



Implementation of Rubik's Cube Algorithm and Rivest-Shamir-Adleman (RSA) Algorithm on Iris Digital Image Security

Muhammad Khudzaifah¹, Siti Habibatul Ma'rifah²(✉), and Hisyam Fahmi¹

¹ Universitas Islam Negeri Maulana Malik Ibrahim Malang, Malang, Indonesia

² Brawijaya University, Malang, Indonesia

sitihabibatulm@gmail.com

Abstract. Technological developments, especially smartphones, have led to an authentication process that must be developed. One of them is the use of the iris in the authentication process. The use of the iris can increase security in the authentication process because the structure of the iris is unique and differs from individual to individual. Authentication requires encryption and decryption processes to secure data. This research uses Rubik's Cube algorithm and the Rivest-Shamir-Adleman (RSA) algorithm. This study aimed to obtain the accuracy and time efficiency results used in the encryption and decryption process. In this study, the encryption process was carried out using Rubik's Cube algorithm, followed by encryption using the RSA algorithm. In the decryption process, it was obtained using the RSA algorithm, then continued with the decryption using the Rubik's Cube algorithm. The experimental results indicate that the encrypted image results differ from the initial image. With the results, the average Structural Similarity Index Metrics (SSIM) is 0,01, and the average Mean Square Error (MSE) is 37620,59. Furthermore, evaluations of the encryption and decryption process time were also carried out using the RSA public key (197,403) and RSA private key (53,403). The maximum iteration of Rubik's Cube algorithm was 1. In the encryption process, the average time was 0,796 s, and the decryption process obtained an average time of 0,652 s. This study has brought new procedures for securing digital images that can be developed in further studies.

Keywords: Decryption · Digital Image · Encryption · Eye's Iris · Rivest-Shamir-Adleman (RSA) Algorithm · Rubik's Cube Algorithm

1 Introduction

Technological advances that continue to develop significantly impact all aspects of life. Smartphones are one clear proof of the rapid development of technology at this time. There are various kinds of innovations in smartphone technology that further increase its functionality and efficiency. Smartphones are also hardware devices designed to be compact to prioritize user flexibility and sophisticated computer systems. One of the essential features of smartphones is the authentication system. Data storage authentication and smartphone screen lock authentication are often found using patterns and

passwords in the form of text or numbers that are considered ancient or less effective. It also has a low level of security because it is easy for others to know [1].

In connection with the COVID-19 pandemic, which has resulted in all community activities being advised to wear masks. To deal with these problems, one form of development on smartphones that is currently developing is an authentication system with biometrics. With the biometric authentication system, data security and flexibility in processing smartphone transactions can be maintained. It was stated that using passwords in computer system security could be easily hacked, so computer system security is increased by using biometric identification [2]. Several biometric authentications that can be used on smartphones are authentication with fingerprints and irises. Biometric authentication can provide a higher level of security because it is unique and has a low error rate [3] as we know that fingerprints and irises are individual parts of the human body that differ from person to person. Considering the current public health conditions that require the use of masks at all times, it is necessary to innovate the smartphone authentication process with a biometric method that can be done without removing the cover. One solution is to use the iris. This is because fingerprints can be difficult to identify due to texture damage due to external activities. At the same time, the iris is an internal organ protected from external damage because it has a cornea layer [3].

The existence of an authentication process certainly requires a high level of data security. This involves the science of cryptography that functions in data encryption and decryption. Therefore, this research needs to be done to find out how the level of accuracy obtained is based on the initial image with the decrypted image, which is processed using the Rubik's Cube algorithm and the RSA algorithm to find out that the algorithm used can produce good encryption and decryption results. In addition, this study was conducted to determine the efficiency of the time used to carry out the encryption and decryption process with a combination of the Rubik's Cube algorithm and the RSA algorithm on the iris image.

The Rubik's Cube algorithm uses the principle of the Rubik's cube game by making shifts to the matrix of the digital image with two keys, namely the Kr key and the Kc key. The Kr essential functions to shift the matrix rows from left to right, and the Kc key changes the matrix columns up or down. Maximum iteration is also used to determine the number of iterations of image randomisation to be performed. The two keys will be generated using the Boolean Exclusive-OR operator [4].

The RSA algorithm is included in the public key cryptography algorithm. As a general key cryptography algorithm, the RSA algorithm has two keys: the public and private keys. The public key is the key used in the encryption process. In this case, the public key is not kept secret from the public. The private key is used in the decryption process and is kept secret from outside parties other than the sender and recipient. In the RSA algorithm, there are three main processes, namely crucial generation, encryption process, and decryption process. The basis for the encryption and decryption process of the RSA algorithm is modulo arithmetic and the concept of prime numbers [5].

In the previous study, research was carried out using the Rubik's Cube algorithm to carry out the randomization process and use the key from the RSA algorithm which is applied to the XOR operation [6]. Then conducted research using a hybrid encryption

algorithm using the Advanced Encryption Standard (AES) algorithm and the RSA algorithm to encrypt the AES key algorithm [7]. The RSA algorithm is often used because it can provide a high level of security based on a complex key generation process based on modulo calculations and factoring in large prime numbers.

Furthermore, researched to apply encryption and decryption processes for images using Rubik's Cube algorithm [8]. They explained that Rubik's Cube Algorithm is needed as an algorithm that functions to randomize pixels in the image to be used. Rubik's Cube algorithm exhibits good encryption and scrambling capabilities and can withstand statistical and differential attacks. Based on previous research, it was found that there was research using Rubik's Cube Algorithm and the RSA Algorithm by implementing a hybrid encryption algorithm [6].

Therefore, this research is to obtain innovations by combining the two algorithms in the encryption and decryption process without implementing a hybrid encryption algorithm, as has been done in previous research. This research needs to be carried out to determine how the accuracy and time obtained in the encryption and decryption process uses a combination of Rubik's Cube algorithm and the RSA Algorithm.

2 Research Methods

2.1 Research Data

The data used in this study is RGB image data (digital image) with size $M \times N$, 320×240 pixel with *.bmp file fSormat consisting of 5 iris images each left and right from 46 people, with a total of 460 images. The data used in this study is the MMU Iris Dataset which can be obtained for free on the Kaggle website which has been licensed by the Multimedia University Iris Database for Biometric Attendance System at the link <https://www.kaggle.com/naureenmohammad/mmu-iris-dataset>. The data held will be processed using a computer program with a programming language, namely python with the help of the Pillow library.

2.2 Data Analysis Technique

The encryption and decryption process in this study is a combination of Rubik's Cube algorithm and the RSA algorithm. The stages used are starting with encryption using Rubik's Cube algorithm, then followed by encryption using the RSA algorithm. The decryption process is carried out using the RSA algorithm, then followed by decryption using the Rubik's Cube algorithm.

2.2.1 Encryption Process

The encryption process is carried out by applying two algorithms, namely the Rubik's Cube algorithm and the RSA algorithm. Here are some steps that need to be done in the encryption process using the Rubik's Cube algorithm with details:

1. Prepare a plain image, which is a digital image of the iris of the eye with dimensions of $M \times N$, which is 320×240 pixels.

2. Converts the plain image into a matrix form and initializes each plain text matrix entry with the value of the gray level R(Red), G(Green), and B(Blue).
3. Randomly initialize the key numbers to form vectors Kr and Kc with lengths N and M . The elements $Kr(i)$ and $Kc(j)$ will take values in the set $A = \{0, 1, 2, \dots, 2\alpha - 1\}$. The values of Kr and Kc do not have to have constant values.
4. Initializes the number of iterations and maximum iterations and assigns an iteration value of 0.
5. Increment each iteration by one so that it returns the value:
iteration = iteration + 1.
6. Each row i in the matrix I_0 will be processed in the following order:
 - a. Calculate the number of elements in row i which will be denoted as $a(i)$, with the equation:

$$a(i) = \sum_{j=0}^{M-1} I_0(i, j),$$

$$i = 0, 1, \dots, N - 1 \quad (1)$$

- b. Calculate the value of $Ma(i)$ with the function:

$$M\alpha(i) = a(i) \bmod 2 \quad (2)$$

- c. Row i moves left, right, or circularly shifted by position $Kr(i)$ which means that the image pixel will be shifted by position $Kr(i)$ to the right or left, and the first pixel will move to the last pixel. If $M\alpha(i) = 0$ then rotation to the right and otherwise rotation to the left.
7. Each column j in the I_0 matrix will be processed in the following order:
 - a. Calculate the number of elements in column j which will be denoted as $\beta(j)$, with the equation:

$$\beta(j) = \sum_{i=0}^{N-1} I_0(j, i),$$

$$j = 0, 1, \dots, M - 1 \quad (3)$$

- b. Calculate the value of $M\beta(j)$ with the equation:

$$M\beta(j) = \beta(j) \bmod 2 \quad (4)$$

- c. Column j moves up, down, or circularly shifted by the $Kc(j)$ position, which means that the image pixels will be shifted by the $Kc(j)$ position towards the top or bottom. So we get:
If $M\beta(j) = 0$ then rotation is up and otherwise it is rotation down.
8. In steps d and e, you will get a random image matrix called I_{scr} . Then on the I_{scr} matrix, the XOR operation is performed with the Kc vector with the equation:

$$I_1(2i - 1, j) = I_{scr}(2i - 1, j) \oplus Kc(j) \quad (5)$$

$$I_1(2i, j) = I_{scr}(2i, j) \oplus rot180(Kc(j)) \quad (6)$$

where \oplus and $rot180(Kc(j))$ are XOR operation processes and vector shifts Kc from top to bottom or vice versa.

9. In the I_1 matrix, XOR operations with Kr are carried out with the equation:

$$I_{enc}(i, 2j - 1) = I1(i, 2j - 1) \oplus Kr(i) \tag{7}$$

$$I_{enc}(i, 2j) = I1(i, 2j) \oplus rot180Kr(i) \tag{8}$$

where $rot180(Kc)$ is the process of shifting from left to right in the vector Kr .

10. If $ITER = I_{max}$ then the encrypted image matrix will be stored in I_{enc} and if it is not the same then it will be repeated from the third step.

Next are several steps that need to be carried out in the encryption process using the RSA algorithm which consists of a key generation process and an encryption process. The process for generating the RSA key is as follows [9]:

1. Determine the values of p and q .
2. p and q are prime numbers with $p \neq q$.
3. Calculate the value of the modulus (n).

With a value of $n = p.q$

Determine the value of $\Phi(n)$ or the value of totient euler (n).

Where $\Phi(n)$ is the number of positive integers that are less than or equal to n and are prime relative to n .

$\Phi(n)$ can be determined by the formula:

$$\Phi(n) = (p-1) \cdot (q-1) \tag{9}$$

4. Specifies the integer value e (public key).

With a value of e between one and $\Phi(n)$ ($1 < e < \Phi(n)$) which has no divisor of $\Phi(n)$ so that $\gcd(e, \Phi(n)) = 1$. Thus it can be said that e is relative prime to $\Phi(n)$.

The Greatest Common Divisor (\gcd) of two numbers is the intersection of the set of prime factors of the two numbers.

5. Determine the integer value d .

With the value of d is $1 < d < \Phi(n)$, so $e.d = 1 \pmod{\Phi(n)}$. According to the Chinese Remainder Theorem (CRT) equation which states that $a.b \pmod{d}$ is equivalent to $a = b + k.m$, or it can be expressed by the equation:

$$e.d = 1 \pmod{\Phi(n)} \tag{10}$$

By experimenting the value of $d = 0, 1, 2, 3, \dots$ so that it fulfills the equation.

6. Public and private keys are formed based on the key generation process above with the following results:

Public key with pair (e, n) .

Private key with pair (d, n) .

The use of the RSA algorithm in the encryption and decryption functions is carried out in the following order:

1. Take the public key pair in the previous key generation process, namely the (e, n) pair.

2. In the encryption process use the function below to generate a matrix from the encrypted image.

$$X = Y^e \text{ mod } n \quad (11)$$

3. Take the private key pair in the previous key generation process, namely the pair (d, n).
4. In the decryption process, use the function below to generate the initial matrix of the encrypted image.

$$Y = X^d \text{ mod } n \quad (12)$$

2.2.2 Decryption Process

The decryption process is carried out by applying two algorithms, namely the RSA algorithm and the Rubik's Cube algorithm. Here are some steps that need to be done in the decryption process using the RSA algorithm with details:

1. Taking the second encrypt image from the encryption process using the Rubik's Cube algorithm and the RSA algorithm.
2. Converts the second encrypt image to a second encrypt image matrix denoted by (I_{enc2}).
3. Prepare a private key pair for decryption.
4. Operate the private key pair on the second encrypt image matrix (I_{enc2}) which then produces the first decrypt image matrix denoted by (I_{dec}).
5. Transform the matrix (I_{dec}) into the first decrypt image.

Next are several steps that need to be carried out in the decryption process using the Rubik's Cube algorithm with details:

1. Enter the first decrypt image of the decryption process using the RSA algorithm.
2. Converts the first decrypt image to the first decrypt image matrix (I_{dec1}).
3. Enter the vector Kr , vector Kc , and the maximum iteration value.
4. Then Initialize the iteration value = 0 and increment the value iteration = iteration + 1.
5. Operate the XOR operator on the vector Kr of the matrix column (I_{dec1}) to get back the matrix I_1 .
6. Operate the XOR operation on the Kc vector of the row matrix (I_{dec1}) to get back the matrix I_{scr1} .
7. Operate on the vector Kr on the row matrix I_{scr1} and operate on the vector Kc on the column matrix I_{scr1} .
8. If the iteration value = maximum iteration, the encrypted image will be stored in the I_{dec2} matrix and the matrix transformation can be carried out into a second decrypt image (the image is decrypted by the Rubik's Cube algorithm). If the iteration \neq the maximum iteration, it will be repeated from step d until it is fulfilled for the iteration value = maximum iteration.

2.2.3 Evaluation

The evaluation stage is the stage to review the methods that have been applied. Where this process serves to determine the level of success related to the algorithm used in this study. Evaluation in this study will use the method of Structural Similarity Index Metrics (SSIM) and Mean Square Error (MSE). Where both are methods for analyzing the level of similarity of the decrypted image that has gone through the encryption process with the initial image before the encryption process. So, can find out and get how the level of accuracy and success of the encryption and decryption process that has been studied.

Structural Similarity Index Metrics or SSIM is one of the methods used to determine and measure the level of similarity between 2 images [10]. The structural Similarity Index Metrics algorithm is done by comparing structural features. With a straight comparison between structural similarity with image quality. So it can be concluded that the higher the similarity, the higher the image quality, and vice versa. The Structural Similarity Index Metrics algorithm has three main comparisons: luminance distortion, contrast distortion, and correlation. [10] explained that the Structural Similarity Index Metrics equation can be written with the following equation:

$$SSIM(a,b) = l(a,b)c(a,b)s(a,b) \quad (13)$$

Each comparison/feature is as follows:

$$\text{Luminance: } l(a, b) = \frac{2\mu_a\mu_b + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (14)$$

$$\text{Contrast: } c(a, b) = \frac{2\sigma_a\sigma_b + C_2}{\sigma_a^2 + \sigma_b^2 + C_2} \quad (15)$$

$$\text{Structure: } s(a, b) = \frac{\sigma_{ab} + C_3}{\sigma_a\sigma_b + C_3} \quad (16)$$

where C_1, C_2, C_3 are constants to avoid errors due to the denominator = 0. Equation $l(a, b)$ is a comparison used to measure the similarity of the values of the luminance (μ) of the two images being tested. The maximum value of $l(a, b)$ is 1. The maximum value will be fulfilled if the value of $\mu_a = \mu_b$.

The equation $c(a, b)$ is a comparison of the contrast values obtained from the comparison of the standard deviation (σ) of the tested image. Same with equation $l(a, b)$ where the maximum value that can be obtained is 1, provided that $\sigma_a = \sigma_b$.

The equation $s(a, b)$ is a structural comparison that measures the correlation coefficient in the 2 images tested. Where σ_{ab} is the covariance value between a and b .

Structural Similarity Index Metrics produces values from a range of 0 to 1 [11]. Where the value of "0" indicates that the two images are not correlated or not the same. While the value of "1" indicates that the two images tested are similar or exactly the same.

Mean Square Error is one of the calculations to determine the magnitude of the error in the insertion process [12]. In the cryptographic process, MSE can function to determine whether the encryption and decryption processes have good accuracy. This can be obtained by comparing the pixel values of the initial image and the image that has

gone through the decryption process. The mathematical calculation of the Mean Square Error can be described by the following equation:

$$MSE = \frac{1}{m \times n} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (A_{i,j} - B_{i,j}) \quad (17)$$

with:

MSE = Image Mean Square Error Value

m = Decrypted image length

n = Decrypted image width

$A_{i,j}$ = One pixel representation of a plain image

$B_{i,j}$ = One pixel representation of the decrypted image

m* n = Image dimensions

3 Results

In this section, we have previously tested the encryption and decryption process using the python program with the help of a text editor, namely visual studio code. Testing is done by inputting digital images and keys in the text editor to obtain the results of digital images that have been processed and the time it takes to perform the encryption and decryption processes.

The following tests were carried out using the Rubik's Cube algorithm and the RSA algorithm on 3 digital iris images contained in the dataset and taken randomly and using key variations from the RSA algorithm and maximum iterations from different Rubik's Cube algorithms.

It can be seen from the results of the tests carried out that the average time is 0.796 s for the encryption process and 0.652 s for the decryption process using a public key pair (197.403) and a private key pair (53.403) and with a maximum iteration = 1 for the Rubik's Cube algorithm. Then, with the same public and private key pairs but with a maximum iteration = 2 Rubik's Cube algorithm, the average time is 1.174 s for the encryption process and 1.063 s for the decryption process. Furthermore, with a maximum iteration = 3 obtained an average time of 1.557 s for the encryption process and 1.485 s for the decryption process. With a maximum iteration = 4, the average time is 1.954 s for the encryption process and 1.843 s for the decryption process. Then, obtained an average time of 11.349 s for the encryption process and 1.845 s for the decryption process using a public key pair (2741.3233) and a private key pair (461.3233) and with a maximum iteration = 1 for the Rubik's Cube algorithm. By using a public key pair (9551,15151) and a private key pair (671,15151) and with a maximum iteration = 1, the average time is 74,484 s for the encryption process and 3.839 s for the decryption process.

The use of large keys in the RSA algorithm and maximum iterations in the Rubik's Cube algorithm can affect the time required by the system to perform the encryption and decryption process. As shown in Table 1, that the smaller the key value and the maximum iteration specified, the time required for the encryption and decryption process will be relatively faster. With the addition of the value to the key and maximum iteration will require additional time so that the encryption and decryption process is relatively longer.

Next is a test that is carried out using the Structural Similarity Index Metrics (SSIM) and Mean Square Error (MSE) methods to measure the level of accuracy of the structural image between the initial image before the encryption and decryption process and the image that has gone through the encryption and decryption process.



















The value of the Structural Similarity Index Metrics (SSIM) obtained can be a benchmark for the level of structural similarity between the initial image and the encrypted image. The smaller the value of Structural Similarity Index Metrics (SSIM), the better the encrypted image results because the resulting image has a different structure from the initial image. However, the decryption process requires a maximum Structural Similarity Index.

Metrics (SSIM) value of 1 to indicate that the decryption process is successful because the decrypted image has the same structural structure as the initial image. Furthermore, for the Structural Similarity Index Metrics (SSIM) value generated from

Table 1. Test Results Encryption and Decryption Process Time

No.	Trial Name	RSA Algorithm Public Key	RSA Algorithm Private Key	Max Iterations Rubik's Cube Algorithm	Encryption Processing Time (Second)	Decryption Processing Time (Second)
1.	aeval1_1	(197,403)	(53,403)	1	0,794	0,656
2.	aeval1_2	(2741,3233)	(461,3233)	1	11,542	1,936
3.	aeval1_3	(9551,15151)	(671,15151)	1	73,833	3,806
4.	aeval1_4	(197,403)	(53,403)	2	1,201	1,125
5.	aeval1_5	(197,403)	(53,403)	3	1,558	1,446
6.	aeval1_6	(197,403)	(53,403)	4	1,977	1,813
7.	bryan1_1	(197,403)	(53,403)	1	0,793	0,651
8.	bryan1_2	(2741,3233)	(461,3233)	1	11,344	1,843
9.	bryan1_3	(9551,15151)	(671,15151)	1	73,798	3,770
10.	bryan1_4	(197,403)	(53,403)	2	1,145	1,027
11.	bryan1_5	(197,403)	(53,403)	3	1,579	1,498
12.	bryan1_6	(197,403)	(53,403)	4	1,945	1,902
13.	chingycl1_1	(197,403)	(53,403)	1	0,801	0,649
14.	chingycl1_2	(2741,3233)	(461,3233)	1	11,162	1,886
15.	chingycl1_3	(9551,15151)	(671,15151)	1	75,822	3,943
16.	chingycl1_4	(197,403)	(53,403)	2	1,176	1,039
17.	chingycl1_5	(197,403)	(53,403)	3	1,535	1,513
18.	chingycl1_6	(197,403)	(53,403)	4	1,940	1,816

Table 2. Test Results Encryption and Decryption Process Time

No.	Plain Digital Image	Encryption Results	SSIM Encryption Value	Encryption MSE Value	Decrypted image	SSIM Value Decrypt	MSE Value Decrypt
1.			0,01	43840.11		1	0
2.			0,01	38397.26		1	0
3.			0,01	40326.03		1	0
4.			0,01	43832		1	0
5.			0,01	44110.59		1	0
6.			0,01	43936.16		1	0

the encryption and decryption process in the test using the same iris digital image but different key values and maximum iterations, the SSIM value is the same.

In Table 2, it can be seen the results of tests carried out with a total of 6 experiments on 1 digital image of the iris of the eye using different public and private keys for the RSA algorithm and different maximum iterations for the Rubik's Cube algorithm. In the encryption process, the average value of Structural Similarity Index Metrics (SSIM) is 0.01 in 18 tests with the same 3 digital irises. This shows that the digital image of the iris produced in the encryption process is not identical or the same as the initial digital image of the iris. Then in the decryption process, the average value of Structural Similarity Index Metrics (SSIM) is 1 for all tests. This shows that the digital image of the iris that has gone through the encryption and decryption process is identical or the same as the initial digital image of the iris.

Furthermore, in the cryptographic process, Mean Square Error (MSE) can function to determine whether the encryption and decryption processes have good accuracy. For the Mean Square Error (MSE) value, various results were obtained in the encryption process with an average value of 37620.59 in 18 tests with different key values and maximum iterations. In the encryption process, a larger key value can result in a smaller Mean Square Error (MSE) value. However, the value shown still shows a high Mean Square Error (MSE) value. Then, by adding the maximum iteration value to the encryption process, it does not show a significant difference in the resulting Mean Square Error (MSE) value. In the decryption process, the Mean Square Error (MSE) value is obtained with an average of 0. This indicates that the decrypted image and the initial digital iris image are identical or the same and have no error value.

4 Conclusion

In the encryption and decryption process using the Rubik's Cube algorithm and the RSA algorithm, it was found that the maximum key value and iteration were directly proportional to the time required for the encryption and decryption process. The greater the value of the key pair in the RSA algorithm and the maximum iteration value of the Rubik's Cube algorithm, the longer the time required to perform the encryption and decryption process. From the test results obtained an average time of 0.796 s for the encryption process and 0.652 s for the decryption process using a public key pair (197.403) and a private key pair (53.403) and with a maximum iteration = 1 for the Rubik's Cube algorithm.

In the encryption and decryption process using the Rubik's Cube algorithm and the RSA algorithm, it was found that the encrypted digital image was random and not identical to the initial digital image of the iris. With the average value of Similarity Index Metrics (SSIM) is 0.01 and the average value of Mean Square Error (MSE) is 37620.59 in the encryption process. Furthermore, for the results of the decryption process, the average value of Similarity Index Metrics (SSIM) is 1 and the average value of Mean Square Error (MSE) is 0 which proves that the decryption process has succeeded in obtaining a digital image of the iris of the eye that is identical to the digital image of the initial iris.

Authors' Contributions. The authors conceived of the presented idea. S.H.M. developed the theory and performed the computations. M.K. verified the analytical methods. S.H.M and M.K. encouraged to investigate and supervised the findings of this work. All authors discussed the results and contributed to the final manuscript.

References

1. J. Nader, A. Alsadoon, P. W. C. Prasad, A. K. Singh, and A. Elchouemi, "Designing Touch-Based Hybrid Authentication Method for Smartphones," in *Procedia Computer Science*, 2015, vol. 70, pp. 198–204. doi: <https://doi.org/10.1016/j.procs.2015.10.072>.
2. E. G. Kristanto, E. Rompas, and S. Wangko, "Identifikasi Iris Opsi Identifikasi Iris," *Jurnal Biomedik*, vol. 5, pp. S7-11, 2013.
3. S. Vatsal and Mr. S. S. Dwivedi, "Advanced IRIS Recognition System: A Review," May 2018.
4. Y. A. Primadhana, R. A. Asmara, and A. R. T. H. Ririd. "Enkripsi Citra Menggunakan Algoritma Kubus Rubik Dengan Pembangkit Kunci Md5." *Jurnal Informatika Polinema*, 2016, vol. 3 no. 1, p. 40. Doi: <https://doi.org/10.33795/jip.v3i1.21>.
5. A. Ginting, R. R. Isnanto, and I. P. Windasari, "Implementasi Algoritma Kriptografi RSA untuk Enkripsi dan Dekripsi Email," *Jurnal Teknologi dan Sistem Komputer*, vol. 3, pp. 253–258, 2015.
6. L. J. Julus, "A Secured Image Encryption Algorithm Based On Rubik's Cube Principle with RSA Encryption," 2022. [Online]. Available: www.ijcrt.org
7. B. M. Belkaid, "Meteosat Images Encryption based on AES and RSA Algorithms Meteosat Image Encryption," 2015. [Online]. Available: www.ijacsa.thesai.org
8. A. V. Diaconu and K. Loukhaoukha, "An improved secure image encryption algorithm based on rubik's cube principle and digital chaotic cipher," *Math Probl Eng*, vol. 2013, 2013, doi: <https://doi.org/10.1155/2013/848392>.

9. J. Khairil Azhar and S. Yuliany, "Implementasi Algoritma RSA (Rivest, Shamir dan Adleman) untuk Enkripsi dan Dekripsi File .pdf," Tasikmalaya, Dec. 2019.
10. H. B. Sumarna, E. Utami, and A. D. Hartanto, "Tinjauan Literatur Sistematis tentang Structural Similarity Index Measure untuk Deteksi Anomali Gambar Systematic Literature Review of Structural Similarity Index Measure for Image Anomaly Detection," *Citec Journal*, vol. 7, no. 2, 2020.
11. U. Sara, M. Akter, and M. S. Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study," *Journal of Computer and Communications*, vol. 07, no. 03, pp. 8–18, 2019, doi: <https://doi.org/10.4236/jcc.2019.73002>.
12. G. M. Male, Wirawan, and E. Setijadi, "Analisa Kualitas Citra pada Steganografi untuk Aplikasi e-Government," 2012.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

