Encryption On Grayscale Image For Digital Image Confidentiality Using Shamir Secret Sharing Scheme

Rodiah, Dyah Anggraini, Fitrianingsih, Farizan Kazhimi

Department of Informatics, Gunadarma University, Depok, Indonesia.

{rodiah,d_anggraini,fitrianingsih}@staff.gunadarma.ac.id,farizankazhimi@gmail.com

Abstract The use of high-frequency internet in the process of exchanging information and digital transaction is often accompanied by transmitting digital image in the form of raster images. Secret sharing schemes are multiparty protocols that related to the key establishment which provides protection against any threats of losing cryptography key. The greater the key duplication, the higher the risk of losing the key and vice versa. In this study, Secret Sharing Method was used by employing Shamir Threshold Scheme Algorithm on grayscale digital image with the size of 256x256 pixel obtaining 128x128 pixels of shared image with threshold values (4,8). The result number of shared images were 8 parts and the recovery process can be carried out by at least using 4 shares of the 8 parts. The result of encryption on grayscale image is capable of producing vague shared image (*i.e.*, no perceptible information), therefore a message in the form of digital image can be kept confidential and secure.

1. Introduction

Confidentiality is very crucial aspect in information exchange. It requires a security process to keep the privacy hidden. One of the forms of information that can be maintained to be hidden or confidential is in the form of images. As the use of internet is frequently high in the process of exchanging information and digital transaction, sending digital image in the form of raster images is often carried out. One of the methods used in image encryption is Secret Image Sharing. The Secret Image Sharing Method is one of the types of encryption that used to encrypt information in the form of images. This method divides the image into several sections or subsets. Each section (subset) does not have any perceptible information (*i.e.,* vague) and the original image can only be generated by combining the divided sections (subsets) [1]. Secret Sharing Schemes are multiparty protocols that related to the key establishment which provides protection against the threat of losing cryptography key. The greater the duplication of the key makes the risk of losing the key become higher and vice versa. Secret Sharing Scheme overcomes this problem without increasing the number of risks. It can also be used to distribute trust or shared control of critical activity with the intervention of only t of n users [2].

Previous studies related to Secret Image Sharing, among others, is the implementation of Secret Image Sharing on compressed files using Huffman Algorithm [3]. This Algorithm was analogized as the key to open the Huffman file. Shamir Secret Sharing was sufficiently adequate to be used to share general data. Unfortunately, in terms of size, Huffman Algorithm was less effective as the size of the shared data was equal to or greater than the size of the confidential data. Hence, the required size of the overall shared data was n times greater than the size of the source. Secret sharing method was used normally in encoding. The header bytes of the data were encrypted using the Shamir secret sharing, or representing Huffman tree (graph) as matrix and then secret sharing, specifically for matrix, was conducted or in other words variation of visual cryptography principle in which pixels were replaced by bits was used. In this study, to obtain secured secret sharing, secret sharing which was a modified version of visual cryptography for the header and naive secret sharing was carried out [3].

The study of secret sharing on image is extended by dividing the image into several subsets [4]. Every subset of the image was a subset of the original image. Scheme in this study generating two divider images of the original image (x), black and white image, in which the image of x_1 stands for subset 1 and the image of x_2 intended for subset 2. x_1 and x_2 were random distributions of black and white pixels and did not show any perceptible information. When x_1 and x_2 were layered or stacked, the information would be perceptible. It is the same as the original image. If there was only x_1, then the information of x would not be perceptible without x_2. The results showed that binary image generate a share image, namely shared 1 and shared 2 which did not display the information as in the original image. The white shared image had same combinations, while the black shared image had different combinations. White pixel combinations did not fully display white color. Whereas, the black pixel combinations fully displayed black color because black was the color of the image information [4].

Another study was also conducted using Dynamic Embedding Method to produce better visual quality on stego-image [5]. During the experiment, one of the authentication bits on one of pixel blocks on stego-image was changed to determine whether the authentication process was successful. The pixel block on stego-image was randomly selected and one of the authentication bits on the randomly selected pixel block was changed. At each phase of verification, the stego-image was detected as fraudulent stego-image. This proved that the Authentication-chaining Method had high probability for a stego-image, of which pixel bits had been altered, passing through verification phases by using 2 authentication bits. The value of Peak Signal-to-Noise Ratio of stego-image generated by Dynamic Embedding method was always higher than the stego-image generated by other methods. The method of Authentication-Chaining used to authenticate stego-image was able to have high probability in detecting stego-image by using 2 authentication bits [5].

Secret Sharing method was used by employing Shamir Threshold Scheme algorithm on grayscale digital image of 256x256 pixel size. The algorithm generates a shared image of 128x128 pixels with threshold values (4.8). The resulting number of shared was 8 parts and the recovery process can be carried out by at least using 4 shares of the 8 parts. The result of encryption on grayscale image is capable of producing vague share image (no perceptible information), therefore a message in the form of digital image can be kept confidential.

2. Method

In this study, an image is encrypted into several shared images, then the shared images would be decrypted into the original image. Secret sharing is a method in which the keys of cryptographic results are divided into several subsets without increasing any confidentiality risks. Secret sharing also manages any problems relating to distribution of the keys by only allowing access to t of n users in which $t \leq n$ to perform the initial key establishment. The idea of secret sharing is to divide the secret keys into several subsets, called shares, then distribute them to several parties. Only subsets of those parties can or are allowed to re-establish the initial keys [6]. The method used in this experiment is Secret Sharing by employing Shamir Threshold Scheme algorithm. The workflow process of impleenting Shamir Threshold Scheme algorithm.

2.1. Secret Sharing Algorithm

The algorithm used in this paper is Shamir's Threshold Scheme applied to perform encryption or to divide an image into multiple Share images. Shamir secret sharing scheme is a threshold scheme based on polynomial interpolation. A number of k points in two-dimensional space (x_1, y_1), (x_2, y_2), ..., (x_k, y_k) with different x_i will form exactly just one polynomial equation $q(x)$ with degree of $k - 1$ so that $q(x_i) = y_i$ is applied for all $1 \leq i \leq k$.

From the above statement and without diminishing the interpretation of it generally, it can be assumed that D data are numbers, and will be divided into several parts with a number of n, then a polynomial equation with degree of k - 1 is randomly selected as in equation (1).

$$q_{(X)} = a_0 + a_1X + \ldots + a_{k-1}X^{k-1} \qquad (1)\ [6]$$

In which $a_0 = D$, and for every share which is formed as equation is calculated (2).

$$D_1 = q(1), \ldots, D_i = q(i), \ldots, D_n = q(n), \qquad (2)\ [6]$$

In any subsets of *k* of *Di*, the coefficient of polynomial equation *q*(*x*) can be searched by performing interpolation, and then calculate *D* = *q*(0). However, a number of *k* – 1 of *Di* cannot or is not sufficient to be calculated in order to get *D*. Shamir secret sharing scheme method uses modulo (modular arithmetic) as a substitute for real arithmatic. As an illustration, can be seen in Figure 1. For *D* data, a prime number of *p* which is greater than *D* and *n* is selected. Coefficients of a_1, a_2, a_3, ..., a_{k-1} in the equation of *q*(*x*) is randomly selected from the set of integers in [0, *p*) and values of D_1, D_2, ..., D_n are calculated using modulo *p* [6]. The threshold value (*t*, *n*) that is used in our experiment is 4.8. There are several major variable components which are used:

1. *M* is the original image in which will be kept confidential with the size of *H* x *W* (Height x Weight).
2. *n* is the number of participants or the number of encrypted results (Share).
3. *t* is part of *n* or the so-called threshold value (*t*, *n*) in which $t \leq n$ and $t = 1/2\ n$ [7].
4. S_i or S_0, S_1, S_2, … , S_{t-1} are random integer values for mod *P* which are the coefficient in the polynomial equation.
5. *P* is a prime number of which value is greater than *M*, S_i, *n*.
6. X_i atau X_1, X_2, … , X_n are random values for the participants.

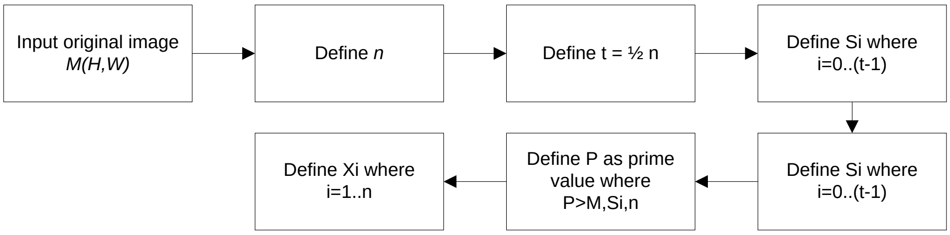


Figure 1. Flowchart shamir threshold scheme

The implementation of Shamir Threshold Scheme in this application is on grayscale images as the secret images or the confidential messages. The gray values in the grayscale images are between 0 and 255, so the variable value of the prime number *P* which is used is the closest prime number of 255 (the maximum value of gray), with *P* = 257 [6].

2.2. Sharing Process on Shamir Threshold

The Sharing process is aimed at dividing the original image into several share images. An original image *M* can be divided into *n* parts for participants by using modulus *P* basis [6].

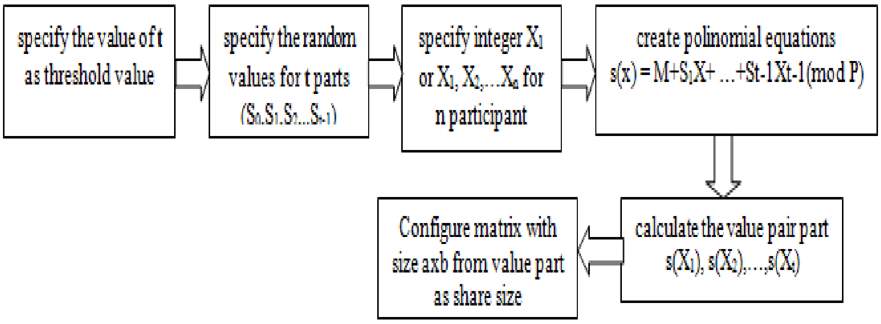

specify the value of t as threshold value
specify the random values for t parts $(S_0, S_1, S_2 \ldots S_{t-1})$
specify integer X_1 or $X_1, X_2, \ldots X_n$ for n participant
create polinomial equations $s(x) = M+S_1X+ \ldots +$St-1Xt-1(mod P)
calculate the value pair part $s(X_1), s(X_2), \ldots, s(X_t)$
Configure matrix with size axb from value part as share size

Figure 2. Scheme of Sharing Process on Shamir Threshold Scheme [6]

2.3. Recovery Process on Shamir Threshold

Recovery process is aimed at performing decryption or re-composing share images within the minimum number of *t* participants into the original image [6]. The steps in decrypting the share images can be seen in Figure 3.

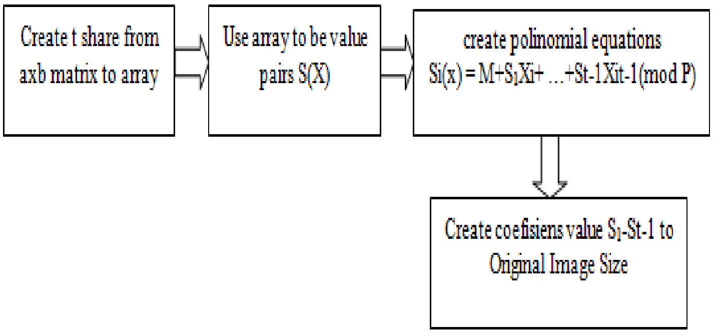

Create t share from axb matrix to array
Use array to be value pairs S(X)
create polinomial equations Si(x) = $M+S_1Xi+ \ldots +$St-1Xit-1(mod P)
Create coefisiens value S_1-St-1 to Original Image Size

Figure 3. Scheme of recovery process on Shamir Threshold Scheme [6]

2.4. Security Analysis on Shamir Threshold

In restoring an image from *t* share, component of S_i coefficients from polynomial of $s(x) = M + S_1X + \ldots + S_{t-1}X^{t-1}$ (mod *P*) is required. Each polynomial must have at least *t* number of the coefficients. That is if it has coefficients which are less than *t*, then the composition cannot be known precisely. The resulting probability to estimate the values to be coefficiently correct is 1/256. Consequently, for that reason, it is quite impossible to restore the image with combination of $t - 1$ or the number of shares which is less than the *t* value [6].

3. Results and Discussion

In this study, the implementation of secret image sharing is specifically aimed at encrypting grayscale images with size of 256x256 pixels with threshold value of 4.8. The experiments were conducted on 24 types of grayscale images with 3 different sizes. The format of the images that being used were .tif or .bmp file. The reason for conducting the experiments using those data are due to indicate differences on processing times as well as the results. The followings are 2 of the 24 samples:

1. Image 1. Grayscale image size 256x256 pixels with .bmp format. File name 'clock_gray_256.bmp'
2. Image 2. Grayscale image size 512x512 pixels with .tif format. File name 'cameraman.tif'.

3.1. Image Experiments

Experiment on Image 1

The image on experiment on image 1 was a grayscale image with the size of 256x256 pixels. The format of the image is .bmp format namely 'clock_gray_256.bmp'. The experiment was conducted by using the following variables:

1. M = 256x256
2. $t = 4$
3. $n = 8$
4. Threshold values for participant $x_1 = 1$, $x_2 = 3$, $x_3 = 5$, $x_4 = 8$.
5. $P = 257$

From above variables, the following steps were taken:

a. Confidentiality on M was changed into integer value M = 256 x 256 = 65536

b. Insert components on polynomial equation as follows:

- $S(x) = M + S_1x^t + S_2x^t + \ldots + S_{t-1}x^{t-1}$ (mod 257)
- $S(1) = 65536 + S_1(1) + S_2(1)^2 + S_3(1)^3$ (mod 257)
- $S(3) = 65536 + S_1(3) + S_2(3)^2 + S_3(3)^3$ (mod 257)
- $S(5) = 65536 + S_1(5) + S_2(5)^2 + S_3(5)^3$ (mod 257)
- $S(8) = 65536 + S_1(8) + S_2(8)^2 + S_3(8)^3$ (mod 257)

Process of dividing image 1 into subsets can be seen in Figure 4.

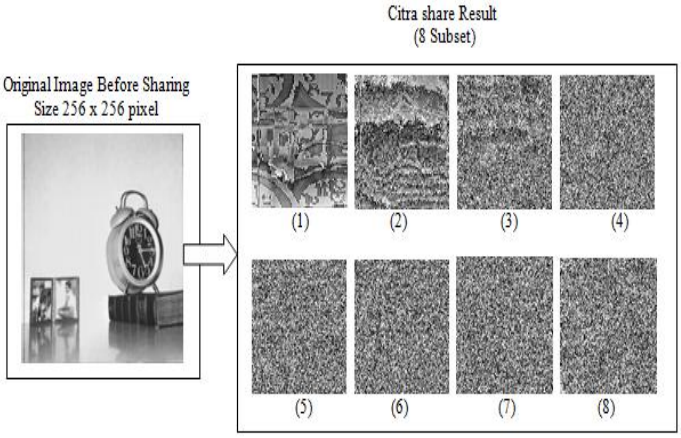


Figure 4. Original image *clock_gray_256.bmp* into 8 Shared Images

Experiment on Image 2

Experiment on image 2, the image used was grayscale image size 512x512 pixels with .bmp format under the file name of 'cameraman.tif'. The experiment was conducted by using the following variable components:

1. M = 512x512
2. $t = 4$
3. $n = 8$
4. Threshold values for participant $x_1 = 1$, $x_2 = 3$, $x_3 = 5$, $x_4 = 8$.
5. Modulus basis = 521

From above variables, the following steps were taken:

a. Confidentiality on M was changed into integer value M = 512x512 = 262144

b. Insert components on polynomial equation as follows:

- $S(x) = M + S_1x^t + S_2x^t + \ldots + S_{t-1}x^{t-1}$ (mod 521)
- $S(1) = 262144 + S_1(1) + S_2(1)^{2} + S_3(1)^{3}$ (mod 521)
- $S(3) = 262144 + S_1(3) + S_2(3)^{2} + S_3(3)^{3}$ (mod 521)
- $S(5) = 262144 + S_1(5) + S_2(5)^{2} + S_3(5)^{3}$ (mod 521)
- $S(8) = 262144 + S_1(8) + S_2(8)^{2} + S_3(8)^{3}$ (mod 521)

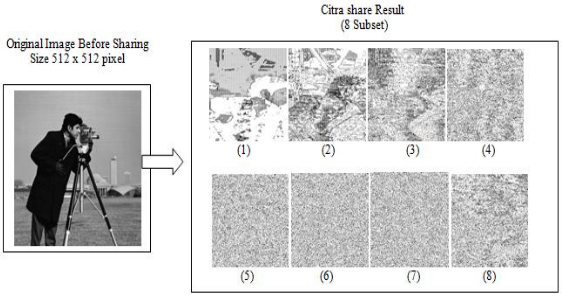


Figure 5. Original Image *'cameraman.tif'* into 8 Shared Images

3.2. Workflows of the Application

3.2.1. Encryption

Figure 6 detail the design flow secret image sharing for encryption process.

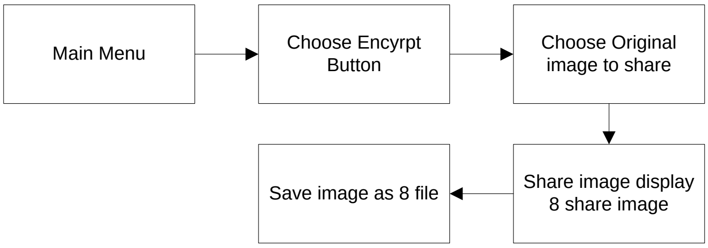

Main Menu
Choose Encyrpt Button
Choose Original image to share
Share image display 8 share image
Save image as 8 file

Figure 6. Encryption Process in Application

From the main menu, select the button with the scrambled rubric image (left button). If the button is clicked, the window will display the image encryption/image sharing.

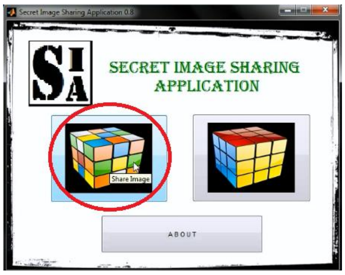
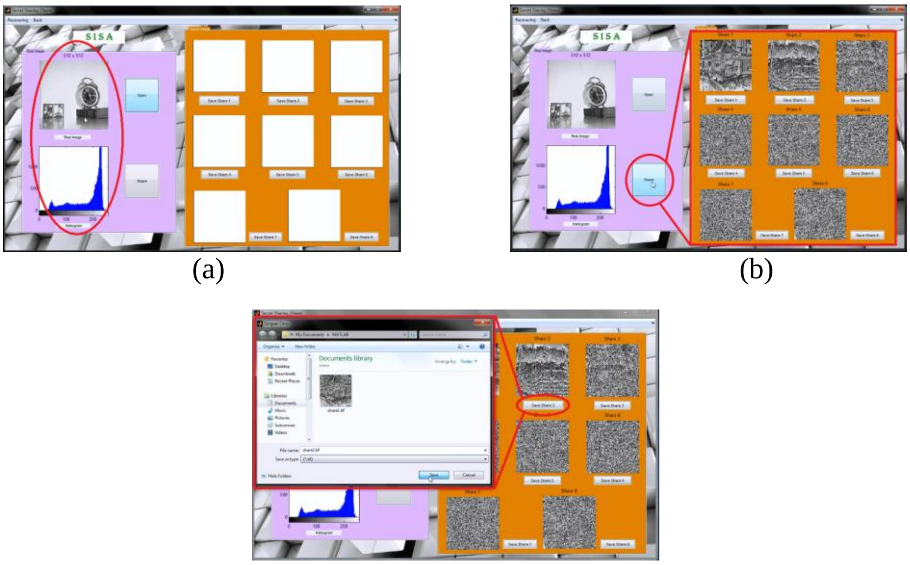

Secret Image Sharing Application 0.8
SI A
SECRET IMAGE SHARING APPLICATION
Share Image
ABOUT

Figure 7. Display of Secret Image Sharing Application Menu

From the window of image encryption/image sharing, click the "Open" button to open the image to be encrypted, as can be seen in Figure 8 (a). The next step is to click the "Share" button to encrypt the opened image into 8 share images as can be seen in Figure 8 (b). To save any of the encrypted share images, click the "Save Share 1", "Save Share 2", ..., "Save Share 8", as needed can be seen in Figure 8 (c).



(a) (b)

(c)

Figure 8(a). Display after successfully opening image. (b) Result of Encryption into 8 Shared Images. (c). Display of Saving the encrypted shared images

3.2.2. Process of Decryption

Figure 9 detail the design flow secret image sharing for decryption process.

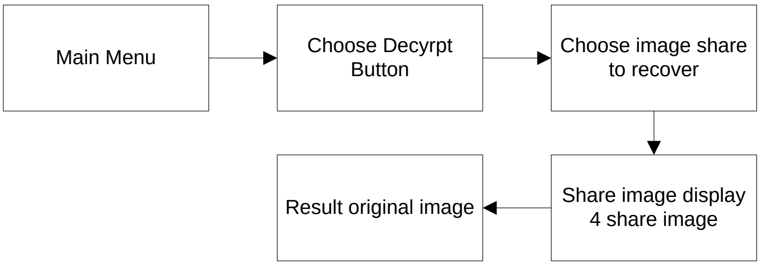
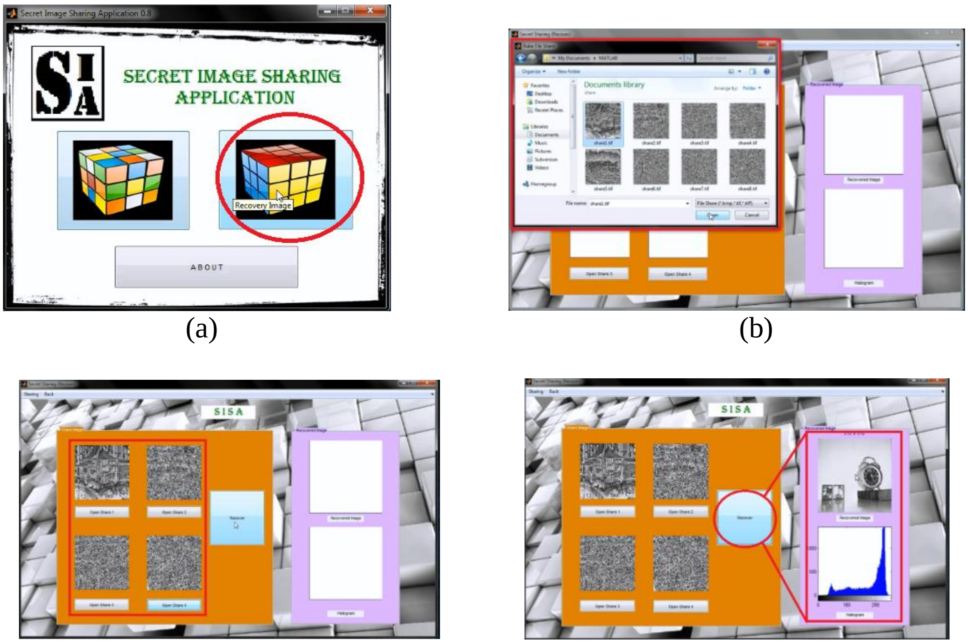

Main Menu
Choose Decyrpt Button
Choose image share to recover
Share image display 4 share image
Result original image

Figure 9. Decryption Process in Application

For decryption process, from the main menu, select the button with a perfectly composed rubric image (right button). If the button is clicked the window will display of image decryption/image recovery can be seen in Figure 10(a). From Figure 10(b) the button of "Open Share 1", "Open Share 2", "Open Share 3", and "Open Share 4", to open 4 of 8 share images/encrypted images are clicked from image decryption/image recovery window. Figure 10(c) shows "Recover" button to perform decryption of the opened shared images into the original image which will be displayed on the right column. Wait until the decryption process is complete. Once it is complete, then the original image will be displayed as can be seen in Figure 10(d).



(a) (b)

(c) (d)

Figure 10. (a). Select Right Button for Decryption. (b). Display of Open Shared Figure. (c). Display of the Opened Shared Images. (d) Display of Result of Image Decryption/Image Recovery

3.2.3. Table of Experiment Results

The experiments were conducted on 24 image samples with different formats file. The results are displayed in the following table.

Table 1. Results of Experiments

No	Image Name	Format	Size (Pixel)	Share Result	Recovery Time
1	Cameraman	.bmp	256 x 256	8	2,5 minute
2	Cameraman	.tif	256 x 256	8	2,5 minute
3	Cameraman	.bmp	512 x 512	8	11 minute
4	Cameraman	.tif	512 x 512	8	11 minute
5	Cameraman	.bmp	1024 x 1024	8	43 minute
6	Cameraman	.tif	1024 x 1024	8	43 minute
7	Clock	.bmp	256 x 256	8	2,5 minute
8	Clock	.tif	256 x 256	8	2,5 minute
9	Clock	.bmp	512 x 512	8	11 minute
10	Clock	.tif	512 x 512	8	11 minute
11	Clock	.bmp	1024 x 1024	8	43 minute
12	Clock	.tif	1024 x 1024	8	43 minute
13	Lena	.bmp	256 x 256	8	2,5 minute
14	Lena	.tif	256 x 256	8	2,5 minute
15	Lena	.bmp	512 x 512	8	11 minute
16	Lena	.tif	512 x 512	8	11 minute
17	Lena	.bmp	1024 x 1024	8	43 minute
18	Lena	.tif	1024 x 1024	8	43 minute
19	Plane	.bmp	256 x 256	8	2,5 minute
20	Plane	.tif	256 x 256	8	2,5 minute
21	Plane	.bmp	512 x 512	8	11 minute
22	Plane	.tif	512 x 512	8	11 minute
23	Plane	.bmp	1024 x 1024	8	43 minute
24	Plane	.tif	1024 x 1024	8	43 minute

Recovery time was influenced by the size of the image, as can be seen in Table 1. From the results of the implementation, the average recovery time for image with 256x256 pixels was 2.5 minutes. Whereas, for image with the size of 512x512 pixels, the average recovery time required 11 minutes and for image with 1024x1024 pixels, it required 43 minutes. It took longer time to recover an image of which pixel size is bigger because the matrix size is greater to be processed.

4. Conclusion and Suggestion

Based on the results of experiments using images 1 and images 2 with the size of 256x256, 1024x1024 and 512x512 pixels, the number of shared images was 8. The threshold that used in this experiment was 4.8 by assuming that 4 is ½ of the threshold value generated by each image and 8 is the number of shared images of the original image. In this study, encryption on grayscale images by implementing Shamir Threshold Scheme algorithm was successful in producing vague share images (no perceptible information) so confidentiality of an information within digital image can be maintained. The experiments as can be seen in Table 1 specify the file size affect the recover time. Shamir Secret Sharing Method implemented in this study is able to anticipate the loss of a key when decrypted process. The trial results showed 4 images combined share of total 8 images share the results of the encryption process is able to restore the original image.

In order to develop the application, it is recommended to implement the experiment on other types of images such as stegano images which have additional confidential information inside the images or on color images that have a large range of colors.

References

[1] L. Bai. A reliable (k,n) image secret sharing scheme. In DASC. pages 31–36. 2006

[2] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone. Handbooks of Applied Cryptography. CRC Press. ISBN: 0-8493-8523-7. 816 Pages. 1996

[3] C. Huang and C. Li. Secret Image Sharing Using Multiwavelet Transform. Journal of Information Science and Engineering, 733-748. 2011.

[4] Jagdeep Verma, dan Vineeta Khemchandani. A Visual Cryptographic Technique to Secure Image Shares, International Journal of Engineering Research and Applications (IJERA), ISSN : 2248-9622, Vol. 2, Issue 1, pp.1121-1125. 2012

[5] Widyadhana, Arya, dan Muchmamad Husni. Penerapan Secret Image Sharing Menggunakan Steganografi dengan Metode Dynamic Embedding dan Authentication-Chaining, Jurnal Teknik ITS, Vol.1. ISSN: 2301 – 9271. 2012

[6]. Wang, Shuang. Distributed Storage scheme Based on Secret Sharing Schemes, University of Oklahoma, Tulsa, OK, USA. 2012

[7] Gonzalez and Woods. Digital Image Procssing. Second Edition. Prentice Hall. 2002