ORIGINAL RESEARCH ARTICLE UHD JOURNAL OF SCIENCE AND TECHNOLOGY

Modified Advanced Encryption Standard for Boost Image Encryption

Nigar M. Shafiq Surameery

Department of Information Technology, College of Computer and Information Technology, University of Garmian, Kalar, Sulaimani, Kurdistan Region, Iraq

ABSTRACT

Cryptography is a field of study that deals with converting data from a readable to an unreadable format. It can provide secrecy, data integrity, authenticity, and non-repudiation services. Security has become a concern for the community because of the technology's potential use in numerous sectors of any company, market, agency, or governmental body, information. The cryptosystems ensure that data are transported securely and only authorized individuals have access to it. Deeply encrypted data that cannot be deciphered through cryptanalysis are in high demand right now. There are a variety of encryption algorithms that can guarantee the confidentiality of data. For multimedia data, standard symmetric encryption algorithms (AES) can give superior protection. However, using the symmetric key encryption approach on more complicated multimedia data (mainly photos) may result in a computational issue. To address this issue, the AES has been modified to satisfy the high computing requirements due to the complex mathematical operations in MixColumns transformation, which slow down the encryption process. The modified AES uses bit permutation to replace the MixColumns transformation in AES because it is simple to construct and does not require any complex mathematical computation. This research focuses on using the Modified Advanced Encryption Standard (MAES) algorithm with 128 and 256 bit key sizes to encrypt and decrypt image data. The algorithms were implemented using the Python programming language without complex mathematical computation. By comparing the MAES algorithm with the original AES algorithm, the results showed that the MAES requires less encrypting and decryption time with higher efficiency for all file sizes.

Index Terms: Advanced Encryption Standard, Bit Permutation, Image File Encryption, Symmetric Cipher, Modified Advanced Encryption Standard

1. INTRODUCTION

The world is evolving due to the advancements in technology, attitudes, and investment. Massive amounts of data are being sent, received, and stored by government, commercial, consumer, and manufacturing groups for intelligence, safety, and other purposes such as medical, video monitoring, and IoT systems. The security requirements for images originate from the need to protect its data from two types of threats: Unintentional loss and corruption, and intentional unauthorized access or manipulation. Cybersecurity is generally recognized as the most basic and effective means of protecting images against both unintentional and intentional fraud [1]. Images data (original data) are converted into an encrypted version (encrypted information) to be saved or sent through an unsecured channel. Encrypted transmission is safe if only the sender and receiver could access the cipher and decipher algorithms and keys. The process of ensuring the integrity, availability, and privacy of data is known as information security.

Access this article online

DOI:10.21928/uhdjst.v6n1y2022.pp52-59

E-ISSN: 2521-4217

P-ISSN: 2521-4209

Corresponding author's e-mail: Nigar M. Shafiq Surameery, Department of Information Technology, College of Computer and Information Technology, University of Garmian, Kalar, Sulaimani, Kurdistan Region, Iraq. E-mail: nigar.mahmoud@garmian.edu.krd

Received: 31-12-2021 Accepted: 01-04-2022 Published: 27-04-2022

Nowadays, the majority of businesses and individuals keep their data on computers. Thus, people have more access to information kept in computer databases. Much of the Stored data are very sensitive and should not be shared with the general public. Data encryption is a common technique for protecting secret information by encrypting it with a pre-existing or pre-written algorithm. The key generation is the most potent encryption aspect, which is divided into two parts: Symmetric key generation and asymmetric key generation. With the help of modern high-performance computing devices, hackers may quickly break the key. Therefore, encrypted data that cryptographic analyzers cannot decrypt are currently in demand [2]. The encryption and decryption processes are affected by the encrypted file size and the length of the key [3]. The key used for encryption/decryption in symmetric cryptography is the same. As a result, the required distribution must occur before information transfer. In symmetric key cryptography, as shown in Fig. 1, the key is highly significant because the system's security is directly related to the key's characteristics, such as key length. Advanced Encryption Standard (AES), DES, TRIPLE DES, BLOWFISH, RC4, and RC6 are examples of symmetric cryptography [4].

AES is one of the most used symmetric algorithms and a new generation of data encryption standards. It offers the most outstanding safety and highest operation speed [5]. The AES is the next-generation cryptographic system with many benefits, including strict security, high efficiency, customization, and convenience.

The research establishment has shown keen interest in image encryption to protect valuable images from hackers in recent years. As a result, several encryption algorithms have been used for encrypting images to achieve this goal. This research aims to show the efficiency and speed of the modified AES algorithm compared to the original AES.

2. LITERATURE REVIEW

With the rapid growth of digital data interchange, secure data transfer is becoming increasingly crucial. since images are increasingly being used in different processes, it is vital to keep confidential image data safe from unauthorized access [6]. In Maolood *et al.* [7], modifications were made to the original AES algorithm to increase security. Using evaluation measures, the effectiveness of the modifications was demonstrated, and the improved algorithm was evaluated. The results showed a small increase in the encryption time of the modified AES algorithm compared to the original AES algorithm because of adding several layers of security. The authors in Ahamad and Abdullah [8], compared the four encryption algorithms (Blowfish, AES, XOR, and RSA), applied to four types of data represented by: Text, image, audio, and video. The simulations demonstrate that AES is more time efficient than all other methods. In another work [9], the AES round numbers have been increased to 16 to be more secure. Theoretical study and practical results revealed that this AES strategy provides great speeds and less data transmission over unprotected networks. Moreover, the research study in Arman *et al.* [10] focuses on symmetric Key cryptosystem represented by the AES algorithm and proposes a significant improvement over AES in terms of privacy and ease of operation. The goal was to achieve a rapid implementation time and low memory consumption while maintaining a high security level. As a result, this solution obtained 88% of efficiency and a faster net processing time. This is a big change in terms of efficiency over the standard AES. In the more advanced researches, to address the poor diffusion rate in the early rounds, the AES cipher round and key schedule have been updated to include more primitive operations such exclusive OR and modulo operations [11]. According to the avalanche effect evaluation results, the results show that the modified AES has enhanced propagation and ambiguity features because there was a 61.98% increase in dispersal in round 1, 14.79% in round 2, and 13.87% in round 3. However, in Shakor and Surameery [12], AES is combined with elliptic curve cryptography (ECC) algorithm which has been used to protect the sensitive COVID-19 files that are stored in the cloud. Although different layers of security have been added, the results indicated that the impact of the hybrid system was more significant than the other secure algorithms without

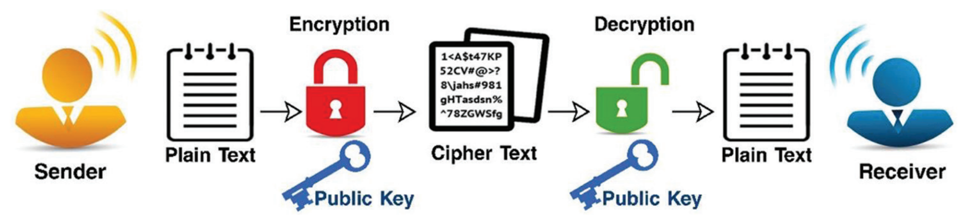

Fig. 1. Symmetric key cryptography.

affecting the consumption time. The author in Mohd and Ashawesh [13] proposed a secured modified AES (MAES) algorithm, which reduces the rounds of AES to 14 rounds to decrees the consumption time of the encryption and decryption process, and improve the digital data security. The results prove that, compared with other studies, the proposed method uses less encryption and decryption time while providing better security, as confirmed by the avalanche effect test. Moreover, in Lin *et al.* [14], an improved AES cryptosystem based on chaotic synchronization with dynamic random keys is proposed. In the traditional AES encryption system, static keys are used, which must be exchanged in advance and confirmed for safe preservation. The chaotic system's synchronization technology, on the other hand, was used to overcome the inadequacies of key storage. In the proposed approach, the static key becomes dynamic and random, and it no longer needs to be kept or transmitted over an open channel. In Hafsa *et al.* [15], the authors use an efficient type of a built-in approach that utilized the AES and elliptic curve encryption (ECC) algorithms for encrypting medical images. It combines the advantages of symmetric AES accelerated data encryption and asymmetric ECC to ensure the exchange of symmetric session keys. The security analysis was successfully performed, and their experiments proved that the proposed technology provides a more straightforward and correct cryptographic basis. Furthermore, the evaluation results demonstrate the proposed algorithm's effectiveness, rapidity, and high security.

Hence, after studying various previous researches on the modification of AES, we came up with a new technique that increases the processing speed and data security.

3. AES

The AES is a symmetric key method that has been developed by the United States National Institute of Standards and Technology as the standard for encrypting digital data. AES is the most powerful algorithm in recent times since it is the only vulnerable to brute force attacks, which makes it difficult for cryptanalysts to crack. It is commonly used in banks and organizations to protect critical information because it promises cybersecurity [16]. The size of the key determines the number of rounds for encryption and decryption in AES, which could be 10, 12, or 14 rounds for 128, 192, and 256 bit keys, respectively [17]. AES is one of the most widely used symmetric key algorithms and a new generation of data encryption standards. It offers the greatest secrecy and highest operation speed [5].

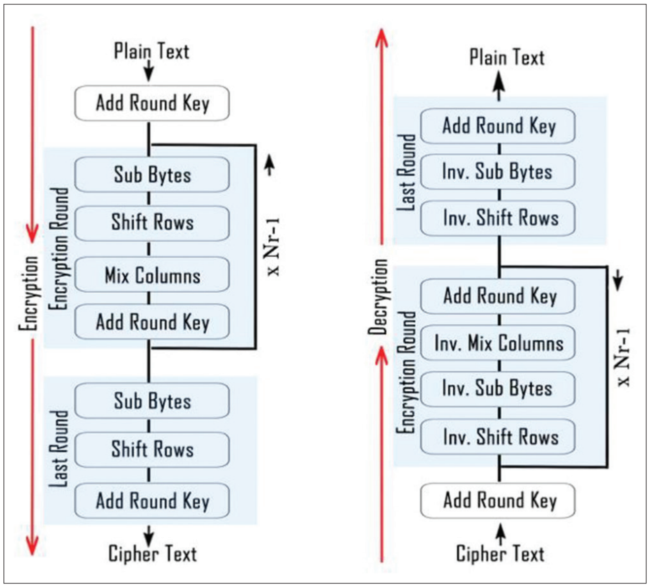

Fig. 2. Structure of AES algorithm.

As shown in Fig. 2, the AES algorithm comprises four invertible transformations: SubBytes, ShiftRows, MixColumns, and AddRoundKey. All of the encryption rounds use these transforms, with the exception of the end round, which removes the MixColumns transformation to make the encryption and decryption methods symmetric.

- SubBytes transform: Which substitutes each byte of the original block of data (D0,…., D15) with the row (initial 4 bits) and column (next 4 bits) of a 16 × 16 Substitution Box (S-Box). The S-Box includes unique mathematical qualities that guarantee changes in individual state bits spread swiftly throughout the ciphertext, causing confusion. During decryption, a reverse substitution box (InvS-Box) is utilized to reverse the impact of the SubBytes translation.
- ShiftRows Transform: This manipulates the state's rows by shifting the bytes in each row using a specific offset. The first row remains constant during this process, while the second, third, and fourth rows are subjected to 1 byte, 2 byte, and 3 byte circular shift operations. The first row remains untouched throughout the decryption process, while the following rows are moved to the right using the same offset that was used to shift them to the left during the encryption process
- MixColumns transform: Represents the mixing operation that applies the XOR operation on the state's columns, combining the four bytes in each column with the four bytes in a fixed matrix.
- AddRoundKey Transform: This is the last transformation that will be applied for each round. The XOR operation

is employed to execute addition operation between the bytes of the modified state and the round key.

The encryption process consists of a number of different transformations applied consecutively over the data block bits, in a fixed number of iterations, called rounds. The number of rounds depends on the length of the key used for the encryption process. The AES encryption process is as follows

```
Cipher(byte in[X*Nb], byte out [X*Nb], word w[Nv*(Nr+1)])

Begin
Byte state[X, Nb]
State=in
AddRoundKey(state, w[0,Nb-1])
For round=1 step 1 to Nr-1
SubBytes(state)
ShifRows(state)
MixColumns(state)
AddRounKey(state, w[round*Nb, (round+1)*Nb-1])
end for
SubBytes(state)
ShifRows(state)
AddRounKey(state, w[Nr*Nb, (Nr+1)*Nb-1])
Out=state
End
```

4. MAES ALGORITHM

To secure the image files and increase their privacy, a modified AES algorithm is employed to encrypt them. The AES is modified to address its higher computational requirement due to the complex mathematical operations in MixColumns transformation making the encryption process slow. Bit permutation includes merely shifting the positions of bits in each stage and does not require any significant mathematical computation. The bit permutation transformation takes the place of the MixColumns transformation in AES, Fig. 3, depicts the bit permutation modification of AES.

For the encryption process, take the state value per column (for example, column 0) from ShiftRows transformation. Each column has four rows, as shown in Fig. 4. Moreover, each state in column 0 is composed of eight bits, represented as ((x,0), b) where x, 0, and b indicate the row, column, and bit number for each state, respectively. Subsequently, the states are partitioned into four blocks matrix, called block 0, block 1, block 2, and block 3 and the transpose of each block matrix is the next step of the process. Finally, to get the new value at state a'(x,y), we need to do a row-wise concatenation of the bit values of the transposed block where x is the column_value and y is the block_value. MAES encryption process is as follows:

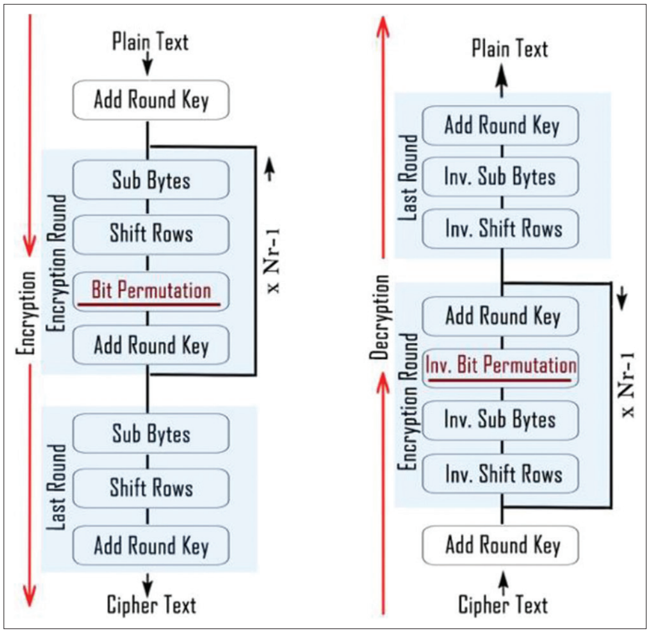

Fig. 3. Structure of MAES algorithm.

```
Cipher(byte in[X*Nb], byte out [X*Nb], word w[Nv*(Nr+1)])

Begin
Byte state[X, Nb]
State=in
AddRoundKey(state, w[0,Nb-1])
For round=1 step 1 to Nr-1
SubBytes(state)
ShifRows(state)
Bit Permutation(state)
AddRounKey(state, w[round*Nb, (round+1)*Nb-1])
end for
SubBytes(state)
ShifRows(state)
AddRounKey(state, w[Nr*Nb, (Nr+1)*Nb-1])
Out=state
End
```

For the decryption process, the inverse bit permutation is utilized. There are four columns in each row. In the first steps, starting with row 0, we take the state value from the InvSubBytes transformation. Next, each state is made up of 8 bits, denoted by ((0, y), b), where 0 denotes row 0, y

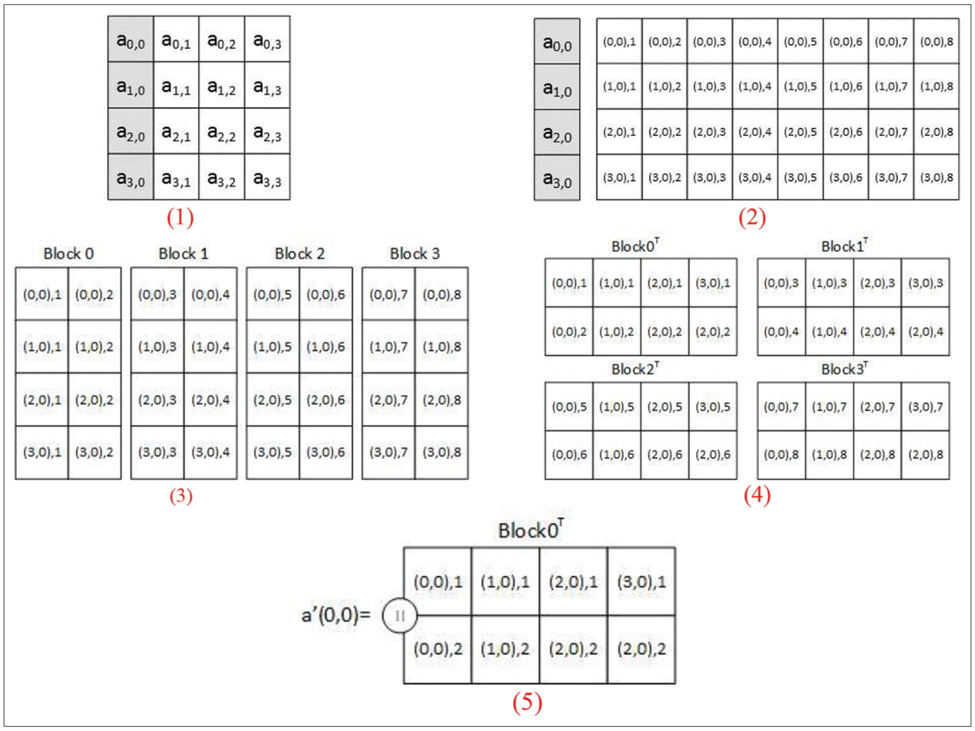

Fig. 4. Bit permutation.

denotes column, and b denotes the bit position in each state. Taking the number of bits per state creates 1×32 matrix. Subsequently, the matrix is partitioned into eight blocks, called block 0 through block 7. After that, the blocks 0 to 7 are transposed, yielding eight 4×1 transposed matrices. Finally, to get the new value at state a'(x,y), we need to do row-wise concatenation of the bit values of the transposed block, where x is the block_value and y is the row_value. Fig. 5.

5. EXPERIMENTAL RESULTS AND DISCUSSION

This section shows the experiment run to derive the effect of image size on the encryption and decryption time and includes a comparison between the modified AES system and the existing standard AES solution. To apply security, AES-128, AES-256, MAES-128, and MAES-256 algorithms were used in the encryption and decryption processes.

These two processes were implemented using the Python programming language. Both standard and modified AES algorithms are written in Spyder, the Scientific Python Development Environment, and simulated on Intel Core i7-2.8 GHz processor with 16 GB RAM and running over Mac OS High Sierra.

The main purpose of this study is to calculate the encryption and decryption speed of each of the algorithm under the study of different image sizes. The AES and MAES algorithms were evaluated for 10 trials to determine the average encryption and decryption time. Their implementation is tried to optimize the maximum performance for the algorithm. Therefore, the throughput for encryption is calculated for each algorithm. Encryption time is used to calculate the throughput of an encryption scheme. The throughput of the encryption scheme is calculated by dividing the total plain image size in kilobytes by total encryption time in second for each algorithm. If the throughput value is increased, the power consumption of this encryption technique is decreased.

Throughput = Plaintext (KB)/Encryption time (Sec.)

In this research, different sizes of image files were experimented such as 200 KB, 400 KB, 800 KB, and 1600 KB. The performance metrics are analyzed as encryption and decryption time and throughput. The performance of the encryption and decryption time was measured in milliseconds.

For the 128 bit key, it was observed that both the encryption and decryption times increase as the image size increases, as

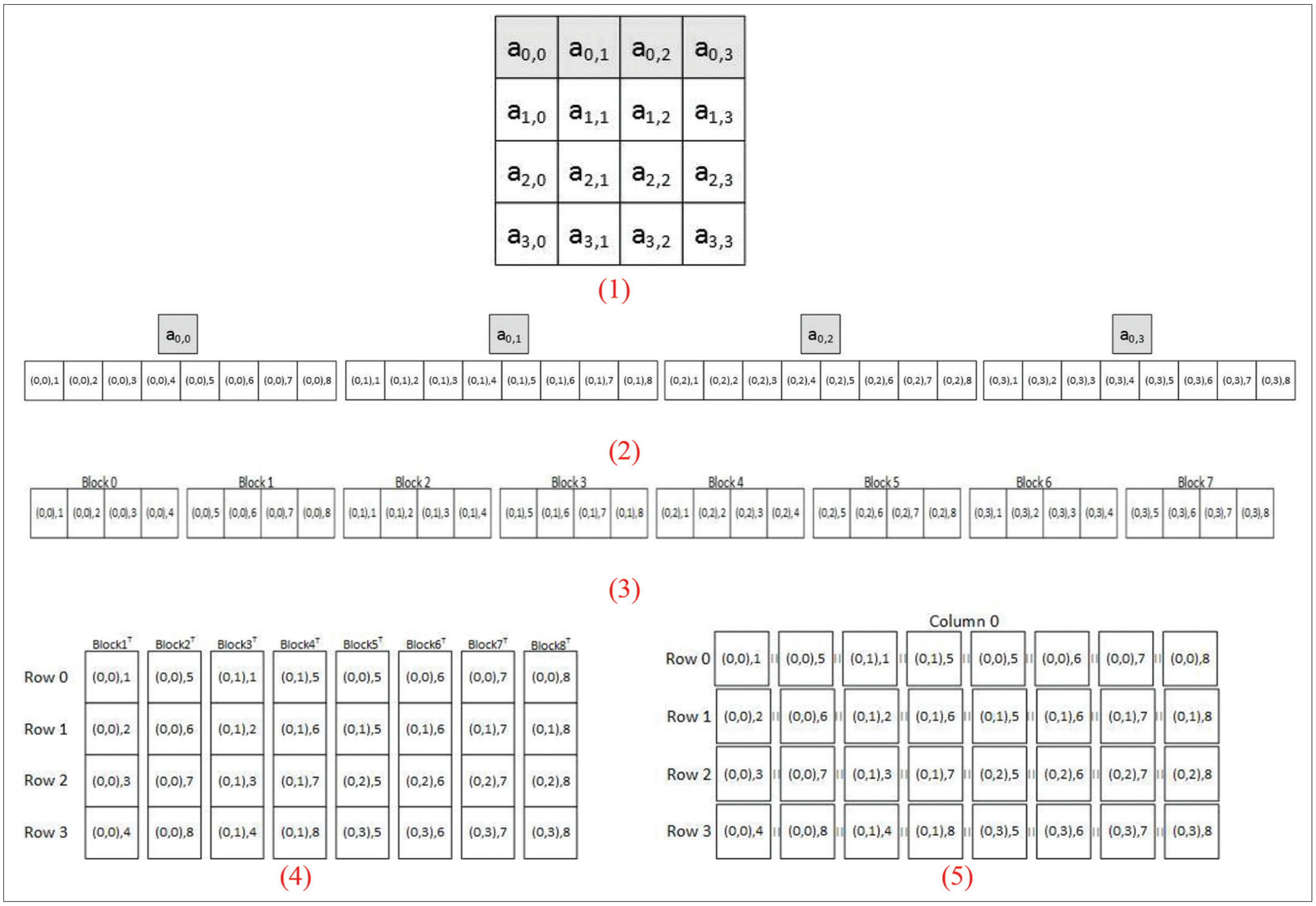
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
(1)
(0,0),1 (0,0),2 (0,0),3 (0,0),4 (0,0),5 (0,0),6 (0,0),7 (0,0),8
(0,1),1 (0,1),2 (0,1),3 (0,1),4 (0,1),5 (0,1),6 (0,1),7 (0,1),8
(0,2),1 (0,2),2 (0,2),3 (0,2),4 (0,2),5 (0,2),6 (0,2),7 (0,2),8
(0,3),1 (0,3),2 (0,3),3 (0,3),4 (0,3),5 (0,3),6 (0,3),7 (0,3),8
(2)
Block 0 Block 1 Block 2 Block 3 Block 4 Block 5 Block 6 Block 7
(3)
Block1^T Block2^T Block3^T Block4^T Block5^T Block6^T Block7^T Block8^T
Row 0 (0,0),1 (0,0),5 (0,1),1 (0,1),5 (0,0),5 (0,0),6 (0,0),7 (0,0),8
Row 1 (0,0),2 (0,0),6 (0,1),2 (0,1),6 (0,1),5 (0,1),6 (0,1),7 (0,1),8
Row 2 (0,0),3 (0,0),7 (0,1),3 (0,1),7 (0,2),5 (0,2),6 (0,2),7 (0,2),8
Row 3 (0,0),4 (0,0),8 (0,1),4 (0,1),8 (0,3),5 (0,3),6 (0,3),7 (0,3),8
(4)
Column 0
Row 0 (0,0),1 (0,0),5 (0,1),1 (0,1),5 (0,0),5 (0,0),6 (0,0),7 (0,0),8
Row 1 (0,0),2 (0,0),6 (0,1),2 (0,1),6 (0,1),5 (0,1),6 (0,1),7 (0,1),8
Row 2 (0,0),3 (0,0),7 (0,1),3 (0,1),7 (0,2),5 (0,2),6 (0,2),7 (0,2),8
Row 3 (0,0),4 (0,0),8 (0,1),4 (0,1),8 (0,3),5 (0,3),6 (0,3),7 (0,3),8
(5)

Fig. 5. Inverse bit permutation.

TABLE 1: Comparison of AES and MAES encryption time using 128 bit key size

File size	Encryption time (ms)	
	AES	Modified AES
200 KB	28.24	27.22
400 KB	52.11	51.26
800 KB	113.11	111.45
1600 KB	242.89	240.04
Average time	436.35	429.97
Throughput	6.87	6.97

TABLE 2: Comparison of AES and MAES decryption time using 128 bit key size

File size	Decryption time (ms)	
	AES	Modified AES
200 KB	29.15	28.70
400 KB	57.02	55.06
800 KB	119.06	117.82
1600 KB	244.88	242.55
Average time	450.11	444.13
Throughput	6.6650	6.754779

TABLE 3: Comparison of AES and MAES encryption time using 256 bit key size

File size	Encryption time (ms)	
	AES	Modified AES
200 KB	30.21	26.01
400 KB	58.49	56.11
800 KB	117.24	115.89
1600 KB	249.33	243.34
Average time	455.27	441.35
Throughput	6.589	6.797

TABLE 4: Comparison of AES and MAES decryption time using 256 bit key size

File size	Decryption time (ms)	
	AES	Modified AES
200 KB	30.95	29.95
400 KB	60.40	59.11
800 KB	124.61	122.71
1600 KB	256.24	254.43
Average time	472.2	466.2
Throughput	6.353	6.435

TABLE 5: Comparison of proposed method with another research solutions

Author	Image size	Key size	Encryption time		Decryption time	
			AES (ms)	Modified AES (ms)	AES (ms)	Modified AES (ms)
[18]	200	128 bit	206,016	196,576	176,618	167,532
This study	200	128 bit	28.24	27.22	29.15	28.70

shown in Tables 1 and 2. However, it was noticed that the encryption and decryption processes of the MAES take less time than the standard one for the same image size.

Regarding the enciphering process, for the file sizes of 200, 400, 800, and 1600 KB, the encryption time of the AES was 28.24, 52.11, 113.11, and 242.89 ms, respectively. In comparison, the encryption time of the MAES for the same file sizes were 27.22, 51.26, 111.45, and 240.04 ms, respectively. Moreover, the decryption time of the AES was 28.24, 52.11, 113.11, and 242.89 ms, respectively. In contrast, the decryption time of the MAES for the same file sizes were 27.22, 51.26, 111.45, and 240.04 ms, respectively.

On the other hand, ultra-high security may be achieved using AES-256 due to the number of rounds, which provides privacy to unauthorized users. Therefore, the system has been tested with AES-256 and MAES-256 for different file sizes. For the file sizes of 200, 400, 800, and 1600 KB, the encryption process of MAES-256 was faster than AES-256 by 0.74, 1.91, 7.37, and 6.91 ms, respectively. In the same way, the decryption process of MAES-256 was faster than AES-256 by 3.94, 3.00, 6.82, and 11.0991 ms, respectively, as shown in Tables 3 and 4. Thus, in comparison with AES, the presented results show that slightly less encryption and decryption times in milliseconds were recorded with the bit permutation in MAES for all of the image file sizes, which increases the security, overall performance, and effectiveness of the system.

By considering the results in Tables 1-4, the average execution time for image data types of AES are 436.35, 450.11, 455.27, and 472.2, respectively, and MAES are 429.97, 444.13, 441.35, and 466.2, respectively. Similarly, the throughput are (6.87, 6.6650, 6.589, and 6.353) and (6.97, 6.754779, 6.797, and 6.435) for AES and MAES, respectively. The results show that the standard AES algorithm has higher encryption average time and the MAES has higher throughput. Hence, MAES is more efficient in image data encryption than AES.

Besides, a comparative analysis of the obtained results was carried out with an existing work existing works. This was a little bit of tasking, as there are no standard performance metrics that are widely and generally acceptable by all researchers in this regard. While some measured the performance of their modified AES version using text files of different sizes, some used images and video files. However, most authors employed execution time as their performance metrics and just few researches employed the encryption and decryption times, separately.

Therefore, the proposed technique was compared with the recent solutions for the “Implementation of Modified AES as Image Encryption Scheme” that was proposed in 2018 [18]. The research was available to investigate the use of modified AES for image encryption.

As presented in Table 5, the proposed technique provides optimistic solutions by recording lower execution time in milliseconds than other research solution. Although this may not be clearly noticeable in real-life application, yet it is significant.

6. CONCLUSION

Cryptography is a widely used approach for ensuring data transfer and image storage security. The proposed algorithm aims to reinforce and improve the standard AES method by replacing the MixColumns transformation with bit permutation. With different file sizes, the new algorithm is tested for encryption/decryption time in milliseconds. The results reveal that the MAES achieves a higher level of security with requiring less time. The key sizes used to compare AES and MAES were 128 bits and 256 bits.

The results showed that the encrypting time in the files whose size was between 200 and 1600 KB in the MAES algorithm was less than AES by 1.595% with the128 bit key size, while MAES has a lower percentage of encryption time than AES by 3.48% when applying a 256 bit key size. In the same way, MAES decryption time was 1.495% faster than AES with a 128 bit key size and 1.5% faster with a 256 bit key size. In addition, the results show that the standard AES algorithm has higher encryption average time and the MAES has higher throughput. Hence, MAES is more efficient in image data encryption than AES. Furthermore, the proposed MAES cryptosystem provides improved security in terms of protection against different types of attacks. As a part of future work, we can use additional linear processes experiments to boost AESs confusion property.

REFERENCES

[1] S. M. Ali Ebrahim. "Hybrid chaotic method for medical images ciphering. *The International Journal of Network Security and Its Applications*, vol. 12, no. 6, pp. 1-14, 2020.

[2] P. Dixit, A. K. Gupta, M. C. Trivedi and V. K. Yadav. "Traditional and hybrid encryption techniques: A survey". *Lecture Notes on Data Engineering and Communications Technologies*, vol. 4, pp. 239-248, 2018.

[3] R. Yudistira. "AES (advanced encryption standard) and RSA (rivest-shamir-adleman) encryption on digital signature document: A literature review". *International Journal of Information Technology and Business*, vol. 2, no. 1, pp. 1-3, 2020.

[4] M. Agrawal and P. Mishra. "A comparative survey on symmetric key encryption techniques". *International Journal of Computational Science and Engineering*, vol. 4, no. 5, pp. 877-882, 2012.

[5] J. Liu, C. Fan, X. Tian and Q. Ding. "Optimization of AES and RSA algorithm and its mixed encryption system". In; *Smart Innovation, Systems and Technologies.* Vol. 82, pp. 393-403, 2018.

[6] M. Zeghid, M. Machhout, L. Khriji, A. Baganne and R. Tourki. "A modified AES based algorithm for image encryption". *International Journal of Computational Science and Engineering*, vol. 1, no. 1, p. 70, 2007.

[7] A. T. Maolood and Y. A. Yasser. "Modifying Advanced Encryption Standard (AES)". Algorithm University of Technology Department of Computer Sciences University of Technology Department of Computer Sciences, pp. 259-285, 2017.

[8] M. M. Ahamad and M. I. Abdullah. "Comparison of encryption algorithms for multimedia". *Rajshahi University Journal of Science and Engineering*, vol. 44, pp. 131-139, 2016.

[9] B. N. Rao, D. Tejaswi, K. A. Varshini, K. P. Shankar and B. Prasanth. "Design of modified AES algorithm for data security". *International Journal For Technological Research In Engineering*, vol. 4, no. 8, pp. 1289-1292.

[10] M. S. Arman, T. Rehnuma and M. M. Rahman. "Design and Implementation of a Modified AES Cryptography with Fast Key Generation Technique". *Proceedings of 2020 IEEE International Women in Engineering* (*WIE*) *Conference on Electrical and Computer Engineering* (*WIECON-ECE*), pp. 191-195, 2020.

[11] E. M. De Los Reyes, A. M. Sison and R. P. Medina. "Modified AES cipher round and key schedule". *Indonesian Journal of Electrical Engineering and Informatics*, vol. 7, no. 1, pp. 28-35, 2019.

[12] M. Y. Shakor and N. M. S. Surameery. "Built-in Encrypted Health Cloud Environment for Sharing COVID-19 Data". In: *3rd International Conference on Computer Communication and the Internet, ICCCI 2021*, pp. 96-101, 2021.

[13] N. A. A. Mohd and A. Y. A. Ashawesh. "Enhanced AES algorithm based on 14 rounds in securing data and minimizing processing time". *Journal of Physics: Conference Series*, vol. 1793, no. 1, p. 012066, 2021.

[14] C. H. Lin, G. H. Hu, C. Y. Chan and J. J. Yan. "Chaos-based synchronized dynamic keys and their application to image encryption with an improved aes algorithm". *Applied Sciences*, vol. 11, no. 3, pp. 1-16, 2021.

[15] A. Hafsa, A. Sghaier, J. Malek and M. Machhout. "Image encryption method based on improved ECC and modified AES algorithm". *Multimedia Tools and Applications*, vol. 80, no. 13, pp. 19769-19801, 2021.

[16] A. Gupta and M. Jaiswal. "*The Safety of Next Generation Internet of Things*". pp. 422-427, 2017.

[17] P. V. Jaswanth, B. R. Reddy, M. S. P. Kumar and M. J. P. Priyadarsini. "Color image encryption using AES and RSA". *The International Journal of Engineering and Advanced Technology*, vol. 9, no. 5, pp. 547-550, 2020.

[18] H. V. Gamido, A. M. Sison and R. P. Medina. "Implementation of modified AES as image encryption scheme". *Indonesian Journal of Electrical Engineering and Informatics*, vol. 6, no. 3, pp. 301-308, 2018.