Arnold Transformed Position Power First Mapping (AT-PPFM) for Secret Digital Image

1st Wilis Kaswidjanti
Informatic Departement
UPN "Veteran" Yogyakarta
Yogyakarta, Indonesia
wilisk@upnyk.ac.id

2nd Hidayatulah Himawan
Informatic Departement
UPN "Veteran" Yogyakarta
Yogyakarta, Indonesia
if.iwan@gmail.com

3rd Afra Oryza Mursita Dewi
Informatic Departement
UPN "Veteran" Yogyakarta
Yogyakarta, Indonesia
afraoryza05@gmail.com

4th Mangaras Yanu Florestiyanto
Informatic Departement
UPN "Veteran" Yogyakarta
Yogyakarta, Indonesia
mangaras.yanu@upnyk.ac.id

5th Rifki Indra Perwira
Informatic Departement
UPN "Veteran" Yogyakarta
Yogyakarta, Indonesia
rifki@upnyk.ac.id

***Abstract*— Digital image data stored and exchanged in cloud storage can be secured using cryptographic and steganographic techniques. The information contained in a data is secured in order to avoid taking data or information from unauthorized parties. The encryption process is done by changing the original image data from what can be understood to be incomprehensible. Encryption uses Arnold Transformation Algorithm which randomizes the pixel of image data by using 4-bit Most Significant hidden images to then be inserted into the encrypted 4-bit Least Significant Cover. Peak Signal to Noise (PSNR) is used to compare image quality before and after extraction. This comparison is based on the test results of the mean error value or Mean Square Error (MSE) of the original image data and the resulting image insertion data. PSNR produced in this study is above the minimum standard value (40 dB), which is between 45.60 dB - 46.10 dB, and the resulting distortion value is very small (MSE> 2). By using a 4-bit insertion of the existing image data, the extraction results are not much different from the hidden image before insertion and the results can be identified. So that the use of Arnold Transform and Position Power First Mapping (PPFM) algorithms reduces distortion and differences as well as the possibility of data leakage from the resulting image data.**

Keywords*— *Arnold Transform, PPFM, Cryptography, Steganography

I. INTRODUCTION

With the Internet, information security including information in the form of images in the process of storing and exchanging images has become an important and worrying problem because it is very vulnerable to many threats and attacks [1][2]. The development of the internet has a significant impact on the progress of information exchange and data storage, one of which is known as cloud computing technology, which is a model that allows the use of computing resources used together [3]. The cloud is a new paradigm in data collection, archiving, analysis, and data sharing based, and provides access to flexible resources [4]. With the various benefits of cloud technology, this encourages organizations and individual users to use and switch their applications and services to this technology [5].

In the use of cloud storage, it does not rule out the possibility that the data exchanged and stored therein may be data that is confidential and should only be known by certain parties. Data security in cloud computing is not guaranteed directly, because the data is placed on a cloud that can be accessed by everyone. In cloud computing, users need to send personal data to the cloud, but don't trust the cloud, so the data will be encrypted and forwarded to the cloud. Concealment of reversible data allows service providers to embed additional messages, such as image labels, notation or authentication information, into encrypted images, and has a reversibility feature to extract additional messages and restore the original image [6].

Security systems that can be used to protect data are cryptographic and steganographic techniques [7]. Cryptographic and steganographic techniques convert the original image into an unreadable image and insert it into a new image that has a different meaning from the original image [8]. Steganography is a method for hiding data by inserting it into multimedia such as pictures, videos and audio [7][9][10]. In steganographic images many techniques are used to hide from seeing confidential information into cover images such as spatial domain methods, changing domain techniques, distortion techniques and mask and filtering techniques [11]. As for the implementation, this research will use the Arnold Transform algorithm and Position Power First Mapping (PPFM).

Arnold Transform's algorithm and Position Power First Mapping (PPFM) continue to develop along with the use of cloud computing technology. This method was invented by a Russian mathematician named Vladimir Igorevich Arnold [12]. According to Wu [6], the Arnold Transform algorithm approach provides a high level of image data encryption by randomizing the host image and producing random pixels that cannot be recognized for authenticity. The proposed algorithm is quite effective and efficient in disguising image data information. Arnold's transformation algorithm based on Cryptography will be against Noise, Sharpening and Contrast Attacks [8]. Arnold Transform algorithm will be used to encrypt image data by randomizing pixels according to the transformation period of the iteration based on the size of the cover image, which will eventually lead to the random form of the cover image [13].

According to Mukhrejee [14], Position Power First Mapping (PPFM) has unique significance in the domain of image steganography, and has a high embedding capacity (bit planting) without a significant distortion range in hiding messages (other images) in it. This shows that this method fulfills all obligations in securing confidential information from hidden image data. The embedding mechanism of

978-1-7281-2380-6/19/$31.00 ©2019 IEEE

Position Power First Mapping (PPFM) is next effectuated [15]. Position Power First Mapping (PPFM) will focus on planting the original image bit into the encrypted cover image in a stable manner so as to minimize the visual difference between the cover image and the final image. In Mukhrejee's study [14], the number of bits implanted was 2 bits.

The combination of the Arnold Transform algorithm and the position power first mapping method by planting 2-bit secret images produces good stego-image quality with high security without any significant quality changes to stego-image, but this research does not produce good quality stego image extraction results. because it cannot maintain the essence of information from hidden images or the extraction results themselves.

This research is based on the importance of image data security strength, smooth camouflage of the stego-image encoded process that will be stored and exchanged into Google Drive. In addition, this research will also focus on maintaining the quality of stego-image extraction results, so that the essence of information information from hidden images can be maintained by embedding 4-bit hidden images into the host image.

From the explanation above, this research will focus on solving digital image security problems that are integrated with cloud storage services by applying Arnold Transform algorithm and Position Power First Mapping (PPFM). The implementation of the Arnold Transform algorithm and Position Power First Mapping (PPFM) is expected to improve security in the exchange and storage of image data in cloud storage to avoid data leakage or data leakage problems and information can not be known by other parties who are not interested.

II. METHODOLOGY

In this research, the encryption consists of encoding and decoding processes. Where the two processes have a connection in the integration of image data to be hidden.

The first process is the encoding process. This encoding process begins by inputting image data in the form of hidden data or data to be hidden and also a cover image which is a cover cover image that will be inserted by a hidden image or secret image. The next stage is the preprocessing stage in which the inserted image will receive the preprocessing stage before entering the encryption and insertion process. Then the cover image will go through a randomization process based on permutation by implementing the Arnold Transform algorithm. After that, the hidden image will enter the insertion stage into the encrypted cover image. After experiencing the insertion, the image from the insertion will go through the stage of returning the pixel position as before with the implementation of the Arnold Transform inverse algorithm. Then the stego-image from the previous stage will be displayed in a preview of the encoding results, then the stego-image can be stored in the local storage of the user or online storage service.

The second process is the decoding process of stego-image. This decoding process begins by inputting image data that has previously been through the encoding process, both those that have been stored in the user's device or in an online storage service. The next stage is the stage of randomization of pixels as is done in the encoding process using the Arnold Transform inverse algorithm. After going through the cover scrambling stage the next step is extraction of hidden images that have been inserted using the anti PPFM method, the extraction results can then be saved into the existing device.

Both of these processes can be seen in Figure 1 which shows the flow of the research process carried out. The encoding and decoding process of the existing stego-image is integrated into one image data that is hidden in the running process.

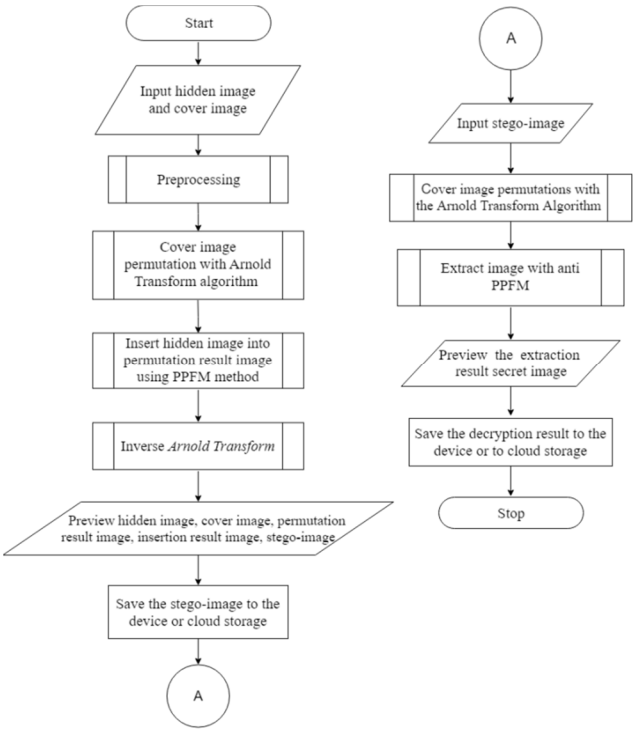

Fig. 1. Encoding and Decoding Flowchart

This preprocessing sub-process involves changing the file extension, first of all this stage begins with reading the cover image file and the hidden image that has been inputted (Figure 2).

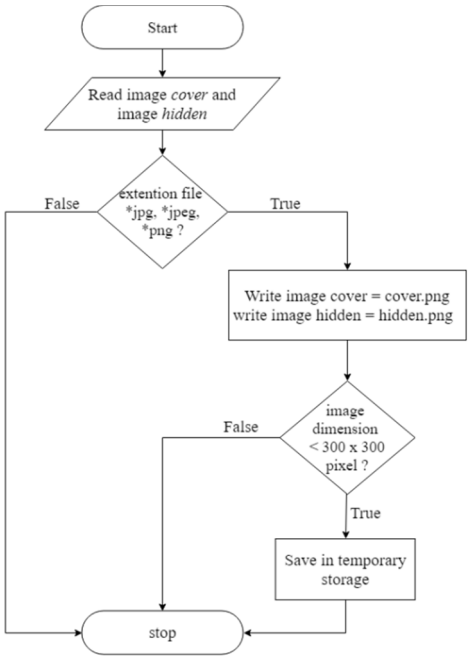

Fig. 2. Preprocessing Flowchart

The cover image or Cover.png image will experience randomization according to permutation using the Arnold Transform algorithm (figure 3).

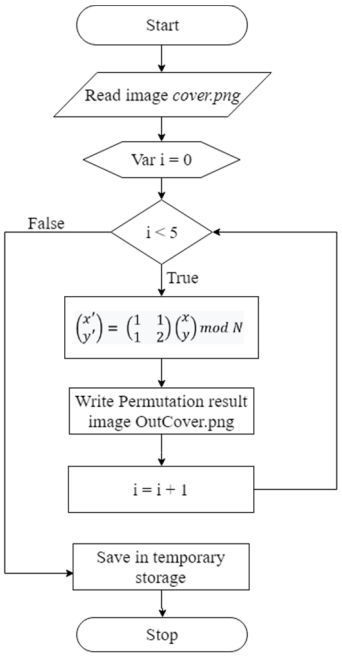

Fig. 3. Arnold Transform Permutation Flowchart [14]

Sub insertion process with PPFM is the process of inserting hidden images into randomized cover images (figure 4).

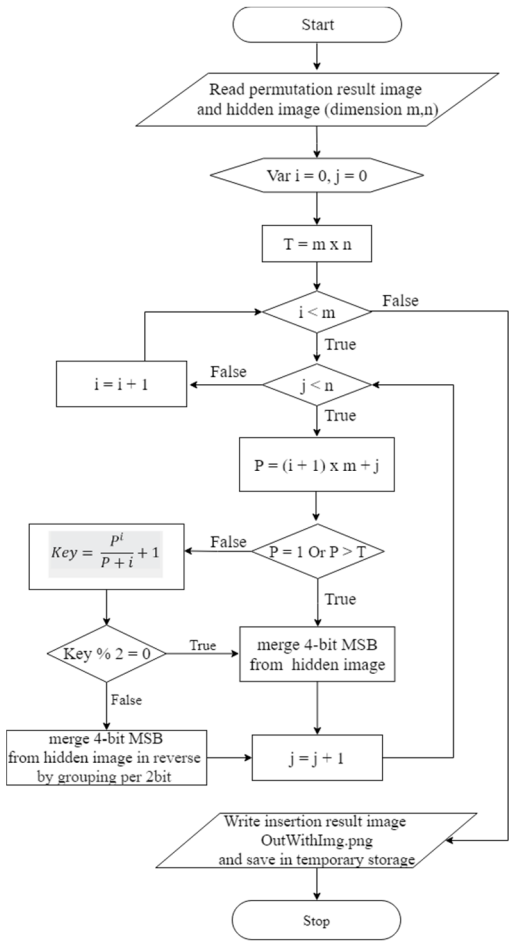

Fig. 4. Inserting Proses with PPFM Flowchart

Stages of returning random pixel positions into regular pixel positions and have meaning or information by permutation using the Arnold Transform inverse algorithm (figure 5).

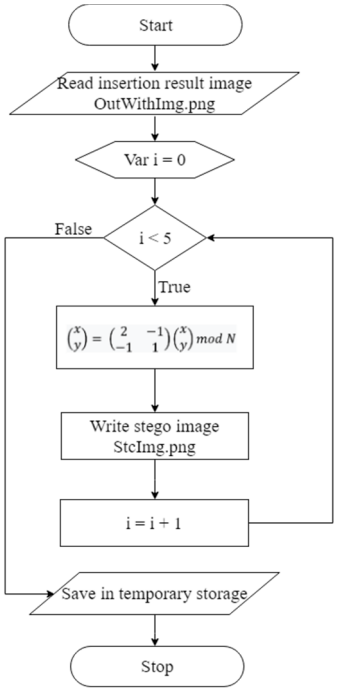

Fig. 5. Inverse Arnold Transform Flowchart [14]

The hidden image extraction sub-process that has been inserted using the anti PPFM method is the process of returning hidden images from the cover image (figure 6).

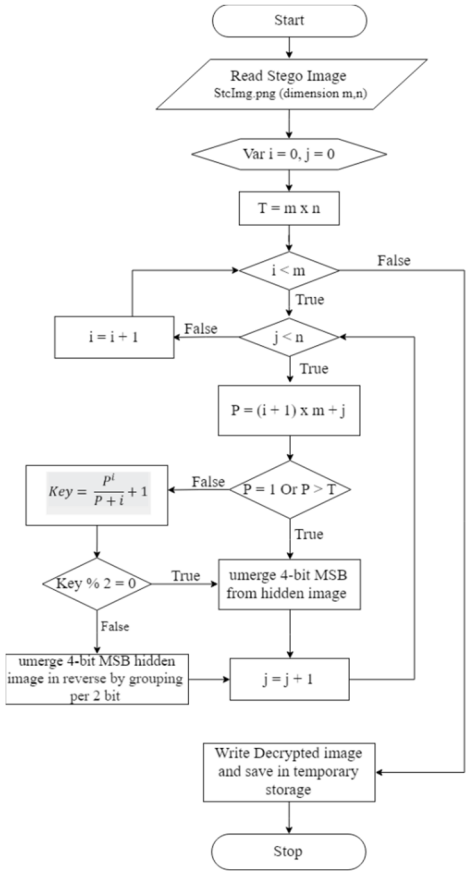

Fig. 6. Extract Image with anti PPFM

III. Results And Discussion

In this study, the extracted confidential data is used to measure the similarity between the original image and the stego image. The purpose of the experimental results is to evaluate the level of distortion of the stego image. This study needs to evaluate how well the proposed method compared to previous studies [14]. The difference in previous research by Mukhrejee [14], is in the number of bits inserted in the cover image. In the previous research, 2-bit most significant was inserted and then inserted into the 2-bit least significant cover that has been encrypted, in this study used 4-bit most significant hidden images and then inserted into the 4-bit least significant encrypted cover. In this study, we used Peak Signal to Noise (PSNR) to compare the quality of the cover image after the secret message was inserted and the hidden image extracted [16]. The MSE value must be determined before calculating the PSNR. MSE is the average error value between the cover image and the insertion image. MSE and PSNR calculations are presented in (1) and (2).

A. Testing of Mean Square Error (MSE) and Peak Signal-to-Noise Ratio (PSNR)

Table I shows the results of the MSE calculation of each image, the calculation is obtained through the average square between the original image and the noise image that has undergone insertion using equation (1). Table 1 also shows the results of the PSNR calculation. PSNR is an example of parameters commonly used as indicators to measure the similarity of two images. These parameters are often used to compare the results of image processing with the initial image or original image. To calculate the PSNR the MSE value is required. PSNR calculation is obtained through calculations using equation (2).

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}(S_{xy} - C_{xy})^2 \quad (1)$$

$$PSNR = 10\log_{10}\left(\frac{C_{max}^2}{MSE}\right) \quad (2)$$

Here, x and y are the coordinates of the pixel value of the stego and original images respectively, N and M are the dimensions or size of the image, S_{xy} states stego image and C_{xy} represent the original image. Then the bits per pixel (bpp) are used to calculate the amount of capacity. Bpp value is obtained by dividing the amount of secret data to be inserted and the number of pixels in the original image. In this research, the results of our experiment are presented in Table I.

In table I, the name of the image is the original name of each image that has gone through the testing process. Then the size of the image is a dimension of the test image, in this case each image is carried out three times the test with different dimensions of the image. Testing is done by comparing previous studies. Lastly, the value is the result of MSE and PNSR calculations.

TABLE I. Table of MSE and PSNR Testing Of stego-image

Image Name	Image Dimension	Mukhrejee et al.[14]		Proposed	
		MSE	PSNR	MSE	PSNR
Green Lake	200x200	1,93	45,31	1,63	46,04
	225x225	1,94	45,30	1,61	46,10
	250x250	1,93	45,32	1,62	46,06
Night City	200x200	2,38	44,39	1,80	45,61
	225x225	2,36	44,43	1,81	45,59
	250x250	2,38	44,40	1,80	45,60

PSNR is often expressed on a logarithmic scale in decibels (dB). PSNR values falling below 30 dB indicate relatively low quality, where distortion caused by insertion is clearly visible. However, the high stego-image quality is at 40dB and above. So from table I it can be concluded that the encryption with the AT algorithm and the insertion with PPFM have a small distortion range that can be seen from the results of the test that produces a good PSNR value for stego images.

TABLE II. Table of MSE and PSNR Testing Of extracted image

Image Name	Image Dimension	Mukhrejee et al. (2-bit) [14]		Proposed (4-bit)	
		MSE	PSNR	MSE	PSNR
The President	200x200	234,02	24,47	25,42	34,11
	225x225	233,07	24,49	25,14	34,16
	250x250	234,48	24,46	25,32	34,13
Girl	200x200	322,64	23,08	23,91	34,38
	225x225	324,15	23,06	24,01	34,36
	250x250	322,56	23,09	23,84	34,39

Based on table I, it can be seen that the insertion of 4 bits in each hidden image pixel into the cover image does not cause significant changes to the cover image. This can be proven based on the PSNR value of stego-image with 4-bit image insertion not having a wide range of differences to the PSNR results of stego-image with 2-bit image insertion having PSNR values above 40 which indicates the quality of stego-image high. Whereas based on table II, the results of extraction of stego-images with 4-bit image insertion have a PSNR value above 30 which indicates that the quality is still good, not low. The extraction result is not much different from the hidden image before insertion and the results can be identified. While the results of image extraction on stego-images with 2-bit insertion have results that are far different from PSNR values below 30 and the results cannot be identified. From the experiment, it can be seen that the PSNR is higher than that of the previous method [12].

IV. Conclusion

The use of Arnold transform algorithm method and Position power first mapping in the process of extracting 4-bit Most Significant image data, resulting a data randomization process with Peak Signal to Noise Ratio (PNSR) of 45.60 dB - 46.10 dB for the generated stego image and the extraction results show that the image quality remains good with PNSR above 30 dB. This analysis technique helps users to hide image data to be stored online by reducing the possibility of data distortion ranges that result in changes of image data significantly. Implementation of this algorithm method results in increased security on the process of exchanging and storing image data in cloud storage.

Acknowledgment

This research was supported by UPN "Veteran" Yogyakarta. We are thankful to our colleagues who provided expertise that greatly assisted the research. We are also grateful to who moderated this paper and in that line improved the manuscript significantly.

References

[1] S. Mukherjee and G. Sanyal, *Distributed Computing and Internet Technology*, vol. 3347. Springer International Publishing, 2005.

[2] A. Joshi and M. Kumari, "Encryption of RGB image using Arnold transform and involutory matrices," *nternational J. Adv. Res. Comput. Commun. Eng.*, vol. 4, no. 9, 2015.

[3] J. L. Brenier, "An Analysis of the Cloud Computing Security Problem Mohamed," *Int. Surg.*, vol. 47, no. 3, pp. 288–290, 1967.

[4] Sarma Nuthalapati, *Power System Grid Operation Using Synchrophasor Technology*. Springer International Publishing, 2019.

[5] G. Ramachandra, M. Iftikhar, and F. A. Khan, "A Comprehensive Survey on Security in Cloud Computing," *Procedia Comput. Sci.*, vol. 110, no. 2012, pp. 465–472, 2017.

[6] F. Di, M. Zhang, Y. Zhang, and J. Liu, "Reversible Data Hiding for Encrypted Image Based on Interpolation Error Expansion," *Int. J. Mob. Comput. Multimed. Commun.*, vol. 9, no. 4, pp. 76–96, 2018.

[7] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Processing*, vol. 90, no. 3, pp. 727–752, 2010.

[8] S. A. Al-taweel, M. H. Al-hada, and A. M. Nasser, "Image in image Steganography Technique based on Arnold Transform and LSB Algorithms," *Int. J. Comput. Appl.*, vol. 181, no. 10, pp. 32–39, 2018.

[9] Y. Xue, W. Liu, W. Lu, Y. Yeung, X. Liu, and H. Liu, "Efficient halftone image steganography based on dispersion degree optimization," *J. Real-Time Image Process.*, vol. 0, no. 0, p. 0, 2018.

[10] J. V. C. I. R, J. Zhang, W. Lu, X. Yin, W. Liu, and Y. Yeung, "Binary image steganography based on joint distortion measurement q," *J. Vis. Commun. Image Represent.*, vol. 58, pp. 600–605, 2019.

[11] S. Bukhari, M. S. Arif, M. R. Anjum, and S. Dilbar, "Enhancing security of images by Steganography and Cryptography techniques," *2016 6th Int. Conf. Innov. Comput. Technol. INTECH 2016*, pp. 531–534, 2017.

[12] M. R. PourArian and A. Hanani, "Blind steganography in color images by double wavelet transform and improved arnold transform," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 3, no. 3, pp. 586–600, 2016.

[13] E. J. Kusuma, C. A. Sari, E. H. Rachmawanto, and D. R. I. M. Setiadi, "A Combination of Inverted LSB, RSA, and Arnold Transformation to get Secure and Imperceptible Image Steganography," *J. ICT Res. Appl.*, vol. 12, no. 2, p. 103, 2018.

[14] S. Mukherjee, "A Novel Image Steganographic Technique Using Position Power First Mapping (PPFM)," pp. 406–410.

[15] S. Mukherjee and G. Sanyal, "Enhanced Position Power First Mapping (PPFM) based image steganography," *Int. J. Comput. Appl.*, vol. 39, no. 2, pp. 59–68, 2017.

[16] A. Kumar and S. Gandharba, "A Novel n-Rightmost Bit Replacement Image Steganography Technique," *3D Res.*, vol. 0123456789, 2019.