Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 13, Number 9 (2017), pp. 5447-5459

http://www.ripublication.com

HSAOR Iteration for Poisson Image Blending Problem via Rotated Five-Point Laplacian Operator

[1]Eng Jeng Hong*, [2]Azali Saudi and [3]Jumat Sulaiman

[1,3]Mathematics with Economics Programme, Faculty of Science and Natural Resources, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia.

[2]Computer Science Programme, Faculty of Computing and Informatics, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia.

Abstract

Finite difference approach via standard and rotated five-point Laplacian operator is used to discretize the Poisson equation to solve the blending problem. Poisson image blending has becoming an important tool in image processing to generate a desirable image which is impossible to acquire. The solution to the Poisson image blending minimization problem is equivalent to the unique solution of the two-dimensional Poisson partial differential equation. Thus, the motivation of this paper is to solve the linear system generated from the Poisson equation by using three selected efficient iterative methods, namely Full-Sweep Successive Over Relaxation (FSSOR), Full-Sweep Accelerated Over Relaxation (FSAOR) and Half-Sweep Accelerated Over Relaxation (HSAOR) iterative methods. The Poisson equation is discretized via standard and rotated five-point Laplacian operator respectively. The linear system formed is then solved by FSSOR, FSAOR and HSAOR iterative methods respectively. These proposed iterative methods are examined via the number or iterations involved and the computational time taken to determine their efficiency. Due to the reason that the computational complexities had reduced in half-sweep algorithm as compared to full-sweep algorithm, HSAOR has the best performance. The

* Corresponding Author

quality of the generated images by the three proposed iterative methods has no significant difference which looks natural and realistic.

Keywords: Poisson equation; Poisson image blending; half-sweep accelerated over relaxation iteration; rotated five-point Laplacian operator

INTRODUCTION

An English idiom sounds, "One picture is worth more than ten thousand words" describes the importance and effectiveness of a picture to deliver the message to the readers. Advanced technologies that have shaped the world we live in has increased the dependency on digital images in our daily life. Digital images are formed when pictures are represented in digital form and when we processed the digital images by a digital computer, it is known as digital image processing. Digital image processing is every part of our life as without noticing it, we are consuming it in our daily life. The historical background of digital image processing can refer to [1]. It has a broad range of applications since it is invented. For instance, medical imaging, geographical mapping, observation of the earth resources and remote sensing where medical imaging is growing rapidly with the new invention on classification of brain Magnetic Resonance (MR) images by referring on the age of the patients and the disease status of Alzheimer and seizure detection [2].

Image processing comprises of few core classes that we normally applied, image analysis, image enhancement, image restoration, image synthesis and image compression. Various studies on these classes had done to obtain the new desired output images which look natural. Nevertheless, we are examining the problem of image composition in this study. Generally, there are two classifications of image composition, namely image cloning and image blending. According to [3], image cloning is a process of positioning an opaque image elements one over another while image blending semitransparent image elements together. The focus of this paper is image blending based on Poisson equation.

Poisson Image Editing (PIE) which is also known as Poisson image blending, is proposed by [4]. It is a gradient domain compositing method where this method is manipulating the gradients instead of the pixels of an image. Poisson image blending is a process of selecting the desired region from the source image and then extract it to the target image. By solving the linear system generated from the Poisson equation with Dirichlet boundary condition, a new blended image is formed. The motivation of the implementation of Poisson equation in image composition can refer to [4] and the details of the Poisson image blending process is presented in the next Section.

The research done by [5] was trying to solve the issue of color inconsistency which was caused by the Dirichlet boundary condition in [4]. A small modification was done on the proposed method in [5] by adding an additional inner Dirichlet boundary condition and enlarging the Laplacian values. With the two boundaries, this improved method solved the issue effectively by generating a more natural and realistic images. Follow by the efficient non-iterative method proposed by Morel *et al*., [6], the Fourier solver. Fourier solver is a fast and non-iterative method where it can solve the Poisson image blending problem in shortest time and meanwhile generate a good quality of images. Neumann boundary condition is used when solving the Poisson equation instead of Dirichlet boundary condition. Moreover, the desired region required no manually selection but by only an algorithm which had eased the blending process and the images is edited without additional computational cost.

More recently, Hussain and Kamel [7] came out a new idea on how to solve the Poisson image blending problem more efficiently. They proposed a method which is based on image pyramid and divide-and-conquer approach. This is a three pyramid level of process. First, the problem was solved at the third pyramid level and the intensity values obtained from here was used in the second pyramid level. In this level, the unknown region was partitioned into two different types of shapes, which was thin slices and small square blocks respectively. Then, it is solved for these two different patterns separately to test the efficiency and this process was repeated in the first pyramid level to form the final output images. Experimental results showed that the computational time had improved and the quality of the generated images has no significant difference compared to other methods.

In addition, to overcome the issues of bleeding artifacts and color bleeding, Afifi and Hussain [8] suggested a modified Poisson blending (MPB) approach which was specially designed to tackle the problem. Instead of just consider the boundary pixels at destination image which caused the bleeding artifacts problem, MPB was considering both boundary pixels at source and destination images. On the other hand, two generated images from MPB process were composited to solve the color bleeding problem. The results obtained from MPB approach is satisfied and even better than other approaches in video inpainting.

Apart from Poisson image blending, image processing also applied in many other classes of problem. For instance, image inpainting. Image inpainting also known as image completion is a process to restore a damage image or remove a selected object in an image. Various methods had invented for this problem, cloning algorithms [9], Alternating Direction Method (ADM) [10] and Laplace equation approach [11]. Besides, image enhancement is another important tool in image processing to improve the quality of an image, for example the recent researches done in [12,13].

On the other hand, image stitching is a process of merging multiple images to generate a new seamless image. The researchers was using gradient domain approach for this problem in order to yield a better panoramic image [14,15]. Furthermore, image composition which was based on optimizing boundary condition and gradient approaches respectively was invented by [16,17]. There are also numerous researches on image processing which were specially focus on editing large scale of image. For example, parallel gradient domain approach [18] and block Poisson method [19].

There are countless of researches had done in the field image processing. However, we are focusing on how to solve Poisson image blending problem efficiently in this paper. Generally, the system of linear equations generated from Poisson approximation equation can be solved by either direct or iterative methods. In this paper, we are interested to examine the efficiency of iterative method on this problem. Poisson equation is first discretized by second order central difference scheme via standard and rotated five-point Laplacian operator. Then, the linear system generated is solve by using full-sweep and half-sweep algorithm. The ultimate goal of this paper is to examine the efficiency of half-sweep algorithm on Poisson image blending problem as none of the researches are employing this technique in this problem.

Besides, the recent research by Martino *et al*., [20] stated that finite difference approach is well performed in most of the Poisson image blending problem. Thus, we selected three proposed iterative methods for this evaluation, namely Full-Sweep Successive Over Relaxation (FSSOR), Full-Sweep Accelerated Over Relaxation (FSAOR) and Half-Sweep Accelerated Over Relaxation (HSAOR) iterative methods. Two evaluation criteria is used in this paper to examine the efficiency of the proposed iterative methods, the number of iterations involved and the computational time taken. Besides Poisson image blending problem, the Laplace's equation which is generated from Poisson equation is applied in path planning problem [21,22].

The remaining of this paper is organized as: the presentation of the idea of Poisson image editing; discretization of Poisson equation via finite difference approach; formation of proposed iterative methods; numerical experiments; experimental results with new generated images and discussion; and lastly the conclusion.

METHODOLOGY

An image is built up by a finite number of elements which is stored in a two-dimensional array. Each elements has its own location and intensity value or brightness in its domain. These elements are known as pixels. The coordinate system of an image is the rotated conventional two-dimensional Cartesian coordinate system by 90^{o} to the clockwise direction. The finite grid network of an image is clearly presented in Figure 1. The x-axis is increasing from left to right while the y-axis is increasing from top to bottom which is the opposite direction from the conventional coordinate system. The size or the resolution of an image is determined by the number of rows and columns in the image and the higher the resolution, the clearer the image is. Besides, the colorful images that we used to see is actually formed by various mixtures of three primary colors: Red, Green and Blue (RGB).

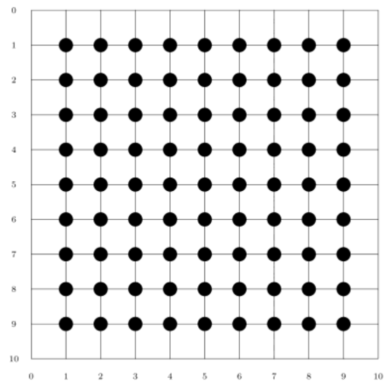


Figure 1: Finite grid network of an image for $n = 10$

Refer to [4], the basic idea of Poisson image blending is starting by selecting the region of interest O from the source image g. Defined ∂O as the boundary of the selected region. Then, the region of interest is blended into the selected destination image f^* to regenerate a new composited image f. In order to produce a seamless blended image, a guidance vector field **v** must be first generated from the source image. Follow by computing a new set of intensity values f in the region of interest which will minimize the different between the gradient of the new image and the vector field. The minimization problem is written as,

$$min_f \iint_O |\nabla f - \mathbf{v}|^2 \ with\, f|_{\partial O} = f^*|_{\partial O} \tag{1}$$

The intensity values are set to be the same at the boundary to build an image which will look more natural and realistic. As we know that, the solution to the minimization problem (1) is equivalent to the unique solution of the Poisson equation with Dirichlet boundary condition,

$$\Delta f = \operatorname{div} \mathbf{v} \ at\ O\ with\, f|_{\partial O} = f^*|_{\partial O} \tag{2}$$

According to [4], the easiest way to obtain the vector field **v** is directly extract the gradient field from the source image and then, Equation (2) is redefined as,

$$\Delta f = \Delta g \ at\ O\ with\, f|_{\partial O} = f^*|_{\partial O} \tag{3}$$

where Δ denotes the Laplacian operator.

Discretization of Poisson Equation via Rotated Five-Point Laplacian Operator

In this paper, the main objective is to investigate the potential of rotated five-point Laplacian operator when it is used for discretized Poisson equation and then solved the Poisson image blending problem. Standard five-point Laplacian operator is used for comparison in this study. Two different finite grid networks which are based on full- and half-sweep algorithms are formed in Figures 1 and 2 respectively.

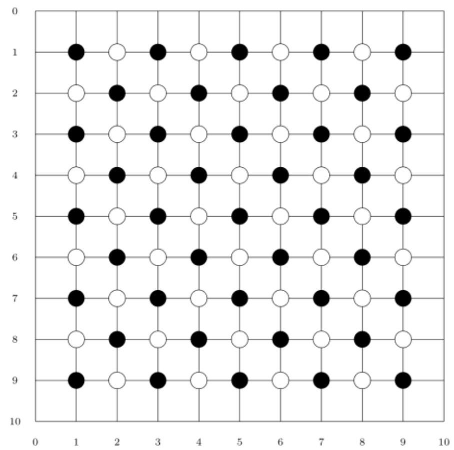


Figure 2: Finite grid network for half-sweep algorithm for $n = 10$

The advantage of half-sweep algorithm as compare to full-sweep algorithm is, its implementation only involved half of the whole inner pixels (•) and then the remaining pixels (∘) is computed directly, refer to Figure 2. On the other hand, the full-sweep algorithm need to calculate all the pixels in the domain which eventually required more number of iterations and computational time.

To solve the minimization problem (1), the two-dimensional Poisson equation which is defined as,

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = f(x, y) \tag{4}$$

is first discretized based on finite difference approach via both standard and rotated five-point Laplacian operators respectively, which are written as

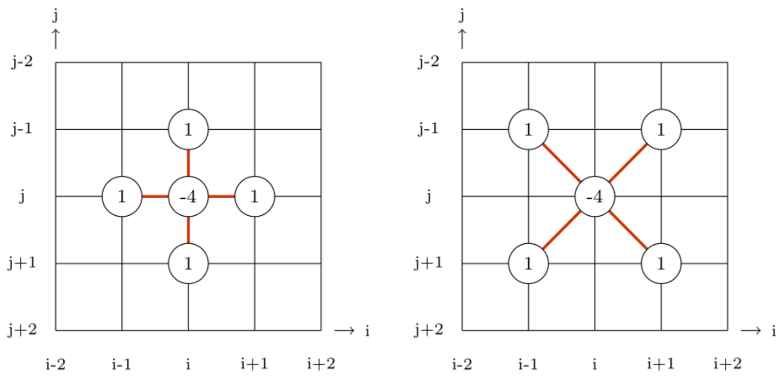
$$U_{i,j} \cong \frac{1}{4}\left[U_{i+1,j} + U_{i-1,j} + U_{i,j+1} + U_{i,j-1} - h^2 f_{i,j}\right] \tag{5}$$

with $h = \Delta x = \Delta y$ where Equation (5) is a full-sweep algorithm and

$$U_{i,j} \cong \frac{1}{4}\left[U_{i+1,j-1} + U_{i-1,j-1} + U_{i-1,j+1} + U_{i+1,j+1} - 2h^2 f_{i,j}\right] \tag{6}$$

with $h = \Delta x = \Delta y$ is a half-sweep algorithm. The computational molecule for

Equations (5) and (6) is showed in Figure 3.



(a) Standard (b) Rotated

Figure 3: Five-Point Laplacian Operators

Then, the linear system is formed based on the approximation equations (5) and (6) respectively,

$$A\underset{\sim}{\boldsymbol{u}} = \underset{\sim}{\boldsymbol{b}} \tag{7}$$

The linear system (7) is solved for n times for three different colors (RGB) channel separately then merged to generate the new output images. n is the number of unknown pixels in the region of interest.

Formation of Successive Over Relaxation and Accelerated Over Relaxation Iterative Methods

Two types of iterative methods which based on full- and half-sweep algorithms are proposed here to solve the large and sparse linear system (7): FSSOR [23,24], FSAOR [25] and HSAOR. Basically, HSAOR iterative method is the extension of FSAOR iterative method with the intention to reduce the computational complexities during the iteration process. First, add a weighted parameter ω into the linear system and rewrite as [25]

$$\omega A\underset{\sim}{\boldsymbol{u}} = \omega\underset{\sim}{\boldsymbol{b}} \tag{8}$$

where $\omega \neq 0$. Then, expressed matrix A to the following form,

$$A = D - L - V \tag{9}$$

where D, L and V are the diagonal matrix, strictly lower triangular matrix and strictly upper triangular matrix respectively.

Next, Equation (9) is substitute into the coefficient matrix ωA and another weighted parameter δ is added as well. Thus, the coefficient matrix is now written as

$$\omega A = (D - \delta L) - [(1 - \omega)D + (\omega - \delta)L + \omega V] \tag{10}$$

with $1 < \omega < 2$ and $1 < \delta < 2$.

Finally, by replacing the new form of coefficient matrix (10) into the linear system and manipulating it, the new form of linear system (7) is rewritten as

$$\underset{\sim}{\boldsymbol{u}} = (D - \delta L)^{-1}[(1 - \omega)D + (\omega - \delta)L + \omega V]\underset{\sim}{\boldsymbol{u}} + \omega(D - \delta L)^{-1}\underset{\sim}{\boldsymbol{b}} \tag{11}$$

and therefore, the general scheme for FSAOR and HSAOR iterative methods is defined as

$$\underset{\sim}{\boldsymbol{u}}^{(\boldsymbol{k}+\mathbf{1})} = (D - \delta L)^{-1}[(1 - \omega)D + (\omega - \delta)L + \omega V]\underset{\sim}{\boldsymbol{u}}^{(\boldsymbol{k})} + \omega(D - \delta L)^{-1}\underset{\sim}{\boldsymbol{b}} \tag{12}$$

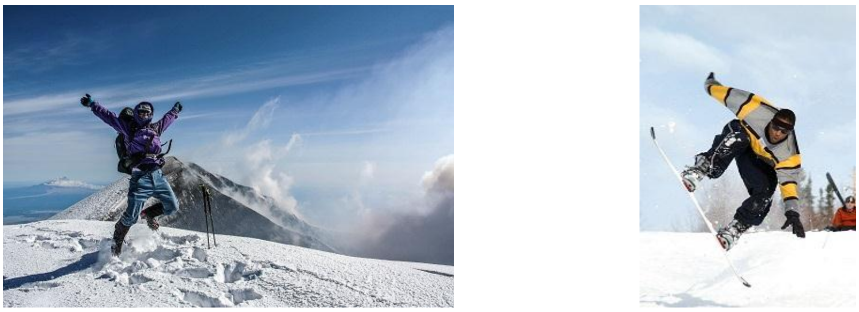
with $k = 1, 2, 3, \dots, n$. Noticed that, when $\omega = \delta$, it reduced to FSSOR iterative method.

RESULTS AND DISCUSSION

In this section, three sets of test examples with source and destination images have chosen to verify the superiority of half-sweep algorithm in solving Poisson image blending problem. These test examples were taken from [26] with different resolutions and are shown in Figure 4. The performance of the three proposed iterative methods is evaluated by the number of iterations involved and the composing time taken to blend the images.

(a)

(b)

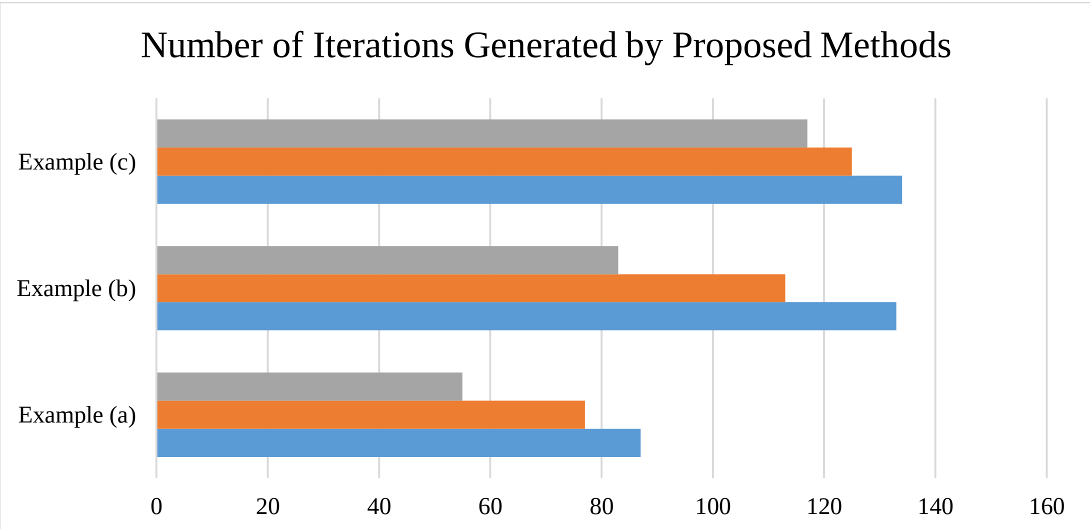


(c)

(i) (ii)

Figure 4: (i) Destination and (ii) source images

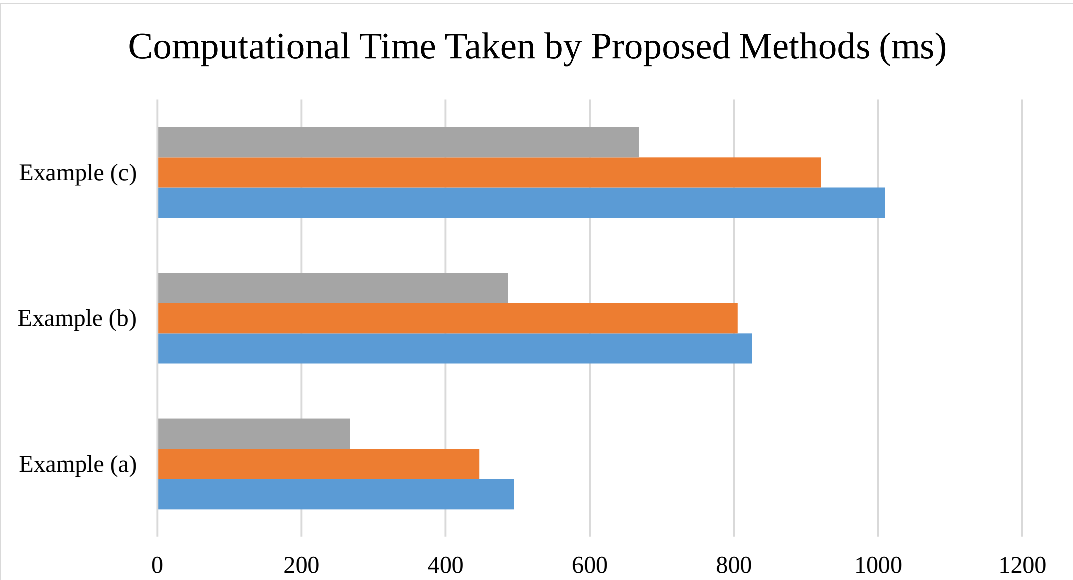
Meanwhile, the experimental results are presented in Figures 5 and 6 respectively. Figure 5 is presenting the findings based on the number of iterations involved for the three proposed iterative methods. Refer to Figure 5, it is clearly showed that the number of iterations generated by HSAOR is the least compared to another two iterative methods. It had reduced the number of iterations by approximately 36.8%, 37.6% and 12.7% respectively for the three test examples as compared to FSSOR iterative method. Besides, when compared with FSAOR iterative method, HSAOR iterative method also reduced the number of iterations by approximately 28.6%, 26.5% and 6.4% in the three examples.



	Example (a)	Example (b)	Example (c)
HSAOR	55	83	117
FSAOR	77	113	125
FSSOR	87	133	134

Figure 5: Number of iterations used by considered iterative methods

Computational time taken by the three proposed iterative methods is another important measurement to investigate the efficiency of an iterative method on solving this problem. The relationship between the number of iterations used and computational time taken is proportional, the lesser the number of iteration used, the corresponding computational time taken is lesser as well, which means the iteration process is shorter. The computational time taken by HSAOR iterative method when compared to FSSOR iterative method had improved approximately more than 50% in every text examples and approximately 40% for first two examples and 28% for the third examples when compared with FSAOR iterative method.



	Example (a)	Example (b)	Example (c)
HSAOR	267	487	668
FSAOR	447	805	921
FSSOR	495	825	1010

Figure 6: Computational time taken by considered iterative methods

In addition, the new blended output images are presented in Figure 7. Visually, the quality of all the new generated images is the same without any significant difference where it is all looks natural and realistic.

(a)

(b)

(c)

(i) FSSOR (ii) FSAOR (iii) HSAOR

Figure 7: New blended images by considered iterative methods

CONCLUSION

The Poisson image blending problem was successfully solved by the proposed iterative methods where half-sweep algorithm has the best performance compared to the full-sweep algorithm. HSAOR used the least number of iteration and shortest iteration process to complete the blending process. The superiority is mainly because the computational complexity had reduce in half-sweep algorithm where it only involved the implementation of half of all the pixels in the selected region during the iteration process. Therefore, half-sweep solver is another efficient alternative way to solve Poisson image blending problem.

ACKNOWLEDGEMENT

The authors are grateful for the fund received from Universiti Malaysia Sabah upon publication of this paper.

REFERENCES

[1] Gonzalez, R. C. and Woods, R. E. 2001. Digital Image Processing, Prentice Hall, New Jersey.

[2] Seo, D., Ho, J. and Vemuri, B. C. 2016. “Covariant Image Representation with Applications to Classification Problems in Medical Imaging,” International Journal of Computer Vision, 116(2): 190-209.

[3] Yang, W., Zheng, J., Cai, J., Rahardja, S. and Chen, C. W. 2009. “Natural and Seamless Image Composition with Color Control,” Journal IEEE Transactions on Image Processing, 18(11): 2584-2592.

[4] Perez, P., Gangnet, M. and Blake, A. 2003. “Poisson Image Editing,” ACM Transactions and Graphics, 313-318.

[5] Qin, C., Wang, S. and Zhang, X. 2008. “Image Editing without Color Inconsistency Using Modified Poisson Equation,” International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 397-401.

[6] Morel, J. M., Petro, A. B. and Sbert, C. 2012. “Fourier Implementation of Poisson Image Editing,” Pattern Recognition Letters, 33: 342-348.

[7] Hussain, K. F. and Kamel, R. M. 2015. “Efficient Poisson Image Editing,” Electronic Letters on Computer Vision and Image Analysis, 14(2): 45-57.

[8] Afifi, M. and Hussain, K. F. 2015. “MPB: A Modified Poisson Blending Technique,” Computational Visual Media, 1(4): 331-341.

[9] Deshpande, S. and Bhatia, S. 2012. “Image Inpainting Using Cloning Algorithms,” International Journal for Image Processing, 6(6): 397-402.

[10] Deokate, B. H., Patil, Dr. P. M. and Lonkar, P. B. 2015. “Image Mapping and Object Removal Using ADM in Image Inpainting: Review,” International Journal on Recent and Innovation Trends in Computing and Communication, 3(3): 931-933.

[11] Agrawal, N., Sinha, P., Kumar, A. and Bagai, S. 2015. “Fast & Dynamic Image Restoration using Laplace Equation Based Image Inpainting,” Journal of Undergraduate Research and Innovation, 1(3): 115-123.

[12] Bardeji, S. G., Figueiredo, I. N. and Sousa, E. 2015. “Image Contrast Enhancement using Split Bregman Method,” Proceeding of the 5th Eccomas Thematic Conference on Computational Vision and Medical Image Processing, 63-67.

[13] Gong, Y, and Sbalzarini, I. F. 2015. “Image Enhancement by Gradient Distribution Specification,” Asian Conference on Computer Vision 2014, 47-62.

[14] Levin, A., Zomet, A., Peleg, S. and Weiss, Y. 2004. “Seamless Image Stitching

in the Gradient Domain," 8th Europeen Conference on Computer Vision, 4: 377-389.

[15] Agarwala, A. 2007. "Efficient Gradient-Domain Compositing Using Quadtrees," ACM Transactions on Graphics, 26(3): 1-4.

[16] Jia, J., Sun, J., Tang, C. K. and Shum, H. Y. 2006. "Drag-and-Drop Pasting," ACM Transactions on Graphics, 25(3): 631-636.

[17] Zhang, Y., Ling, J., Zhang, X. and Xie, H. 2014. "Image copy-and-paste with Optimized Gradient," The Visual Computer, 30: 1169-1178.

[18] Philip, S., Summa, B., Bremer, P. T. and Pascucci, V. 2011. "Parallel Gradient Domain Processing of Massive Images," 11th Eurographics Symposium on Parallel Graphics and Visualization, 11-19.

[19] Lee, D. W., Yoo, S. B., Choi, M., Ra, J. B. and Kim, J. 2012. "Block Poisson Method and Its Application to Large Scale Image Editing," 19th IEEE International Conference on Image Processing, 2121-2124.

[20] Martino, J. M. D., Facciolo, G. and M.-Llopis, E. 2016. "Poisson Image Editing," Image Processing On Line, 6:300-325.

[21] Saudi, A. and Sulaiman, J. 2012. "Robot Path Planning Using Four Point-Explicit Group Via Nine-Point Laplacian (4EG9L) Iterative Method," Procedia Engineering of International Symposium on Robotics and Intelligent Sensors 2012, 182-188.

[22] Saudi, A. and Sulaiman, J. 2016. "Path Planning Simulation Using Harmonic Potential Fields Through Four Point-EDGSOR Method Via 9-Point Laplacian," Jurnal Teknologi, 78: 12-24.

[23] Young, D. M. 1954. "Iterative Methods for Solving Partial Differential Equations of Elliptic Type," Transactions of the American Mathematical Society, 76: 92-111.

[24] Eng, J. H., Saudi, A. and Sulaiman, J. 2017. "Application of SOR Iteration for Poisson Image Blending," International Conference on High Performance Compilation, Computing and Communications, 60-64.

[25] Hadjidimos, A. 1978. "Accelerated Overrelaxation Method," Mathematics of Computation, 32: 149-157.

[26] Test examples are available at https://pixabay.com.