International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 10, No.1 (Nov-2021)

http://dx.doi.org/10.12785/ijcds/100198

Accelerated Red-Black Strategy for Image Composition Using Laplacian Operator

Nordin Saad[1], Andang Sunarto[2] and Azali Saudi[1]

[1]*Intelligent Robotics Lab, Faculty of Computing and Informatics, Universiti Malaysia Sabah, Kota Kinabalu, Malaysia*
[2]*Institut Agama Islam Negeri Bengkulu, Indonesia*

Received 22 April. 2020, Revised 16 March. 2021, Accepted 3 April. 2021, Published 25 Nov. 2021

Abstract: Digital image composition deals with the problem of embedding portions of the source image into the target image to generate a single desirable image seamlessly. The aim is to produce a new image that combines both source and target images so that the seam between the two images is less noticeable. Image composition based on numerical differentiation using the Laplacian operator to obtain the solution of the Poisson equation is presented. The proposed method employs a red-black strategy to speed up the computation by using an accelerated parameter to the existing relaxation method. This modified variant of the relaxation method known as the Modified Accelerated Over-Relaxation (MAOR) method is derived from the existing Modified Successive Over-Relaxation (MSOR) method. Several examples were tested to examine the effectiveness of the proposed method. The results showed that both modified accelerated variants performed faster than their corresponding standard modified successive variants.

Keywords: Red-black Strategy, Accelerated Iterative Method, MSOR, MAOR, Poisson Equation, Laplacian Operator, Image Composition, Finite Difference

1. INTRODUCTION

Image composition involves computation of image gradients of both source and target images. This powerful image editing technique is often capable of composing two images with very minimal noticeable seams. A similar approach is also used for various other image manipulations, including natural image matting [1], image stitching [2], surface reconstruction [3], image sharpening [4], image completion [5] and inpainting [6].

This paper describes the solution to the image composition problem using numerical differentiation technique. The gradients of the images are obtained by solving the Poisson equation. The numerical approach transforms the problem into a linear system. Often, the generated linear system is large and sparse, thus a numerical iterative method is employed to avoid huge memory consumption. The addition of an acceleration parameter to the resulting finite difference discretization provides an additional parameter to speed up the computation. Additionally, the proposed method used modified variants of the relaxation methods in which the red-black strategy is employed. With this red-black strategy, image gradients for pixels assigned to red and black are computed independently. Thus, the proposed method is very suitable for parallel implementations. Several similarity indexes are used to ensure high quality generated images.

The main contributions of this work include: (1) demonstrating the effectiveness of an advanced numerical iterative method using the modified variant with additional acceleration parameter for image editing based on the Poisson equation, (2) proposing red-black strategy for robust computational iteration procedure that is suitable for implementation on machines capable of performing parallel processing, and (3) conducting extensive image similarity comparison to show that the proposed method is capable of generating highly similar final composited image to the images generated using standard existing methods.

2. RELATED WORKS

In 2020, Qiao et al. [1] proposed a deep learning based approach to the image matting problem. In their work, they constructed a large-scale image matting dataset comprised of 59,600 training images and 1000 test images. With such a large set of datasets, it improved the robustness of the proposed hierarchical structure aggregation model. Yun et al. [2] proposed an image stitching method to stitch multiple images to generate a panoramic image with a piecewise rectangular boundary. They addressed the limitations of the existing methods by formulating image stitching with regular boundaries in a unified optimization. The results showed that their method efficiently produced visually pleasing panoramas with regular boundaries and unnoticeable distortions. Xie et al. [3] introduced an approach for surface reconstruction by

E-mail address: nordin@ums.edu.my, andangs@iainbengkulu.ac.id, azali@ums.edu.my

preserving the surface discontinuity in a discrete geometry way. Experiments showed that the proposed method outperformed the existing methods on an extensive 3D dataset, in terms of mean angular error and computational complexity. Image sharpening based on partial differential equations was proposed by Wang et al. [4] by employing the Hausdorf derivative Laplacian operator. It was shown in the numerical results that the proposed operator outperforms the high-pass filtering, the Robert operator and the traditional integer-order Laplacian operator. In [5], Zdunek and Sadowski proposed an approach to model the incomplete images with overlapping blocks of Tucker decomposition representations where the factor matrices are determined by a hybrid version of the Gaussian radial basis function and polynomial interpolation. In the experiments, it was demonstrated that the proposed approach considerably outperformed the baseline and state-of-the-art low-rank completion methods on various image completion and resolution up-scaling problems. An image completion method based on Generative Adversarial Networks was proposed by Jiang et al. [6]. The proposed model is based on an auto-encoder architecture in which the algorithm contains one generator, one global discriminator, and one local discriminator. The generator is used for inpainting the missing area. The global discriminator evaluates whether the repair result has global consistency, and the local discriminator identifies whether the repair area is correct. The results showed that the proposed model was capable of dealing with large scale missing pixels and was able to produce high-quality realistic completion images.

Image manipulation using the Poisson equation by combining image gradients was first proposed by Perez et al. [7], where the image editing problem was formulated in variational terms. Since then, several works on image processing based on image gradients have been proposed. Du et al. [8] proposed a method for surface reconstruction algorithm that aims to obtain the minimum absolute error between the input and the solution gradient field. The same approach was also proposed by Reddy et al. [9], in which the algorithm attempts to find the gradient field that best fits the input gradient field using linear algebra and graph analogy. Tao et al. [10] suggested a compositing technique that reduces bleeding artifacts in the composited images. First, they define the boundary gradients in such a way that the gradient field produced is integrable, and second, they control the integration process to concentrate residuals where they are less visible by modifying the modified Poisson equation to trace the targeted residuals in less observable textured regions.

Sadek et al. [11] proposed an algorithm based on a variational model to support video editing using a gradient-based method. The approach utilized an optimal flow to propagate chosen features in the gradient domain. The proposed approach was capable of handling fast and abrupt illumination changes in time and smooth illumination changes in space. It was presented in their report that the model was capable of handling video containing a large number of frames and relatively complex sequences.

More recently, image editing based on Fourier implementation was proposed by Morel et al. [12]. The proposed method provides an exact and fast Poisson solver when it involved the complex shape of a region of interest (ROI). The authors also proposed an automatic method for selecting the ROI.

Limare et al. [13] presented the Retinex algorithm for solving the Poisson equation using Fourier implementation. The proposed solution applied Neumann boundary conditions. In this method, the gradient whose modulus is below a certain threshold is set to zero. The proposed method successfully removed the small details and shades in the new image.

More recently, Hussain and Kamel [14] proposed an efficient method for solving the Poisson equation. The proposed method utilized the image pyramid and divide-and-conquer approaches. The Poisson equation was solved for three pyramid levels in this method. Moreover, a modified Poisson blending (MPB) solver had been proposed by Afifi and Hussain [15] in order to overcome the issues of bleeding artifacts and colour bleeding. To overcome the bleeding artifacts problem which occurred in the traditional methods, the proposed method also considered the boundary pixels of the images.

The solution to the linear system generated by the Poisson equation can be computed using analytical or direct methods. Only a few research had employed iterative method in solving the Poisson equation for image editing, such as in [7] and [16]. In our previous works, the MSOR [17] and block iterative method were implemented [18]. Therefore, in this work, four iterative methods that support parallel processing are implemented. Apart from image editing, iterative methods such as SOR and AOR had been successfully applied to solve path planning problem [19] and solving partial differential equations such as in [20] and [21].

3. IMAGE COMPOSITION

Any digital image is constructed based on a 2D array of pixels in which each pixel is assigned its own intensity value or brightness. The coordinate system of an image is the rotated conventional 2D Cartesian coordinate system by 90^{o} in clockwise direction. Each pixel in the image is formed by mixtures of red, green, and blue colors. Image resolution and size are represented by the total number of rows and columns.

In the image composition process, each pixel value of an image will be stored in a finite grid of a 2D array as illustrated in Figure 1. The composition process begins by selecting the desired region of interest **S** from a source image **G**. Next, the boundary of the chosen region is selected and denoted by ∂**S**. The extracted region is then placed into the desired location of a target image **F***. This process will generate a new output image **F**. The composition procedure continues by first obtaining the vector field **V**, which is obtained from the source image **S**. By applying Poisson equation as a constraint, a set of new pixel values inside region ∂**S** will be computed through minimization of the gradient difference value of vector **V** and the desired region of the target image **F***.

This minimization procedure can be expressed as follows:

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

The intensity values at the boundary are assigned the same values to generate a seamless image. The intensity values are extracted from the image itself. It will be used as the initial values for the generated linear system constructed from the discretization Poisson equation via the finite difference method. The details are discussed in the following section.

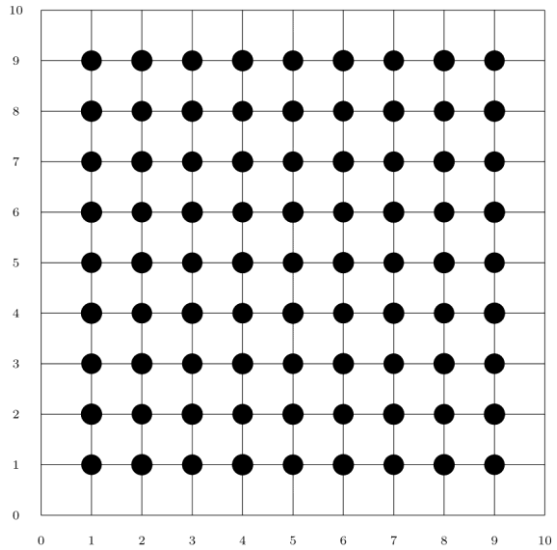

Figure 1. Finite grid of size n = 10.

4. Formulation of the Proposed Method

To implement the considered iterative methods, the Poisson equation is first discretized using the finite difference method. By employing the Laplacian operator, the two-dimensional Poisson equation can be defined as,

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

Equation (4) is then discretized using five-point approximation equation to obtain

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

where $h = \Delta x = \Delta y$. The iterative scheme for equation (5) can be written as,

$$U_{i,j}^{(k+1)} = \frac{\omega}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k+1)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k+1)} - h^2 f_{i,j}\right] + (1-\omega) f_{i,j}^{(k)} \tag{6}$$

where a weighted parameter, ω is added. By adding another accelerated parameter, λ into Equation (6), it can be redefined as,

$$U_{i,j}^{(k+1)} = \frac{\omega}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k)} - h^2 f_{i,j}\right] + \frac{\lambda}{4}\left[U_{i-1,j}^{(k+1)} - U_{i-1,j}^{(k)} + U_{i,j-1}^{(k+1)} - U_{i,j-1}^{(k)}\right] + (1-\omega) f_{i,j}^{(k)} \tag{7}$$

The solution of the minimization problem (1) is obtained by solving the unique solution of the Poisson equation with Dirichlet boundary condition,

$$\Delta f = \text{div } v \text{ at } O \text{ with } \quad f|_{\partial O} = f^*|_{\partial O} \tag{2}$$

To generate a new image seamlessly, the gradient of vector **V** is extracted from the source image **G**. Thus, Equation (2) can be rewritten as,

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

The Δ in Equation (3) denotes the Laplacian operator.

with $1 \le \omega < 2$ and $1 \le \lambda < 2$. Equation (6) is used to implement the iteration procedure known as Successive Over-Relaxation (SOR) [22]. While equation (7) is known as Acceleration Over-Relaxation (AOR) [23] iterative method.

A. Red-Black Strategy

The standard ordering of computing each node in the finite grid (Figure 1) with the SOR or AOR methods is not suitable for parallel processing, since the computation can be conducted only in sequential order. Thus, the red-black [24] ordering is suggested as illustrated in Figure 2. With red-black ordering, the computations of nodes in the finite grid are executed in red-only and black-only phases. In the red only phase at iteration (k+1), the computation only involves red nodes, and in the black-only phase, the remaining black nodes are processed. By applying the finite difference equation (5), then using SOR, the red-only phase uses the following iteration scheme,

$$U_{i,j}^{(k+1)} = \frac{\omega}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k)} - h^2 f_{i,j}\right] + (1-\omega) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ even} \tag{8}$$

and during the black-only phase, the scheme is as follows,

$$U_{i,j}^{(k+1)} = \frac{\omega}{4}\left[U_{i+1,j}^{(k+1)} + U_{i-1,j}^{(k+1)} + U_{i,j+1}^{(k+1)} + U_{i,j-1}^{(k+1)} - h^2 f_{i,j}\right] + (1-\omega) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ odd.} \tag{9}$$

Equations (8) and (9) represent the Red-Black SOR (RBSOR) method.

Accordingly, the AOR iteration schemes for the red-black ordering (RBAOR method) are

$$U_{i,j}^{(k+1)} = \frac{\omega}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k)} - h^2 f_{i,j}\right] + (1-\omega) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ even} \tag{10}$$

and

$$U_{i,j}^{(k+1)} = \frac{\omega}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k)} - h^2 f_{i,j}\right] + \frac{\lambda}{4}\left[U_{i+1,j}^{(k+1)} - U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} - U_{i-1,j}^{(k)} + U_{i,j+1}^{(k+1)} - U_{i,j+1}^{(k)} + U_{i,j-1}^{(k+1)} - U_{i,j-1}^{(k)}\right] + (1-\omega) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ odd.} \tag{11}$$

for the red-only and black-only phases, respectively.

Clearly, the red-only phase involves independent computations that are executed simultaneously and

similarly for the black-only phase execution. Note that parallel processing can be achieved since the computations of the red node rely on its four black neighboring nodes and vice versa, as shown in Figure 3 (a) and (b), respectively. If red nodes are computed first, their updated values are computed based on the previous values of black nodes. Accordingly, the computations of black nodes will be based on the updated values of red nodes.

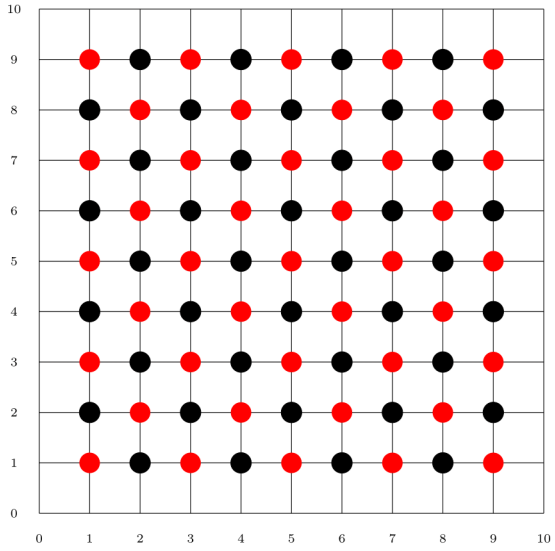

Figure 2. Finite grid of red-black strategy.

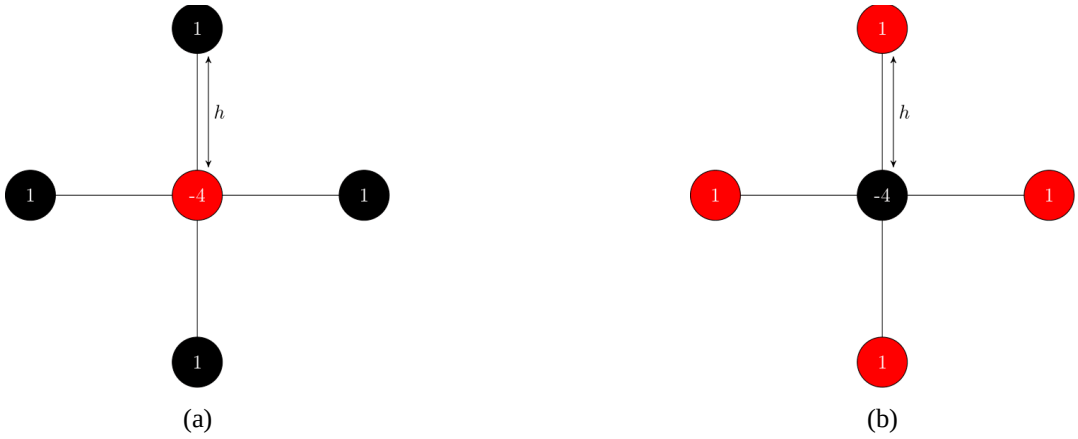

Figure 3. Computational molecules, (a) red and (b) black nodes.

B. Modified Iterative Methods

The Modified Successive Over-Relaxation (MSOR) method is described in [25] and given by

$$U_{i,j}^{(k+1)} = \frac{\alpha}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k)} - h^2 f_{i,j}\right] + (1-\alpha) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ even} \tag{12}$$

and during the second phase, the scheme is as follows,

$$U_{i,j}^{(k+1)} = \frac{\beta}{4}\left[U_{i+1,j}^{(k+1)} + U_{i-1,j}^{(k+1)} + U_{i,j+1}^{(k+1)} + U_{i,j-1}^{(k+1)} - h^2 f_{i,j}\right] + (1-\beta) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ odd,} \tag{13}$$

where α and β are fixed parameters such that $0 < \alpha < 2$ and $0 < \beta < 2$.

Therefore, the Modified Accelerated Over-Relaxation (MAOR) method can be derived and written as

$$U_{i,j}^{(k+1)} = \frac{\alpha}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k)} - h^2 f_{i,j}\right] + (1-\alpha) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ even} \tag{14}$$

and

$$U_{i,j}^{(k+1)} = \frac{\beta}{4}\left[U_{i+1,j}^{(k)} + U_{i-1,j}^{(k)} + U_{i,j+1}^{(k)} + U_{i,j-1}^{(k)} - h^2 f_{i,j}\right] + \frac{\lambda}{4}\left[U_{i+1,j}^{(k+1)} - U_{i+1,j}^{(k)} + U_{i-1,j}^{(k+1)} - U_{i-1,j}^{(k)} + U_{i,j+1}^{(k+1)} - U_{i,j+1}^{(k)} + U_{i,j-1}^{(k+1)} - U_{i,j-1}^{(k)}\right] + (1-\beta) f_{i,j}^{(k)} \quad \text{for all } i+j \text{ odd.} \tag{15}$$

Consequently, by applying red-black ordering, Equations (12) and (14) can be computed in the first phase, and

Equations (13) and (15) are computed during the second phase.

In this work, these four methods are implemented in Java. The iteration procedure keeps executing until the error rate, ε, calculated using the square root of the sum of squares, is less than 1.

The main steps of the finite difference approach of the four methods are described in Algorithm 1. During the composition process, the linear system (4) was solved three times for R, G, and B colors separately using the four considered methods.

Algorithm 1: The iteration procedure to obtain the solution of Poisson

Input: Source image **G**, Target image **F***, Regions of Interest (ROI) **S**, and boundary of ROI ∂**S**

Output: New edited image **F**

1 Load Source and Target Images

2 Create a mask for the selected ROI in source image **G**

3 **Place the extracted region inside the** ROI into the desired locaction of target image **F***

4 Setup *U* and *V* (to store the previous and updated pixel values inside ROI using mask created in Step 2

5 Initialize values for ω, λ, α, β.

6 Compute red nodes where $(i+j)$ is even using the previous values of black nodes in *U*.

7 Compute black nodes where $(i+j)$ is odd using the updated values of red nodes stored in *V*.

8 Check converge criterion and copy the updated values in *V* into *U*.

9 If the convergence criterion is satisfied, terminate the iteration procedure, otherwise repeat step 5.

10 F* = *V*

SET A

SET B

SET C

(a) (b)

Figure 4. Pairs of target (a) and source (b) images.

TABLE I. THE NUMBER OF PIXELS IN THE SELECTED REGION

Selected region	Balloon	Goose	Eland
Number of pixels	12,392	7,600	7,939

5. RESULTS OF EXPERIMENTS

The experiments involve three sets of images of various sizes being tested. Figure 4 shows the images of sets A, B, and C. Each set is a pair of source and target images. The four methods (the standard red-black variants: RBSOR and RBAOR, and their corresponding modified variants: MSOR and MAOR) were applied to the three sets of images with different numbers of unknown pixels, as shown in Table I.

A. Computational Cost Measurements

This section presents the computational time of the composition process for the three sets of images using the tested methods, as shown in Tables II-IV. The results in these three tables show that MAOR gives fewer iterations than the other methods. Consequently, the MAOR method provides the best performance in terms of computational time for all three sets of images. The modified variants of the MSOR and MAOR methods outperform their corresponding standard non-modified variants, RBSOR and RBAOR, in which the number of iterations has been reduced by approximately 7-10%, respectively. The AOR variants perform better than the SOR variants by decreasing the iteration count by approximately 18-22%.

TABLE II. COMPUTATIONAL RESULTS FOR IMAGE SET A

METHODS	Number of iterations, k	Computational time, t
RBSOR	886	12s 692ms
RBAOR	746	10s 541ms
MSOR	826	11s 484ms
MAOR	678	9s 425ms

TABLE III. COMPUTATIONAL RESULTS FOR IMAGE SET B

METHODS	Number of iterations, k	Computational time, t
RBSOR	486	5s 993ms
RBAOR	388	4s 840ms
MSOR	448	5s 429ms
MAOR	347	4s 261ms

TABLE IV. COMPUTATIONAL RESULTS FOR IMAGE SET C

METHODS	Number of iterations, k	Computational time, t
RBSOR	455	6s 20ms
RBAOR	375	5s 35ms
MSOR	422	5s 516ms
MAOR	339	4s 498ms

B. Image Quality Measurements

In the composition process, image quality measurement plays an important role. It can be utilized to compare the final images generated by the tested method with the ideal output images produced by the standard Gauss-Seidel method. In this work, the statistical analysis of variance (ANOVA) method [26] is used where the quality is measured using seven metrics.

Mean Square Error (MSE) is used to measure distortion and can be expressed as follows:

$$MSE = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}(A_{ij} - B_{ij})^2. \tag{16}$$

The variable (m x n) denotes the image size, A is the reference image, and B is the processed image. The MSE compares the reference and processed images, in which smaller value represents better result. The two images are identical if the MSE value is equal 0.

The Peak Signal-to-Noise Ratio (PSNR) value is used to measure the ratio of power to the noise. It is measured in decibel scale. Two images with the same value of PSNR are identical. PSNR can written as below:

$$PSNR = 10\log\frac{255^2}{MSE} \tag{17}$$

The Structural Similarity Index Measure (SSIM) tests the similarity between the reference and processed images [27]. The aim is to obtain SSIM value that is equal to 1, which means the two images are identical. The SSIM formulae is given as [28]

$$SSIM(x,y) = \frac{(2\varphi_x\varphi_y + C_1)(2\tau_{xy} + C_2)}{(\varphi_x^2 + \varphi_y^2 + C_1)(\tau_x^2 + \tau_y^2 + C_2)} \tag{18}$$

where $\{\varphi_x, \varphi_y\}$ denotes the mean itensity, $\{\tau_x, \tau_y\}$ denotes standard deviation set of x and y images, respectively. The parameter τ_{xy} denotes the cross correlation. Both parameters C_1 and C_2 are small constants value to ensure stability, particularly when the denominator is too close to zero [29]. SSIM value equal 1 means the two images are identical.

The formulae of Structural Content (SC) metric is given as follows

$$SC = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(A_{ij})^2}{\sum_{i=1}^{m}\sum_{j=1}^{n}(B_{ij})^2} \tag{19}$$

Normalized Absolute Error (NAE) quality measure can be expressed as follows:

$$NAE = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(|A_{ij} - B_{ij}|)^2}{\sum_{i=1}^{m}\sum_{j=1}^{n}(A_{ij})^2} \tag{20}$$

Average Difference (AD) provides the average of change for the two images and is written as below:

$$AD = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}[A_{ij} - B_{ij}] \tag{21}$$

Maximum Difference (MD) measures the maximum difference of pixel value between the two images. MD is defined as follows:

$$MD = \max(|A_{ij} - B_{ij}|) \tag{22}$$

The aim of MD is to obtain a very small difference value. If the values obtained from equations (19) – (22) are equal, it represents the images generated by the tested methods are identical.

A comparison between the generated images from the four tested methods for all sets of images A, B, and C is illustrated in Figures 5, 6, and 7, respectively. Visually, the images generated by each set are identical. Tables V, VI, and VII show the values of the seven measurement metrics. The MSE and SSIM values are approximately near 0 and 1, respectively. Whereas, the values obtained from PSNR, SC, NAE, AD and MD metrics are nearly equal. These values mean that the output images generated by the four different methods are almost identical. Based on the obtained MSE and PSNR values shown in the tables, the output images generated by the SOR variants are slightly more identical to the ideal output generated by the Gauss-Seidel method compared to the images generated by the AOR variants. Since the differences are very small, the visual appearances of the composed images generated by both variants are identical and, thus the visual differences are hardly noticeable.

SET A

RBSOR

RBAOR

MSOR

MAOR

Figure 5. The images generated by set A.

SET B

RBSOR

RBAOR

MSOR

MAOR

Figure 6. The images generated by set B.

SET C

RBSOR

RBAOR

MSOR

MAOR

Figure 7. The images generated by set C.

TABLE V. SIMILARITY MEASUREMENT FOR IMAGE SET A

METHODS	MSE	PSNR	SSIM	SC	NAE	AD	MD
RBSOR	0.318310	53.10230	0.99997	0.99767	0.00102	0.13993	8.0
RBAOR	0.378105	52.35468	0.99997	0.99746	0.00112	0.15265	8.0
MSOR	0.332355	52.91478	0.99997	0.99762	0.00105	0.14300	8.0
MAOR	0.393595	52.18031	0.99997	0.99741	0.00114	0.15588	9.0

TABLE VI. SIMILARITY MEASUREMENT FOR IMAGE SET B

METHODS	MSE	PSNR	SSIM	SC	NAE	AD	MD
RBSOR	0.00388	72.24248	0.99999	0.99994	3.57641	0.00692	3.0
RBAOR	0.00434	71.75591	0.99999	0.99993	3.88134	0.00751	3.0
MSOR	0.00399	72.12652	0.99999	0.99993	3.65652	0.00708	3.0
MAOR	0.00443	71.67167	0.99998	0.99992	3.95887	0.00766	3.0

TABLE VII. SIMILARITY MEASUREMENT FOR IMAGE SET C

METHODS	MSE	PSNR	SSIM	SC	NAE	AD	MD
RBSOR	0.06301	60.13671	0.99999	0.99891	4.32086	0.05003	4.0
RBAOR	0.07401	59.4382	0.99999	0.99881	4.73282	0.05480	5.0
MSOR	0.06471	60.02109	0.99999	0.99889	4.39513	0.05089	5.0
MAOR	0.07844	59.18570	0.99999	0.99876	4.86237	0.05630	5.0

6. CONCLUSIONS

This paper presents the implementation of four iterative methods using red-black ordering to obtain the solution of the Poisson equation to solve the image composition problem. Based on the results, it is clearly shown that the MAOR outperforms the other methods in terms of computational time. It is also shown that the AOR variants perform faster than their corresponding SOR variants. The SOR variants, however, are slightly better than AOR variants in terms of similarity to the ideal output images produced by the Gauss-Seidel method, but since the differences are very small, this slight drawback is hardly noticeable visually. With red-black ordering, the red and black nodes are computed alternately, in which the computational molecules for the nodes of the two colors are independent. Therefore, the proposed modified variants are very suitable for parallel implementation. From the comparison obtained with the seven measurement metrics, it can be observed that all images generated by the tested methods are identical. This means that the quality of images produced by the proposed methods are almost identical, in which no significant differences are noticeable.

ACKNOWLEDGMENT

This work is supported by the Universiti Malaysia Sabah under grant no SGA0117-2019.

REFERENCES

[1] Y. Qiao, L. Y. Liu, X. Yang, D. Zhou, M. Xu, Q. Zhang and X. Wei, Attention-Guided Hierarchical Structure Aggregation for Image Matting, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13676-13685.

[2] Y. Zhang, Y. Lai and F. Zhang, Content-Preserving Image Stitching with Piecewise Rectangular Boundary Constraints, IEEE Transactions on Visualization and Computer Graphics, 2020, doi: 10.1109/TVCG.2020.2965097.

[3] W. Xie, M. Wang, M. Wei, J. Jiang, J. Qin, Surface Reconstruction From Normals: A Robust DGP-Based Discontinuity Preservation Approach, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 5328-5336.

[4] F. Wang, W. Chen, and L. Qiu, Hausdorff derivative Laplacian operator for image sharpening. Fractals, vol. 27, no. 3, 2019, pp. 1950060.

[5] R. Zdunek and T. Sadowski, Image Completion with Hybrid Interpolation in Tensor Representation. Appl. Sci., vol. 10, no. 3, 2020, pp. 797.

[6] Y. Jiang, J. Xu, B. Yang, J. Xu and J. Zhu, Image Inpainting Based on Generative Adversarial Networks, IEEE Access, vol. 8, pp. 22884-22892, 2020, doi: 10.1109/ACCESS.2020.2970169.

[7] P. Perez, M. Gangnet, and A. Blake, Poisson image editing, ACM Transactions and Graphics, vol. 22, 2003, pp. 313-318.

[8] Z. Du, A. Robles-Kelly, and F. Lu, Robust surface reconstruction from gradient field using the L1 norm, in Digital Image Computing Techniques and Applications, 9th Biennial Conference of the Australian Pattern Recognition Society on, IEEE, 2007, pp. 203-209.

[9] D. Reddy, A. Agrawal, and R. Chellappa, Enforcing integrability by error correction using l1-minimization, 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[10] M. W. Tao, M. K. Johnson, and S. Paris, Error-tolerant image compositing, International Journal of Computer Vision, 103, 2013, pp. 178-189.

[11] R. Sadek, G. Facciolo, P. Arias, and V. Caselles, A Variational Model for GradientBased Video Editing, International Journal of Computer Vision, vol. 103, 2013, pp. 127-162.

[12] J-M. Morel, A. B. Petro, and C. Sbert, Fourier implementation of Poisson image editing, Pattern Recognition Letters, vol. 33, 2012, pp. 342-348.

[13] N. Limare, A. B. Petro, C. Sbert, and J-M. Morel, Retinex Poisson Equation: a Model for Color Perception, Image Processing On Line, vol. 1, 2011, pp. 1-12.

[14] K. F. Hussain and R. M. Kamel, Efficient poisson image editing, Electronic Letters on Computer Vision and Image Analysis, vol. 14, no. 2, 2015, pp. 45-57.

[15] M. Afifi and K. F. Hussain, MPB: A modified poisson blending technique, Computational Visual Media, vol. 1, no. 4, 2015, pp. 331-341.

[16] J. M. D. Martino, G. Facciolo, and E. Meinhardt-Llopis, Poisson image editing. Image Processing, 2006, On Line 6, pp. 300-325.

[17] E. J. Hong, A. Saudi, and J. Sulaiman, Numerical assessment for poisson image blending problem using MSOR iteration via five-point laplacian operator, Journal of Physics: Conference Series, vol. 890, no. 012010, 2017.

[18] E. J. Hong, A. Saudi, and J. Sulaiman, Performance analysis of the explicit decoupled group iteration via five-point rotated laplacian operator in solving poisson image blending problem, Indian Journal of Science and Technology, vol. 11, no. 12., 2018.

A. Saudi and J. Sulaiman, Indoor Path Planning using Harmonic Functions via Half-Sweep Arithmetic Mean Method, International Journal of Computing and Digital Systems, 2019, no.5, pp. 523-528.

[19] N. A. Syafiq, M. Othman, N. Senu, and F. Ismail, An experimental study of the modified accelerated overrelaxation (MAOR) scheme on stationary helmholtz equation, Journal of Physics: Conference Series, vol. 1366, no. 1, 2019, pp. 012093.

[20] K. Ghazali, J. Sulaiman, Y. Dasril and D. Gabda, Newton Method with AOR Iteration for Finding Large Scale Unconstrained Minimizer with Tridiagonal Hessian Matrices, J. Phys.: Conf. Ser., 2019, 1298: 012002.

[21] Y. Saad, Iterative methods for linear systems of equations: A brief historical journey, 2019, arXiv:1908.01083.

[22] A. Hadjidimos and T. Michael, The solution of the linear complementarity problem by the matrix analogue of the accelerated overrelaxation iterative method, Numerical Algorithms, 2016, vol. 73, no. 3, pp. 665-684.

[23] B. Johnson, S. Thomas, and R. J. Sheeba, A High-Performance Dense Optical Flow Architecture Based on Red-Black SOR Solver, J. Sign. Process. Syst. 2020, vol. 92, pp. 357–373.

[24] J. V. L. Chew and J. Sulaiman, Application of MSOR iteration with Newton scheme for solutions of 1D nonlinear porous medium equations, AIP Conference Proceedings, 2016, 1739: 020017,

[25] S. H. A. Hashim, F. A. Hamid, J. J. Kiram, J. Sulaiman, The relationship investigation between factors affecting demand for broadband and the level of satisfaction among broadband customers in the South East Coast of Sabah, Malaysia, Journal of Physics: Conference Series, 2017, 890: 012149, doi: 10.1088/1742-6596/890/1/012149.

[26] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Transactions on Image Processing, 2004, vol 13, no. 4, pp. 600-612.

[27] F. Memon, M. A. Unar, and S. Memon, Image quality assessment for performance evaluation of focus measure operators, 2016, arXiv:1604.00546.

[28] K. Thung and P. Raveendran, A survey of image quality measures, 2009, doi: 10.1109/TECHPOS.2009.5412098.

Nordin Saad obtained his master's degree in Information Technology from Universiti Kebangsaan Malaysia (UKM) and his bachelor's degree in Information Technology from Universiti Utara Malaysia (UUM). He joined Universiti Malaysia Sabah (UMS) as a lecturer in Information Technology in May 1999. Currently, he is a senior lecturer in the Faculty of Computing and Informatics, UMS. His research interests include ICT infrastructure, IoT, AI and Smart Agriculture. He also works on the modelling and simulation of image compositions.

Andang Sunarto obtained his PhD from Universiti Malaysia Sabah, a master's degree in Computer Science from Universitas Gajah Mada and a bachelor's degree in Statistics from Universitas Islam Indonesia. He joined the Institut Agama Islam Negeri, Bengkulu, Indonesia, as a senior lecturer (Computer Science and Mathematics) in September 2006. He is a researcher and was appointed as a fellow working in the fields of Numerical Analysis, Algorithms, C++ Programming, and Scientific Calculation, Modelling and Simulation. His research employs the application of IoT and Numerical Analysis to improve models of fractional partial differential equations. He also works on the modelling and simulation of numerical analysis.

Azali Saudi holds a PhD degree in Mathematics from Universiti Malaysia Sabah (UMS), a Master's degree in Artificial Intelligence from the University of Edinburgh, and a Computer Science degree from Universiti Kebangsaan Malaysia. He is currently attached as Associate Professor at the Faculty of Computing and Informatics, UMS. He also acts as the Director of EcoCampus Management Centre at UMS. His research interests include the areas of computational mathematics, artificial intelligence, and robot vision. He also leads research projects related to intelligent robotics, computer vision, deep learning, and computational mathematics.