modules of the other elements of the row [8]–[10].

Now, to close the issue of the model, consider the consideration of the zero derivative at all boundaries of the calculation area [10].

Let our calculation area be in a rectangle $[x_a; x_b]\times[y_a; y_b]$. To begin with, consider the zero derivative on the left boundary:

$$\left.\frac{\partial u}{\partial x}\right|_{(x_a;y)}=0.$$

Based on this, we need to adjust equation (7) on the left boundary:

$$AX_{2,j}u_{1,j}+CX_{2,j}u_{3,j}+AY_{2,j}u_{2,j-1}+CX_{2,j}u_{2,j+1}-¿ \\ -B_{2,j}u_{2,j}=D_{2,j},\ j=\overline{2,N_y+1}. \tag{8}$$

Given that

$$\left.\frac{\partial u}{\partial x}\right|_{(x_a;y)}\approx\frac{u_{2,j}-u_{1,j}}{h_x}=0,\ u_{1,j}=u_{2,j},\ j=\overline{2,N_y+1}. \tag{9}$$

Substitute the obtained boundary condition (9) on the left into (8):

$$AX_{2,j}u_{2,j}+CX_{2,j}u_{3,j}+AY_{2,j}u_{2,j-1}+¿ \\ +CX_{2,j}u_{2,j+1}-B_{2,j}u_{2,j}=D_{2,j},\ j=\overline{2,N_y+1}. \tag{10}$$

Finally, let's transform (10) into the form:

$$CX_{2,j}u_{3,j}+AY_{2,j}u_{2,j-1}+CX_{2,j}u_{2,j+1}-¿ \\ -(B_{2,j}-AX_{2,j})u_{2,j}=D_{2,j},\ j=\overline{2,N_y+1}. \tag{11}$$

In equation (11), it should be noted that the coefficients are adjusted:

$$\overline{B_{2,j}}=B_{2,j}-AX_{2,j},\ AX_{2,j}=0,\ j=\overline{2,N_y+1}.$$

Now we need to correct equation (7) on the right boundary:

$$AX_{N_x+1,j}u_{N_x,j}+CX_{N_x+1,j}u_{N_x+2,j}+AY_{N_x+1,j}u_{N_x+1,j-1}+¿$$
$$+CX_{N_x+1,j}u_{N_x+1,j+1}-B_{N_x+1,j}u_{N_x+1,j}=D_{N_x+1,j},\ j=\overline{2,N_y+1}. \quad (12)$$

Given that

$$\left.\frac{\partial u}{\partial x}\right|_{(x_b;y)}\approx\frac{u_{N_x+2,j}-u_{N_x+1,j}}{h_x}=0,\ u_{N_x+1,j}=u_{N_x+2,j},\ j=\overline{2,N_y+1}. \quad (13)$$

Substitute into (12) the obtained boundary condition (13) on the right:

$$X_{N_x+1,j}u_{N_x,j}+CX_{N_x+1,j}u_{N_x+1,j}+AY_{N_x+1,j}u_{N_x+1,j-1}+¿$$
$$+CX_{N_x+1,j}u_{N_x+1,j+1}-B_{N_x+1,j}u_{N_x+1,j}=D_{N_x+1,j},\ j=\overline{2,N_y+1}. \quad (14)$$

Finally, let's transform (14) into the form:

$$X_{N_x+1,j}u_{N_x,j}+AY_{N_x+1,j}u_{N_x+1,j-1}+CX_{N_x+1,j}u_{N_x+1,j+1}-¿$$
$$-\left(B_{N_x+1,j}-CX_{N_x+1,j}\right)u_{N_x+1,j}=D_{N_x+1,j},\ j=\overline{2,N_y+1}. \quad (15)$$

In equation (15), it should be noted that the coefficients are adjusted:

$$\overline{B_{N_x+1,j}}=B_{N_x+1,j}-CX_{N_x+1,j},\ CX_{N_x+1,j}=0,\ j=\overline{2,N_y+1}.$$

We adjust equation (7) at the lower limit:

$$AX_{i,2}u_{i-1,2}+CX_{i,2}u_{i+1,2}+AY_{i,2}u_{i,1}+CX_{i,2}u_{i,3}-¿$$
$$-B_{i,2}u_{i,2}=D_{i,2},\ i=\overline{2,N_x+1}. \quad (16)$$

Given that

$$\left.\frac{\partial u}{\partial y}\right|_{(x;y_a)}\approx\frac{u_{i,2}-u_{i,1}}{h_y}=0,\ u_{i,1}=u_{i,2},\ i=\overline{2,N_x+1}. \quad (17)$$

Substitute into (16) the obtained boundary condition (17) on the right:

$$AX_{i,2}u_{i-1,2}+CX_{i,2}u_{i+1,2}+AY_{i,2}u_{i,2}+CX_{i,2}u_{i,3}-¿$$
$$-B_{i,2}u_{i,2}=D_{i,2},\ i=\overline{2,N_x+1}. \quad (18)$$

Finally, let's transform (18) into the form:

$$AX_{i,2}u_{i-1,2}+CX_{i,2}u_{i+1,2}+CX_{i,2}u_{i,3}-¿$$
$$-\left(B_{i,2}-AY_{i,2}\right)u_{i,2}=D_{i,2},\ i=\overline{2,N_x+1}. \quad (19)$$

In equation (19), it should be noted that the coefficients are adjusted:

$$\overline{B_{i,2}}=B_{i,2}-AY_{i,2},\ AY_{i,2}=0,\ i=\overline{2,N_x+1}.$$

We adjust equation (7) at the lower limit:

$$AX_{i,N_y+1}u_{i-1,N_y+1}+CX_{i,N_y+1}u_{i+1,N_y+1}+AY_{i,N_y+1}u_{i,N_y}+CX_{i,N_y+1}u_{i,N_y+2}-\text{¿} \\ -B_{i,N_y+1}u_{i,N_y+1}=D_{i,N_y+1},i=\overline{2,N_x+1}. \quad (20)$$

Given that

$$\left.\frac{\partial u}{\partial y}\right|_{(x;y_b)}\approx\frac{u_{i,N_y+2}-u_{i,N_y+1}}{h_y}=0,u_{i,N_y+1}=u_{i,N_y+2},i=\overline{2,N_x+1}. \quad (21)$$

Substitute into (20) the obtained boundary condition (21) on the right:

$$AX_{i,N_y+1}u_{i-1,N_y+1}+CX_{i,N_y+1}u_{i+1,N_y+1}+AY_{i,N_y+1}u_{i,N_y}+\text{¿} \\ +CY_{i,N_y+1}u_{i,N_y+1}-B_{i,N_y+1}u_{i,N_y+1}=D_{i,N_y+1},i=\overline{2,N_x+1}. \quad (22)$$

Finally, let's transform (22) into the form:

$$AX_{i,N_y+1}u_{i-1,N_y+1}+CX_{i,N_y+1}u_{i+1,N_y+1}+AY_{i,N_y+1}u_{i,N_y}-\text{¿} \\ -\left(B_{i,N_y+1}-CY_{i,N_y+1}\right)u_{i,N_y+1}=D_{i,N_y+1},i=\overline{2,N_x+1}. \quad (23)$$

In equation (23), it should be noted that the coefficients are adjusted:

$$\overline{B_{i,N_y+1}}=B_{i,N_y+1}-CY_{i,N_y+1},CY_{i,N_y+1}=0,i=\overline{2,N_x+1}.$$

For a rectangular area, the values at the corners of this area also raise questions. Here, too, everything is simple. As an option, you can make "stubs" as the arithmetic mean of two known nearest neighboring limit values:

$$u_{1,1}=\frac{u_{1,2}+u_{2,1}}{2},u_{N_x+2,N_y+2}=\frac{u_{N_x+1,N_y+2}+u_{N_x+2,N_y+1}}{2}, \\ u_{1,N_y+2}=\frac{u_{1,N_y+1}+u_{2,N_y+2}}{2},u_{N_x+2,1}=\frac{u_{N_x+1,1}+u_{N_x+2,2}}{2}. \quad (24)$$

It should be noted that the boundary nodes of the region (24) do not take part in the calculation procedures, since the 5-point difference scheme does not pass through these nodes.

3.2. Testing the numerical method on various images

The resulting algorithm was tested on 30 different images. The timing results for the first six images are shown in Tables 1–4. It should be noted that the block sweep algorithm performed better than all others for solving the discrete analog for the set task of equalizing the uneven distribution of pixel intensity in the image. The algorithm of the Scipy library of the Python programming language also performed well [11]–[13]. It should also be noted that computational methods of linear algebra more effectively solve systems of linear algebraic equations when the matrices of this system have a diagonal advantage [14], [15]. The greater the difference between the modulus of the diagonal element and the sum of the modules of the other elements of the row (and for each row), the faster the iterative method or algorithm converges.

The calculation error of the iterative methods, the results of which are given in the above tables, is equal to $\varepsilon = 10^{-7}$.

Table 1

Time characteristics of finding a solution to the problem using the Seidel method for six different images

Image	Vertical Pixels	Horizontal Pixels	Total Pixels	Time for λ=0.0001, sec	Time for λ=0.0005, sec	Time for λ=0.001, sec	Time for λ=0.01, sec
1	216	197	42552	0,210937312	0,089896336	0,056801848	0,01207815
2	450	297	133650	0,389937561	0,136101551	0,081752021	0,02127122
3	216	218	47088	0,225398738	0,123498702	0,067352707	0,01153639
4	249	186	46314	0,076201501	0,042581451	0,031306698	0,00780742
5	192	256	49152	0,255173569	0,191466012	0,100008831	0,01471437
6	341	404	137764	0,589075354	0,301040809	0,196141768	0,03152153

Table 2

Time characteristics of finding a solution to a problem (for six different images) using built-in Python tools, namely the Scipy library for solving a system of linear equations

Image	Vertical Pixels	Horizontal Pixels	Total Pixels	Time for λ=0.05, sec	Time for λ=0.1, sec	Time for λ=0.15, sec	Time for λ=0.2, sec
1	216	197	42552	0,003438	0,002027	0,00184	0,00128
2	450	297	133650	0,006911	0,004971	0,00313	0,00261
3	216	218	47088	0,002791	0,001621	0,00123	0,00091
4	249	186	46314	0,003047	0,001878	0,00159	0,00095
5	192	256	49152	0,004353	0,002112	0,00171	0,00131
6	341	404	137764	0,008112	0,005327	0,00345	0,00262

Next, we test the same algorithm based on the already known approximate algorithm of longitudinal-transverse running (the results are shown in Table 3).

Table 3

Time characteristics of finding a solution to the problem using longitudinal-transverse running method for six different images

Image	Vertical Pixels	Horizontal Pixels	Total Pixels	Time for λ=0.0001, sec	Time for λ=0.0005, sec	Time for λ=0.001, sec	Time for λ=0.01, sec
1	216	197	42552	0,13429230	0,09056245	0,08532415	0,09177863
2	450	297	133650	0,01377897	0,01531005	0,01567527	0,01533334
3	216	218	47088	0,01409682	0,01637427	0,01681701	0,01563258
4	249	186	46314	0,09475455	0,07616487	0,06050883	0,06152226
5	192	256	49152	0,06123956	0,06735462	0,04519732	0,04193212
6	341	404	137764	0,13429230	0,09056275	0,08513215	0,09177863

We will do the same for the analytical block sweep algorithm, which makes it possible to obtain an exact solution to the problem, taking into account the fact that the discrete analogue of the Poisson equation has a clear sparse and block structure. (results are shown in Table 4).

Table 4
Time characteristics of finding a solution to the problem using the matrix run method for six different images

Image	Vertical Pixels	Horizontal Pixels	Total Pixels	Time for λ=0.05, sec	Time for λ=0.1, sec	Time for λ=0.15, sec	Time for λ=0.2, sec
1	216	197	42552	0,078135991	0,07168536	0,068275028	0,069264852
2	450	297	133650	0,013946464	0,016654	0,013836376	0,018284613
3	216	218	47088	0,016022084	0,01541467	0,013604557	0,017517877
4	249	186	46314	0,065097637	0,060555248	0,050360774	0,051381867
5	192	256	49152	0,039356135	0,03993877	0,044371414	0,042931808
6	341	404	137764	0,078135991	0,07168536	0,068275028	0,069264852

The visual results of the program are shown below.

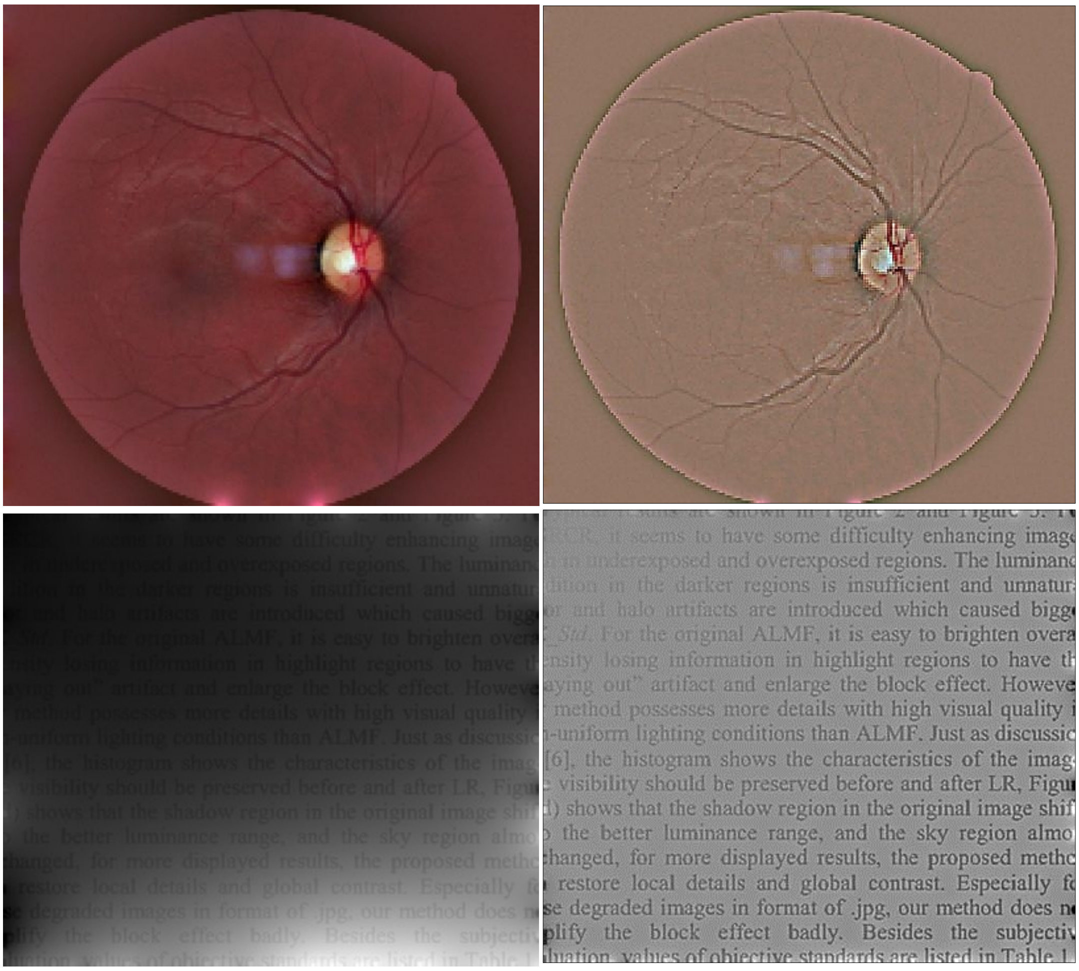

Figure 1: An example of the described algorithm for two different images:
on the left – the original image, on the right – the result of algorithm processing

As can be seen from Figure 1, all images have a better pixel intensity distribution as a result of applying the algorithm.

4. Conclusions

The practical result of this work is a comparative analysis of unevenly lit images based on the field of the given field of gradients of these images, to obtain a visually improved image.

In the work, mathematical dependencies are obtained for constructing an approximate solution to the set problem of equalizing the uneven distribution of pixel intensity. An analysis of classical computational methods of second-order elliptic equations of mathematical physics, which are the basis of the variational approach of Poisson image processing, was carried out. Seidel's method performed the worst in terms of time, the block run method using the built-in Python library Scipy gave the best result in terms of time.

In order to obtain better time indicators for a specific accuracy, it is best to use multi-grid approximate methods, which make it possible to significantly reduce the number of iterations, which means to reduce the time to execute the program.

It should also be noted that the speed of iterative methods and algorithms will be higher if calculations are performed in the normal system (from 0 to 1) and not in the classic system (from 0 to 255).

References

[1] Ganesh R. Naik, Wellington P. (2024) Biomedical Signal Processing. A Modern Approach. ISBN 10: 103206191X, ISBN 13: 9781032061917

[2] Xue-Cheng T., Suhua W., Haiguang L. (2018) Mathematical Methods in Image Processing and Inverse Problems: IPIP 2018, Beijing, China, April 21–24 (Springer Proceedings in Mathematics & Statistics, 360) 1st ed. 2021. DOI: https://doi.org/10.1007/978-981-16-2701-9.

[3] Zhou H., Qin R., Liu Z., Qian Y., Ju X. (2022) Optimizing Performance of Image Processing Algorithms on GPUs. Proceeding of 2021 International Conference on Wireless Communications, Networking and Applications. WCNA 2021. Lecture Notes in Electrical Engineering. Springer, Singapore. DOI: https://doi.org/10.1007/978-981-19-2456-9_95.

[4] Mitsunori G. (2019) Image Processing Using ImageJ. Japanese Journal of Radiological Technology 75, no. 7. DOI: http://dx.doi.org/10.6009/jjrt.2019_jsrt_75.7.688.

[5] Sivakumar S., Saminathan S., Ranjana R., Mohan M., Pareek P. (2023) Malware Detection Using the Machine Learning Based Modified Partial Swarm Optimization Approach, International Conference on Applied Intelligence and Sustainable Computing (ICAISC), Dharwad, India, 2023, pp. 1–5. DOD: 10.1109/ICAISC58445.2023.10199796.

[6] Lande J., Maheswari M., Kamali S. M., Dineshkumar R. (2023) Hybrid Optimization Based Long Short-Term Memory for Android Malware Detection. International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT), Bengaluru, India, 2023, pp. 1–5. DOD: 10.1109/EASCT59475.2023.10393408.

[7] Kotian P., Sonkusare R. (2021) Detection of Malware in Cloud Environment using Deep Neural Network, 6th International Conference for Convergence in Technology, Maharashtra, India, 2021, pp. 1-5. DOI: 10.1109/I2CT51068.2021.9417901.

[8] Khaidurov V., Zaporozhets A., Tsiupii T. (2022) Creation of High-Speed Methods for Solving Mathematical Models of Inverse Problems of Heat Power Engineering. Springer, Systems Decision and Control in Energy III, vol. 399, 41–74. DOI: https://doi.org/10.1007/978-3-030-87675-3. ISSN: 2198-4182.

[9] Khaidurov V., Yailymov B., Shelestov A. (2023) Mathematical Model for Determining the Geometric Location of the Environmental Pollutant Based on Sensor Data. Proceedings of the IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS, pp. 703–707.

[10] Khaidurov V., Tsiupii T., Zhovnovach T. (2021) Modelling of ultrasonic testing and diagnostics of materials by application of inverse problems. CEUR Workshop Proceedings, 2021, 3039, Pp. 1–5.

[11] Azimzadeh P., Forsyth P.A. (2016) Weakly Chained Matrices, Policy Iteration, and Impulse Control. SIAM J. No. 54 (3), 1341–1364.
[12] Chen Y., Wan J., Lin J. (2018) Monotone mixed finite difference scheme for Monge–Ampère equation. J. Sci. Comput. 76, 1839–1867.
[13] Chen Y., Wan J. (2017) Multigrid methods for convergent mixed finite difference scheme for Monge-Ampère equation. Comput. Visual. Sci., 1–15.
[14] Chen Y., Wan J. (2018) Numerical method for image registration model based on optimal mass transport. Inverse Problems Imaging 12(2), 401–432.
[15] Chow S.N., Li W., Zhou H. (2019) A discrete Schrödinger equation via optimal transport on graphs. Journal of Functional Analysis 276, 2440–2469.