Numerical Solution of Non-Linear Diffusion Equation in Image Blurring Using Two-Point EGSOR Iterative Method

N A Basran¹, J H Eng², A Saudi³, J Sulaiman⁴

¹,²,⁴Mathematics with Economics Programme, Faculty of Science and Natural Resources, Universiti Malaysia Sabah,
³Knowledge Technology Research Unit, Faculty of Computing and Informatics, Universiti Malaysia Sabah, Kota Kinabalu, Malaysia,

E-mail: nurulafiqahbasran@yahoo.com

Abstract. The non-linear diffusion equation is known to be a significant application in solving image processing issues. The equations provided the image filtering techniques that blurring the image without degrade the edge information which is also one of crucial study in computer vision. Nonetheless, an intense amount of computations is needed in filtering the image as the sizes that keep getting bigger. Along these lines, this paper constructs an analysis to speed up the required computation in solving the developed linear system with the faster iterative method, i.e. two-point Explicit Group Successive Over-Relaxation or known as 2-EGSOR. For the performances comparison, the standard Gauss-Seidel (GS), Successive Over-Relaxation (SOR) and 2-EGSOR iterative method will be set to produce almost the similar quality image of Jacobi iterative method measured by using percentage error of the overall pixels difference. Subsequently, it is discovered the 2-EGSOR offers faster approach to blur the image compared the others iterative methods with the least iterations and computational time.

1. Introduction
Utilizing the non-linear diffusion equation on handling the image processing problems is first initiated by Perona and Malik [1]. They proposed this approach to introduce the algorithm for better edge detection. It has resolved the weakness of linear or heat diffusion PDE that cannot preserve the well-defined edge and crucial features of the images. This is because the rate of diffusion in linear heat equation is constant across the whole image domain that causes the entire image domain blurred out. A solution to this problem is to utilize the diffusion coefficient based on image gradients that can preserve edges information by using the nonlinear diffusion equation. Hence, Perona and Malik introduced the diffusion coefficient which known as edge stopping function to control the smoothing rate [2, 3]. Further details of this concept will be elaborate in section 2.1.

The non-linear equation is one of the Partial Difference Equations (PDEs) based image processing. Recently, there also a wide utilization and application of others PDEs in image processing and becoming a significant study to develop better algorithms and instruments in solving image processing problems. The PDEs based image processing methods has been used in various problems by past researchers, i.e image editing [4,5], image denoising [3], image enhancement [6], image segmentation [7], image smoothing [8] and edge-detection [9]. Besides, the efficient solution of the non-linear diffusion equation to solve various image processing issues has spurred Catte et al. [10] on upgrading the edge preserving
behavior of Perona Malik model. Due to the present of high amount noise the image cannot be denoise effectively, so they proposed to smoothen the image first by convolving with gaussian filter before the gradient computed. Meanwhile, an improved nonlinear diffusion algorithm is proposed by Wu and Zhong [11] for the purpose of solving the image denoising problem. Noise is referring to the random signal that appears as random speckles which significantly corrupting the image quality. Therefore, this new method has been verified as an efficient method to properly denoise the images compared to the other existing methods as it able reduce image noise while maintain important details better by using wavelet coefficient.

A few different strategies are likewise proposed for solving image processing problems using diffusion equation as mentioned before in this section. In any cases, the primary motivation behind this study is to numerically assess the performance of iterative methods in solving nonlinear diffusion equation. The past researchers have employed the block iterative methods in image processing field. They find out the block iterative methods outperform the standard point techniques in producing the seamless blended image [12-14] from the solution of Poisson equation which is one of the elliptic PDEs. Then, this study will look into the solution of non-linear diffusion equation for image blurring which is one of the parabolic PDEs. Other than that, block iterative methods also has widely been studied in path planning problems [15] and others numerical problems [16, 17].

Based on the performance of block iterative methods as discussed earlier, this paper examined the capability of 2-point Explicit Group Successive Over-Relaxation (2-EGSOR) iterative method to unravel the solution of non-liner diffusion equation for image blurring. This method also known is a family of block iterative by considering the two point in one group of iteration. The experimental results of Gauss-Seidel and SOR methods are resolved to distinguish the performances among those iterative methods and the proposed method. The execution of these iterative techniques is analyzed by the amount of iterations and the computational time taken.

The further details of the experiment are discussed and sorted out as follows. Section 2 clarify about the idea of nonlinear diffusion equation based on model developed by [1]. In section 3, the formulation of iterative method utilizing two-point Explicit Group Successive Over-Relaxation by Evans [18,19] to solve the linear system are shown. The discoveries and the examination of the numerical aftereffects of various iterative techniques to blur the pictures are expounded in section 4. At last, section 5 concluding the general paper.

2. Preliminaries
2.1 Nonlinear diffusion equation

As mentioned earlier, the non-linear diffusion equation is developed by Perona and Malik [1]. This method has upgraded the application of linear diffusion equation or otherwise called heat equation which significantly used in handling the various image processing issues of degradation [20], edge detection [21] and smoothing or blurring an image [9,22]. The problem of the linear heat equation filtering is it tends to blur the entire image without maintaining the shape of the significant features as the rate of diffusion constant throughout the image domain [23]. This equation can be shown as follow [1]:

\[ I_t = \text{div}(d(x,y,t)\nabla I(x,y)). \]  

(1)

Based on equation (1), \( \text{div} \) and \( \nabla I \) allude as the divergence operator and gradient magnitude operator respectively to the spatial of \( x \) and \( y \). Then, the symbol of \( d(x,y,t) = c(||\nabla I(x,y,t)||) \) known as diffusion coefficient. It controls the diffusion rate at any location of an image domain \( (x,y) \). The diffusion coefficient \( c(.) \) used in this study is given as:

\[ c(\nabla I) = \frac{1}{1 + \left( \frac{||\nabla I||}{K} \right)^2}. \]

(2)
The diffusion coefficient, \( c(.) \) as appeared in equation (2) is one of the edges stopping function suggested by [1]. Its known as edge stopping functions as its able control the diffusion processes with the diffusivity is upgrading in the inside of homogenous district while zero at the borderline. It gives the blurring impact lessen at any spots with high conceivable outcomes to be the borderline observed by the local gradient magnitude function, \(|\nabla I|\) where \( d(x,y,t) = c(|\nabla I|) \). Hence, the image domain will not entirely be blured out at the same rate and successfully maintain the edges features on the image.

2.2 Standard point Gauss-Seidel and SOR iterative method

The discretization of equation (1) through finite difference method by using implicit scheme has been developed to form the approximation equation of standard five-point as follow:

\[
(1 + \lambda c_U + \lambda c_D + \lambda c_L + \lambda c_R) I_{i,j,k+1} - \lambda c_D I_{i,j-1,k+1} - \lambda c_D I_{i,j+1,k+1} - \lambda c_L I_{i+1,j,k+1} - \lambda c_R I_{i-1,j,k+1} = I_{i,j,k}.
\]

(3)

where \( \lambda = \frac{\Delta t}{h^2} \) as \( h = \Delta x = \Delta y \). To simplify the equation (3), let \( \beta = 1 + \lambda c_U + \lambda c + \lambda c_L + \lambda c_R \), \( \delta_U = \lambda c_U \), \( \delta_D = \lambda c_D \), \( \delta_L = \lambda c_L \), and \( \delta_R = \lambda c_R \), then the following equation can be developed as shown follows:

\[
\beta I_{i,j,k+1} - \delta_U I_{i,j+1,k+1} - \delta_D I_{i,j-1,k+1} - \delta_L I_{i+1,j,k+1} - \delta_R I_{i-1,j,k+1} = I_{i,j,k}.
\]

(4)

The approximate equation (4) brings about the large linear system with sparse coefficient matrix which can be stated in algebraic equations as:

\[
\sum_{j=1}^{m} a_{ij} I_j = b_i, \quad i = 1, 2, 3, \ldots, m.
\]

(5)

Therefore, the point Gauss-Seidel and SOR [24] iterative methods for the linear system (5) can be shown as:

\[
I_{i}^{(n)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j=1}^{i-1} a_{ij} I_j^{(n-1)} - \sum_{j=i+1}^{m} a_{ij} I_j^{(n-1)} \right)
\]

\[
I_{i}^{(n)} = (1 - \omega) I_i^{(n)} + \frac{\omega}{a_{ii}} \left( b_i - \sum_{j=1}^{i-1} a_{ij} I_j^{(n-1)} - \sum_{j=i+1}^{m} a_{ij} I_j^{(n-1)} \right)
\]

(6)

where \( i = 1, 2, 3 \ldots m \) and \( n = 1, 2, 3 \ldots \).

By applying the iterative methods of Gauss-Seidel and SOR for solving equation (4) in the form of algebraic equation (6), the corresponding iteration scheme can be shown as below:

\[
I_{i,j}^{k+1} = \frac{1}{\beta} \left( I_{i,j}^{k} + \delta_U I_{i,j+1}^{k+1} + \delta_D I_{i,j-1}^{k+1} + \delta_L I_{i+1,j}^{k} + \delta_R I_{i-1,j}^{k} \right)
\]

\[
I_{i,j}^{k+1} = \frac{\omega}{\beta} \left( I_{i,j}^{k} + \delta_U I_{i,j+1}^{k+1} + \delta_D I_{i,j-1}^{k+1} + \delta_L I_{i+1,j}^{k} + \delta_R I_{i-1,j}^{k} \right) + (1 - \omega) I_{i,j}^{k}
\]

(7)

for \( k = 1, 2, 3, \ldots n \).

The SOR iterative is a modification of Gauss-Seidel with the present of the weighted parameter, \( \omega \). The weighted parameter can be set in the range of \( 1 < \omega < 2 \) and the ideal estimation of the weighted parameter can be acquired through a few tests that gives minimal number of iterations.

3. Formulation of Two-Point Explicit Group Successive Over-Relaxation Iterative Method

The 2-point Explicit Group iterative has been constructed to solve the linear system of (4) where equation (7) to form block of two node points as following:
$$I_{l_{i,j}}^{k+1} = \frac{1}{\beta} \left( I_{l_{i,j}}^k + \delta U I_{i,j+1}^k + \delta_D I_{l_{i,j+1}}^{k+1} + \delta_L I_{l_{i+1,j}}^k + \delta_R I_{l_{i+1,j-1}}^{k+1} \right)$$

$$I_{l_{i+1,j}}^{k+1} = \frac{1}{\beta_1} \left( I_{l_{i+1,j}}^k + \delta U I_{i+1,j+1}^k + \delta_D I_{l_{i+1,j+1}}^{k+1} + \delta_L I_{l_{i+2,j}}^k + \delta_R I_{l_{i+2,j-1}}^{k+1} \right)$$

The equations above can be formulated into matrix form as:

$$\begin{bmatrix}
\beta & -\delta_L \\
-\delta_R & \beta_1
\end{bmatrix}
\begin{bmatrix}
I_{l_{i,j}}^{k+1} \\
I_{l_{i+1,j}}^{k+1}
\end{bmatrix}
= \begin{bmatrix}
I_{l_{i,j}}^k + \delta U I_{i,j+1}^k + \delta_D I_{l_{i,j+1}}^{k+1} + \delta_R I_{l_{i,j-1}}^{k+1} \\
I_{l_{i+1,j}}^k + \delta U I_{i+1,j+1}^k + \delta_D I_{l_{i+1,j+1}}^{k+1} + \delta_R I_{l_{i+1,j-1}}^{k+1} + \delta_L I_{l_{i+2,j}}^k
\end{bmatrix}$$

Then, the inverse matrix of the coefficient lattice of equation (9) need to be identified in order to form a general scheme of 2-point Explicit Group iterative method. Hence, the general scheme of 2-EG can be shown as below [18]:

$$\begin{bmatrix}
I_{l_{i,j}}^{k+1} \\
I_{l_{i+1,j}}^{k+1}
\end{bmatrix}
= \frac{1}{\beta \beta_1 - \delta_L \delta_R_1}
\begin{bmatrix}
\beta_1 & \delta_L \\
\delta_R_1 & \beta_1
\end{bmatrix}
\begin{bmatrix}
I_{l_{i,j}}^k + \delta U I_{i,j+1}^k + \delta_D I_{l_{i,j+1}}^{k+1} + \delta_R I_{l_{i,j-1}}^{k+1} \\
I_{l_{i+1,j}}^k + \delta U I_{i+1,j+1}^k + \delta_D I_{l_{i+1,j+1}}^{k+1} + \delta_R I_{l_{i+1,j-1}}^{k+1} + \delta_L I_{l_{i+2,j}}^k
\end{bmatrix}$$

The equation (10) can be simplify as following:

$$I_{l_{i,j}}^{k+1} = \frac{1}{\beta \beta_1 - \delta_L \delta_R_1} (\beta_1 S_1 + \delta_L S_2)$$

$$I_{l_{i+1,j}}^{k+1} = \frac{1}{\beta \beta_1 - \delta_L \delta_R_1} (\delta_R_1 S_1 + \beta S_2)$$

where $S_1$ and $S_2$ denoted as:

$$S_1 = I_{l_{i,j}}^k + \delta U I_{i,j+1}^k + \delta D I_{l_{i,j+1}}^{k+1} + \delta_R I_{l_{i,j-1}}^{k+1}$$

$$S_2 = I_{l_{i+1,j}}^k + \delta U I_{i+1,j+1}^k + \delta D I_{l_{i+1,j+1}}^{k+1} + \delta_R I_{l_{i+1,j-1}}^{k+1} + \delta_L I_{l_{i+2,j}}^k$$

It can be seen that the calculations for both points $I_{l_{i,j}}^{k+1}$ and $I_{l_{i+1,j}}^{k+1}$ are totally independent. As shown in Figure 1., the implementation of 2-EG iterative method occurs in each group of two node points meanwhile the ungroup nodes positioned next to the boundary are computed using direct method [19]. To form 2-EGSOR iterative method, the weighted parameter, $\omega$ need to be added into the equation (11). Then, it can be rewritten as:

$$I_{l_{i,j}}^{k+1} = \frac{\omega}{\beta \beta_1 - \delta_L \delta_R_1} (\beta_1 S_1 + \delta_L S_2) + (1 - \omega) I_{l_{i,j}}^k$$

$$I_{l_{i+1,j}}^{k+1} = \frac{\omega}{\beta \beta_1 - \delta_L \delta_R_1} (\delta_R_1 S_1 + \beta S_2) + (1 - \omega) I_{l_{i+1,j}}^k$$

![Figure 1. Illustration of 2-EGSOR iterative method computational network in finite grid.](image-url)
The iteration process of equations (7) and (13) continues until the convergence criterion is satisfied, which in this study we imply the convergence error tolerance when the overall pixel difference between the images produced by using Jacobi iterative methods which is less than 5.0%.

4. Findings

The figure 2 illustrates the three instances of color image with distinct sizes. In this work, three iterative methods are used to each input image. For the performance assessment, the number of iterations and time taken for the images to blur are also recorded. The classical Jacobi iterative method is assigned to be the referral technique for the final blurred image. The iterations for Gauss Seidel, SOR and 2-EGSOR are stopped when the overall pixel distinction of output images to the Jacobi output image that more than 5 is less than 5%. At that point, the threshold value and time-step or value of \( \lambda \) fixed at \( K = 2 \) and 1.0 respectively.

![Figure 2.](image)

(a) (b) (c)

Figure 2. (a), (b) and (c) are the test pictures with sizes of 512x512, 1024x1024 and 2048x2048 respectively [25].

The iterations are running three times for colour Red, Green and Blue independently as the algorithm filtered the colour image based on the RGB channels. This causing each shading recorded different number of iterations \( k \) and computational time \( t \). The iterations \( k \) means for the three channels run for each image are taken and recorded in Figure 3 and Table 1 respectively. For Jacobi method, the number of iterations used for the control parameter of the final image in this experiment are \( k = 500 \), \( k = 1000 \) and \( k = 2000 \) for example (a), (b) and (c) respectively.

In the Figure 3, the result of the iterations number for Gauss-Seidel, SOR and 2-EGSOR methods. It illustrated for both SOR and 2-EGSOR iterative methods required less iterations contrasted with Gauss-Seidel to produce the final images. The reduction percentage for both SOR and 2-EGSOR method to the Gauss-Seidel methods in terms of number of iterations are approximately by 65.47%-78.58% and 65.87%-79.27% respectively. While, the 2-EGSOR method has slightly reduced the number of iterations against SOR method by 1.15%-3.21%.
Other than that, it also can be observed form Table 1 that the computational time, $t$ for SOR and 2-EGSOR have reduced along the number of iterations against Gauss-Seidel methods. Approximately the computational time, $k$ for SOR and 2-EGSOR against Gauss-Seidel have drop by 60.76%-79.93% and 65.07%-81.64% respectively. Meanwhile, the 2-EGSOR method has also slightly reduced the computational time, $k$ against SOR method by 1.10%-8.51%. There is no difference in terms of quality of the final output images produced by all three methods as illustrated in Figure 4 which the pixels difference (error) of more than 5 between image produced by Jacobi method and those three iterative methods are less than 5% as shown in Table 1.

Table 1. The computational time, $t$ (seconds) for image to blur by Gauss-Seidel, SOR and 2-EGSOR iterative methods.

| Example/ Method | Gauss-Seidel | SOR | 2-EGSOR |
|-----------------|--------------|-----|---------|
|                 | $t$ | error | $t$ | error | $t$ | error |
| (a) 512x512     | 20.44 | 0.0496 | 8.02 | 0.0497 | 7.14 | 0.0499 |
| (b) 1024x1024   | 326.90 | 0.0498 | 65.59 | 0.0497 | 60.01 | 0.0493 |
| (c) 2048x2048   | 2738.49 | 0.0499 | 761.94 | 0.0497 | 622.53 | 0.0496 |
5. Conclusion

Three iterative methods of Gauss-Seidel, SOR and 2-EGSOR have been conducted to compute the solutions of non-linear diffusion equation for application in image blurring. The Jacobi has used as a control method with iterations, $k = 500$, $k = 1000$ and $k = 2000$ for example (a), (b) and (c) respectively which based on the size of image. As expected, the results showing the 2-EGSOR iterative method able to surpass the performance of Gauss-Seidel and SOR in producing the same quality of Jacobi image. The reduction percentage of number of iterations and blurring time for 2-EGSOR against Gauss-Seidel iterative method approximately each by more than 60%. As shown in Figures 4, there is no noticeable difference in terms of quality of the final images as the pixels difference of more than 5 produce by Jacobi method is less than 5%. Aside from the 2-EGSOR iterative method which is categorized as a family of two-point block and one parameter iterative method, further examination ought to be made to research the proficiency of the four-point block such as 4-EGSOR [13].
6. References

[1] Perona P and Malik J 1990 Scale-space and edge detection using anisotropic diffusion \textit{IEEE Transactions on Pattern Analysis and Machine Intelligence} \textbf{12}(7) pp 629-639

[2] Weickert J, Romeny B M T H and Viergever M A 1998 Efficient and reliable schemes for nonlinear diffusion filtering \textit{IEEE Transactions on Image Processing} \textbf{7}(3) pp 398-410

[3] Sharma P, Khan K and Ahmad K 2017 Image denoising method based on local neighborhood and modified mean in wavelet packet domain, \textit{Global Sci-Tech} \textbf{9}(3) pp 129-137

[4] Yu H and Deng L J (2016) Image Editing Via Searching Source Image. Applied Mechanics, Mechatronics and Intelligent Systems: pp 373-380

[5] Hong, J H, Saudi A and Sulaiman J 2017 Application of SOR Iteration for Poisson Image Blending \textit{Proceedings of the International Conference on High Performance Computing, Computing and Communications} (Kuala Lumpur) pp 60-64

[6] Hama S M and Al-Ani M S 2013 Medical image enhancement based on an efficient approach for adaptive anisotropic diffusion \textit{International Journal of Advances in Engineering and Technology} \textbf{6}(3) pp 1424-1430

[7] Danachandra N, Manglem K and Chanu Y J 2015 Image segmentation using K-means clustering algorithm and subtractive clustering algorithm \textit{Procedia Computer Science} \textbf{54}(2015) pp 764-771.

[8] Li Y, Liu F, Turner I W and Li T 2018 Time-fractional diffusion equation for signal smoothing, \textit{Applied Mathematics} \textbf{326} pp 108-16

[9] Dollár P and Zitnick C L 2015 Fast edge detection using structured forests \textit{IEEE Transactions on Pattern Analysis and Machine Intelligence} \textbf{37}(8) pp 1558-1570

[10] Catte F, Lions P L, Morel J M and Coll T 1992 Image selective smoothing and edgedetection by nonlinear diffusion \textit{SIAM Journal on Numerical Analysis} \textbf{29}(1) pp 182-193

[11] Wu W and Zhong C 2010 An Improved Nonlinear Diffusion Algorithm for Image Denoising \textit{International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE)} (Changchun) pp 188-191

[12] Eng J H, Saudi A and Sulaiman J 2017 Numerical analysis of the explicit group iterative method for solving Poisson image blending problem \textit{International Journal of Imaging and Robotics} \textbf{17}(4) pp 15-24

[13] Eng J H, Saudi A and Sulaiman J 2017 Numerical evaluation for solving Poisson image blending problem by four-point Egsor iterative method \textit{Far East Journal of Mathematical Sciences} \textbf{102}(12) pp 3127-39

[14] Eng J H, Saudi A and Sulaiman J 2018 Performance analysis of the explicit decoupled group iteration via five-point rotated laplacian operator in solving Poisson image blending problem \textit{Indian Journal of Science and Technology} \textbf{11}(12)

[15] Saudi A, Sulaiman, J and Tanalol S H 2017 Path planning in structured environment using harmonic potential fields via block iterative method \textit{5th International Conference on Future Computational Technologies} (Kyoto) pp 99-106

[16] Chew J V L and Sulaiman J 2017 Application of four-point Newton-EGSOR iteration for the numerical solution of 2D porous medium equations, \textit{Journal of Physics: Conference Series} \textbf{890}

[17] Muthuvalu M S, Aruchunan E, Sulaiman J, Karim S A A and Rashidi M M 2013 The 2-point explicit group successive over-relaxation method for solving fredholm integral equations of the second kind \textit{Applied Numerical Mathematics and Scientific Computation} pp 60-70.

[18] Evans D J 1986 Group explicit iterative methods for solving large linear systems \textit{International Journal of Computer Mathematics} \textbf{17}(1) pp 81-108

[19] Evans D J and Yourif W S 1986 Explicit Group Iterative Methods for solving elliptic partial differential equations in 3-space dimensions. \textit{Int. J. Computer Maths.} \textbf{18} pp 23-340
[20] Chen Y, Yu W and Pock T 2015 On learning optimized reaction diffusion processes for effective image restoration *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (Boston: IEEE) pp 5261-69

[21] Nomura A, Okada K and Mizukami Y 2016 Preprocessing in a reaction-diffusion algorithm designed for image edge detection *IEEE International Symposium on Signal Processing and Information Technology* (Limassol, Cyprus: IEEE) pp 166-71

[22] Liu C, Yan X and Yang B 2017 An adaptive anisotropic thermal diffusion filter for image smoothing *10th Internatioan Congress on Image and Signal Processing, BioMedical Engineering and Informatics* (Shanghai, China: IEEE) pp 1-5

[23] Kamalaveni V, Anitha Rajalakshmi R and Narayanan Kutty K A 2015 Image denoising using variations of perona-malik with different edge stopping functions *Procedia Computer Science* **58** pp 673-8

[24] Young D M 1954 Iterative Methods for solving Partial Difference Equations of Elliptic Type *Trans. Amer. Math. Soc.* **76** pp 92-111

[25] Test examples are available in https://www.pexels.com