*VFAST Transactions on Computer Sciences*
*http://vfast.org/journals/index.php/VTCS@ 2017 ISSN(e): 2308-8168; ISSN(p): 2411-6335*
*Volume 5, Number 1, January-December, 2017*

*pp. 09-17*

# DIGITAL IMAGE ENCRYPTION IMPLEMENTATIONS BASED ON AES ALGORITHM

AHMAD ABDULQADIR ALRABABAH*[1], MUASAAD ALRASHEEDI[2]
[1]Faculty of Computing and Information Technology in Rabigh, King Abdulaziz University, Rabigh 21911, KSA. E-mail: aahmad13@kau.edu.sa.
[2]Faculty of Computer Study, Arab Open University, KSA Branch

Revised May 2017

ABSTRACT: *Objectives: To increase needed for exchanging digital photos electronically, due to alarming demand for multimedia applications, and because of the increasing use of images in electronic processes. Hence, the need for protection by unauthorized user is necessary. Method: This paper primarily is focusing on the necessary protection of these images using a specific analyzes algorithm: Advanced Encryption Standard (AES) with a full its description, which is known as an algorithm (Rijndael). Findings: It will be determined the address decryption, which is made up of different styles in all encryption and decryption steps in order to protect the valuable information. This algorithm will be implemented on MATLAB software programming. Application: The above results and analysis for this crypto system based on AES algorithm give a high performance. So we have reason to believe that use this method to encrypt the image will have a very good prospect in the future.*
***Keywords:*** *Encryption Advanced Encryption Standard (AES), Cryptography, MATLAB.*

**1**. **Introduction**. In recent years; the technology advancement has increased the possibility to transfer digital images, which can be easily found online. The encrypted images are converted from digital data to privacy. The application of most keys and algorithms is to restore the original data from symbols and blades. The encrypted photos and videos have greater application in many fields such as medical imaging in internet connection, military communication systems and multimedia systems[1,4,15].

Basically, two available encryption algorithms types namely: symmetric and asymmetric systems; asymmetric such as Data Encryption Standard and Advanced Encryption Standard and it is normally used as an identical key for both sender and receiver in order to encrypt the text messages. As an example of systems asymmetric is considered the Elliptic Curve Cryptosystem (ECC), which uses variant keys to encrypt and decrypt[2,14]. Asymmetric encryption systems are more suitable for encrypting large amounts of data at very high speed. In this paper we design a kind of digital image encryption based on AES algorithm. Through the digital image processing to get the AES encryption standard data, encrypt the date in packet. Eventually put all the data together, reduction of encrypted image, achieve the desired encryption effect[3,9,17].

**2. Materials and Methods:** Content of different types of images the encryption algorithms had been proposed. In order to keep data secured from possible various attacks and for the data integrity, we must encrypt the digital image before the transmitting or saving[6,13]. In above Figure 1 Blocks are given which is implemented step by step for image encryption by using AES algorithm for digital image encryption and decryption[18].

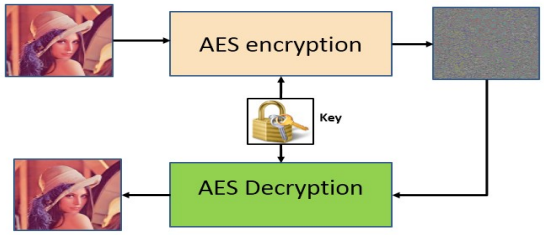


**Figure-1:** Digital Image Encryption/Decryption Process

The Advanced Encryption Standard (AES) algorithm has been applicable in mobile phones, ATMs, and smart cards. Converts plain text into cipher text can be applied by using this algorithm and may return to the original text with lot of differs about the idea of the original text[5,20]. Figure 1 shows the process of encrypted standards using a key revenge.

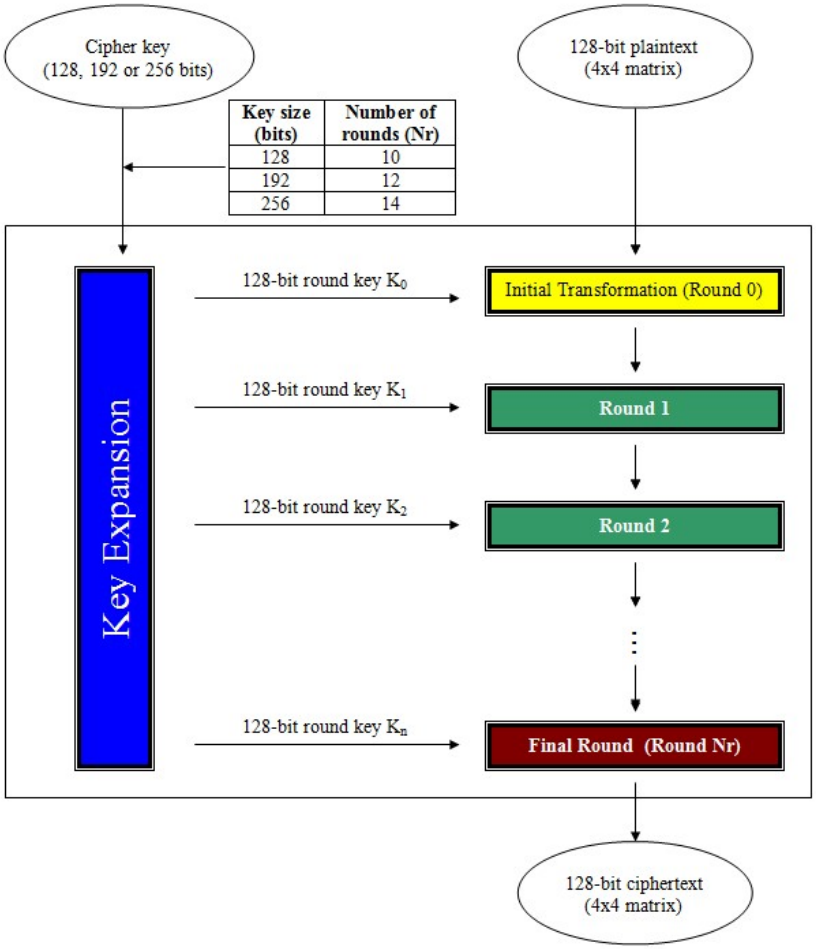


**Figure-2:** Encryption of AES Operation

The Rijndael law can identify the key sizes and block of any 32-bits complications but in range of 128-bits to 256-bits. So, the problem is to break the key to be more complex in coding. Also known as the AES as the

Rijndael has a fixed size of 128-bits and the size of the key is 128,192,256, AES algorithm might be implemented in efficient way by hardware and software[10,19]. However, AES allows data length of 128 bits, which can be divided into four blocks of the main process, these blocks are working on a set of data units and organized in a matrix of 4 × 4, which called State, data are passed by Nr rounds (Nr = 10, 12, 14) [4, 6]. These rounds are controlled by the next transformations. The Key-Block-Round combinations, which conform to this standard, are shown in Table 1.

Table 1. Key Block – Round Combinations

| | **Key Length** *(Nk words)* | **Block Size** *(Nb words)* | **Number of Rounds** *(Nr)* |
|---|---|---|---|
| **AES-128** | 4 | 4 | 10 |
| **AES-192** | 6 | 4 | 12 |
| **AES-256** | 8 | 4 | 14 |

The AES encryption algorithm in each round consists of 4 Transformations:

**2.1. Sub Bytes Transformation**. Which is used to replace bytes and to make each unit of bytes operate independently based on the use of replacement schedule. This image square is called the table and it includes a 256 number of (0-255) and matched to the other side results[8,16]. It will add value to the simple process consists of each round of words or notes, which are created by the extension of key routine[11,17].

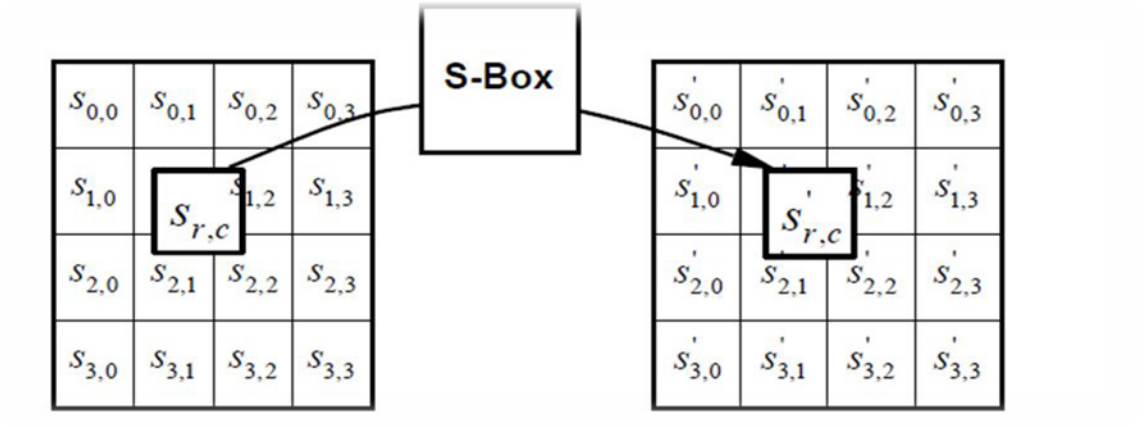


**Figure-3:** Sub Bytes Transformation

**2.2 Shift Rows Transformation.** The transformation of the ranks moves periodically in the last three rows through multiple offsets, and left first row without a change in the process of transformation, then converts each byte of the second row to one post to the left, after turns third and fourth rows by two or three functions, it is done gradually, respectively as shown on Figure 4. This simple switch turns bytes in the last three rows periodically and removes the third column from 1 byte to 3-byte values. The value of the shift entirely depends on the grade (r) that already has a significant impact on the byte shift to fewer jobs per class happen[7,15].

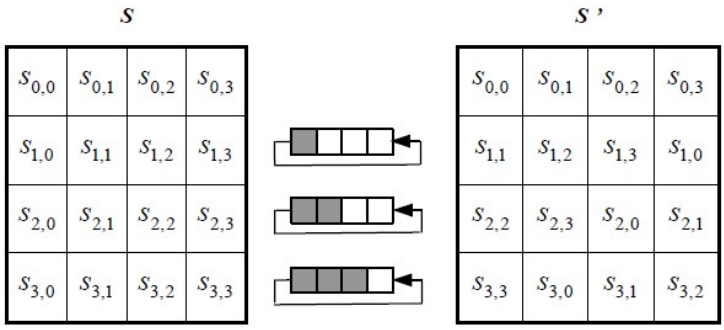


**Figure-4:** Shift Rows Transformation

**2.3 Mix Columns Transformation**. Mixed columns similar to hitting a columns matrix by each vector column with fixed matrix, it will be dealt with bytes instead of numbers, running and replaced each byte of the status column on the process as it was expected; the mixed columns are replaced 4 bytes by the 4 original ones, Figure 5.

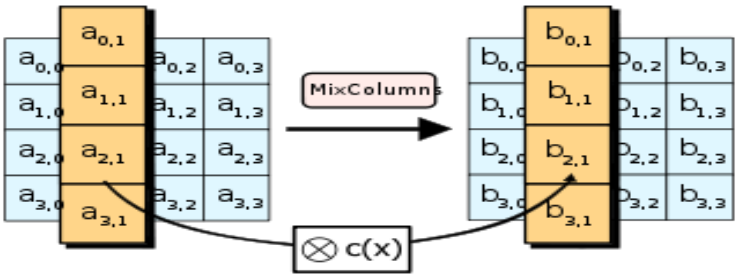


**Figure-5:** Mix Columns Transformation

**2.4 Add Round Key Transformation**. Add a Round Key to the phase shift is added key for the tour, which will result in a shift of mixed columns process by bitwise XOR operation, which is derived from Round Key. Each round of master key is using key encryption expansion algorithm or decoding, which needs 128-byte Round Key, Figure 6[12,20].

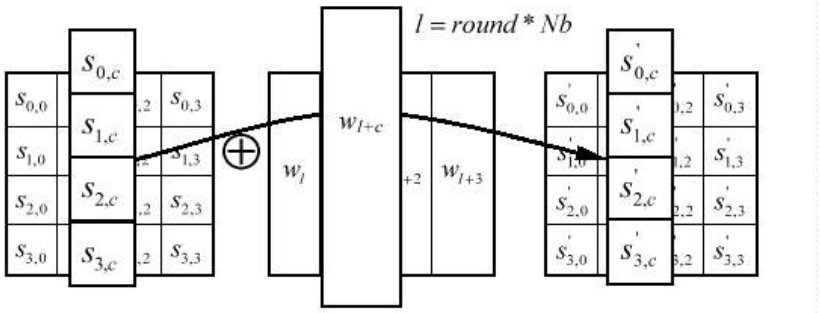


**Figure-6:** Add Round Key Transformation

The development of decoding operations that are in a reverse order, compared with arranged decoding mode and therefore at the beginning of the preliminary round, followed by nine occurrences of the regular round reverse, but ends with the addition of key stage consists regular tours inverse[3,11].

**2.5 Inv. Shift Rows Transformation**. The Inv. Shift Rows actually reversed the process for Shift rows. In this case the removal bytes will be allocated in the last three rows periodically by different numbers of bytes, [5,14] where the first row r = 0, Hence, there is no shift for three lower ranks of the converted value of transformation and it is totally depends on the row number as shown in Figure 7.

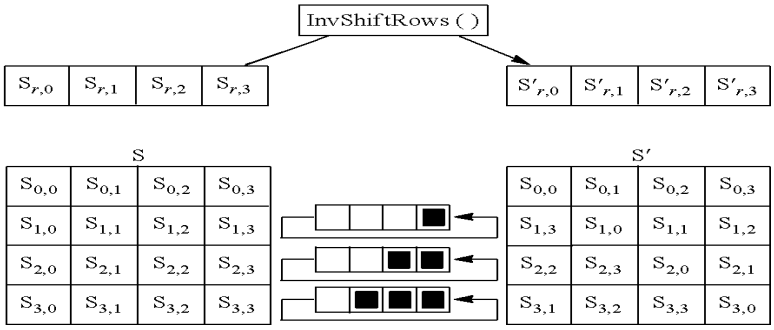


**Figure-7:** Inv. Shift Rows Transformation

**2.6 Inv. Sub bytes Transformation.** The Inv. Sub Bytes, it actually reversed the process for Sub bytes transformation. In this case, the applied inverted S box per byte are obtained by the inverse of the application which is converted to GF (28), so it will be used the Inverse Sub bytes transformation in the next Figure 8.

| x \ y | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 52 | 09 | 6a | d5 | 30 | 36 | a5 | 38 | bf | 40 | a3 | 9e | 81 | f3 | d7 | fb |
| 1 | 7c | e3 | 39 | 82 | 9b | 2f | ff | 87 | 34 | 8e | 43 | 44 | c4 | de | e9 | cb |
| 2 | 54 | 7b | 94 | 32 | a6 | c2 | 23 | 3d | ee | 4c | 95 | 0b | 42 | fa | c3 | 4e |
| 3 | 08 | 2e | a1 | 66 | 28 | d9 | 24 | b2 | 76 | 5b | a2 | 49 | 6d | 8b | d1 | 25 |
| 4 | 72 | f8 | f6 | 64 | 86 | 68 | 98 | 16 | d4 | a4 | 5c | cc | 5d | 65 | b6 | 92 |
| 5 | 6c | 70 | 48 | 50 | fd | ed | b9 | da | 5e | 15 | 46 | 57 | a7 | 8d | 9d | 84 |
| 6 | 90 | d8 | ab | 00 | 8c | bc | d3 | 0a | f7 | e4 | 58 | 05 | b8 | b3 | 45 | 06 |
| 7 | d0 | 2c | 1e | 8f | ca | 3f | 0f | 02 | c1 | af | bd | 03 | 01 | 13 | 8a | 6b |
| 8 | 3a | 91 | 11 | 41 | 4f | 67 | dc | ea | 97 | f2 | cf | ce | f0 | b4 | e6 | 73 |
| 9 | 96 | ac | 74 | 22 | e7 | ad | 35 | 85 | e2 | f9 | 37 | e8 | 1c | 75 | df | 6e |
| a | 47 | f1 | 1a | 71 | 1d | 29 | c5 | 89 | 6f | b7 | 62 | 0e | aa | 18 | be | 1b |
| b | fc | 56 | 3e | 4b | c6 | d2 | 79 | 20 | 9a | db | c0 | fe | 78 | cd | 5a | f4 |
| c | 1f | dd | a8 | 33 | 88 | 07 | c7 | 31 | b1 | 12 | 10 | 59 | 27 | 80 | ec | 5f |
| d | 60 | 51 | 7f | a9 | 19 | b5 | 4a | 0d | 2d | e5 | 7a | 9f | 93 | c9 | 9c | ef |
| e | a0 | e0 | 3b | 4d | ae | 2a | f5 | b0 | c8 | eb | bb | 3c | 83 | 53 | 99 | 61 |
| f | 17 | 2b | 04 | 7e | ba | 77 | d6 | 26 | e1 | 69 | 14 | 63 | 55 | 21 | 0c | 7d |

**Figure-8:** Inv. Sub Bytes Transformation

**2.7 Inv. Mix Columns Transformation.** The Inv. Mix Columns is the reversed of the process for Mix Columns transformation. The operations considered gradually columns one by one and treated each column as a border quad-term pillars of the cells that go beyond, where GF (28) and have hit the value of X 4 + 1 with a plurality of fixed borders a - 1 (x) = {0b} x3 + {0d} x2 + {09} x + {0e}, we can call hitting of this matrix and replace the units which followed the byte in the painting in a separate column in the following Figure 9.

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

**Figure-9:** Inv. Mix Columns Transformation

**2.9 Inverse of Add round key Transformation.** Add round key is the reversed of the process for Add round key transformation, because it depends on the application while the shape of the main tables of encryption and decryption is still the same shape, it is very important to note that there are some characteristics of the AES algorithm allow the Inverse equivalent to remain in the same previous transitions.

This is accomplished with the changes in the main table and characteristics that allow for this Inverse power parity, XOR running on the equivalent transformations and inverse differences from other transformations in Sub-bytes transfers, as well as Shift rows transfers to be light as results of sub-byte transfers immediately followed by Shift rows remittances[2,9,13].

**Key Expansion:** In AES encryption, it is well known that the secret key is kept secure for both the recipient and the sender. The AES algorithm and the secret key cannot be fixed any way even if it knows how the text cipher encryptions are designed. The AES algorithm using one of the three main sizes: AES-128, AES-196 and AES-256 key sizes respectively, these keys are secured by the core values and therefore, there is no value will make the encryption introduced from the other. Extended keys and expansion routine keys to be used in the encryption algorithm AES shown in Figure 10, this expansion routine Key cannot lead role in the all-time but when it is needed [4,8,17].

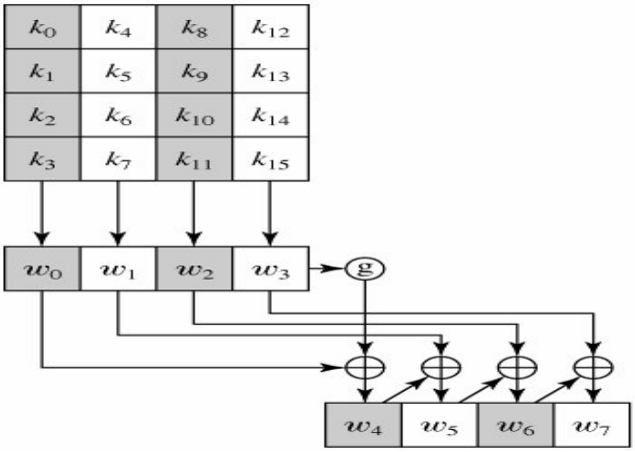


**Figure-10:** Key Expansion

**3.Results and Discussions:** The goal of this research is to study the application of Advanced Encryption Standard algorithm (AES) for secure and efficient multimedia delivery. The importance of balanced applications such as the work of iterative adjustments and the structure of the tables for decryption and optimization of the AES algorithm processes have been implemented on a 128 byte message. Encrypted and decrypted text analyses have also been studying the proposed AES algorithm efficiency.

It is also expected that AES algorithm study will have an effective role in strategic applications, because the encryption algorithm applied on hardware is also take a place in strategic communications equipment, it is safe and possible to develop this algorithm in terms of height and speed of time that have approved mainly on logistical aspects not on the technical aspects. It is possible to encrypt and decrypt by AES encryption used in many highly sensitive applications like Image encryption, ATM networks, Confidential Cooperate Documents, Personal Storage Devices, Smart Cards, Government Documents, Person Information Protection and other applications.

This research has simulated the digital image encryption and decryption in MATLAB software using interface guide. First of all we would take an image and then we obtain its corresponding matrix. Then it was necessary to encrypt the image matrix using AES crypto system. The outcomes clearly illustrate the original images (color) and encrypted images. Note that the image after decryption is the same as it was before in the original view (figure 11 and figure 12).

The performance of proposed method is evaluated through some factors (visible scene, histogram distribution). Experimental results clearly show that the proposed method has good result with very low encryption time in comparison with other two methods.

**3.1 Visible scene and histogram:** For a test lena.bmp image, histogram distribution is shown in Figure 13. Note that the histogram uniformly distributed, this means the proposed method is strong against. Entropy is calculated by the following equation.

$$H(x) = \sum_{i=1}^{K} P(x_i)\, log_2 \frac{1}{P(x_i)} = -\sum_{i=1}^{K} P(x_i)\, log_2\, P(x_i) \qquad (1)$$

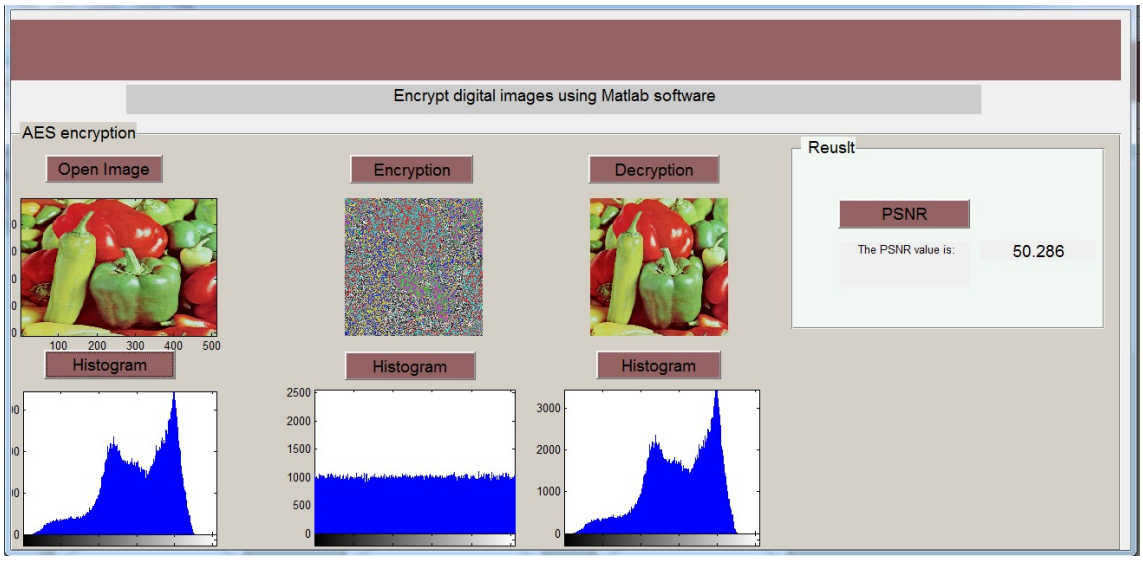


**Figure-11:** Digital Image Encryption Using MATLAB Interface

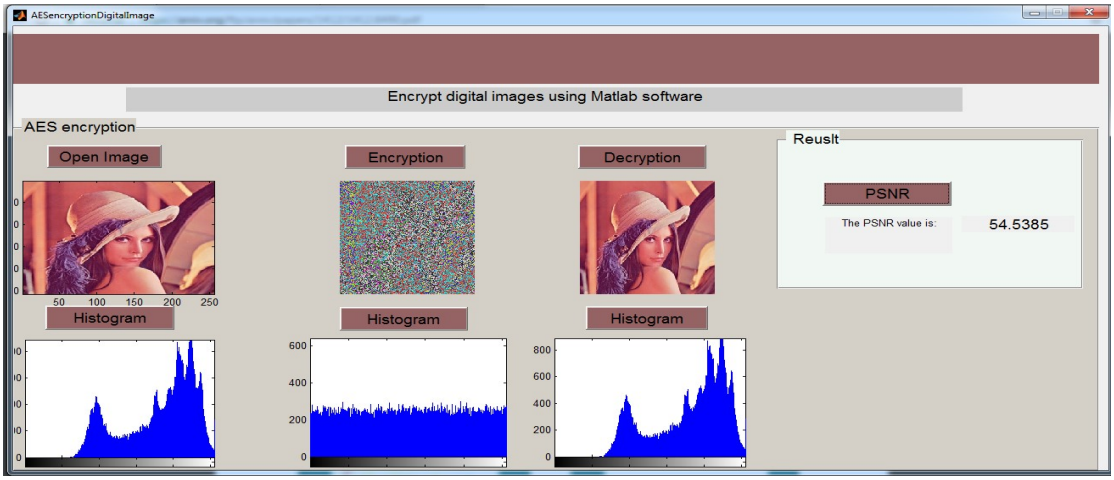


**Figure-12:** Simulation for Digital Image Encryption

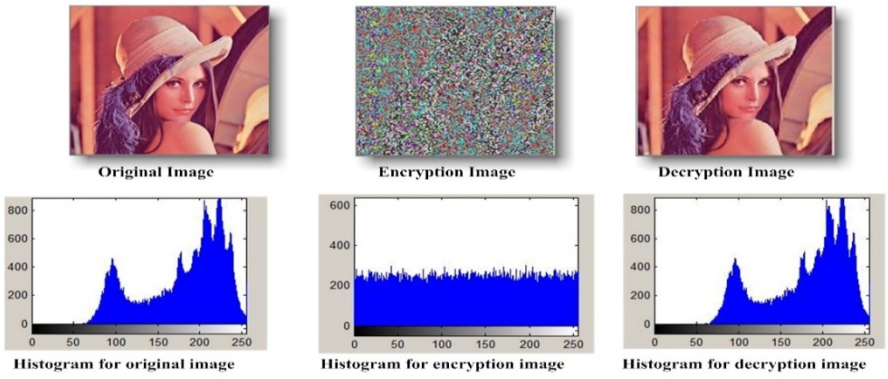


**Figure-13:** AES Crypto System Test and Result

The solution of the histogram of the original image; the encrypted image and the decrypted image are shown as Fig.13.

From the histogram we can see that there is obvious change between the encrypted image histogram and the original image histogram. There are great changes from the distribution of original image pixels and encrypted image pixels. This result approve that the AES algorithm has good effect for image encryption. It also suggests that this kind of algorithm had a high performance for security. In the process of the image transmission it will be not susceptible to tampering or eavesdropping.

**4.2 Peak signal to noise ratio:** It tells about the quality of the image. It is estimate of peak error if we have high value of PSNR then the quality of image is high. PSNR can be calculated as:

$$PSNR = 10\log_{10}\frac{(255)^2}{MSE} \qquad (2)$$

When MSE (Mean Square Error) is a measure of the deformity in the reformed image, it can be seen in equation 3.

$$MSE = \frac{1}{MSE}\sum_{i=1}^{U}\sum_{j=1}^{V}[X(i,j) - Y(i,j)]^2 \qquad (3)$$

Simulation results for color image encryption process shows that AES algorithm gives a better PSNR result. Table 2 present a performance result for our crypto system.

Table 2. AES Crypto System Evaluation

| Test image | PSNR |
|---|---|
| Lena | 54.53 |
| baboon | 46.20 |
| Pepper | 50.28 |
| Cat | 54.59 |
| monarch | 55.25 |
| serrano | 47.09 |

**Conclusion.** This paper presents the method that uses the AES algorithm with the key control to encrypt the digital image. This method incorporates a variety of characteristics. As the MATLAB has powerful numerical calculation function, especially for image processing calculations. Matrix processing is the basic unit infrastructure of the AES algorithms, the implementation of the image encryption and decryption based on AES algorithm in the MATLAB environment become easy. From the above simulation results and analysis, coupled with the histogram, key sensitivity analysis and PSNR performances, this method can approve very good effect on image encryption. And the decryption essence has the same structure with the encryption, so it can easily restore the original image. Due to the AES algorithm is easy to implement in software. So we have reason to believe that use this method to encrypt the image will have a very good prospect in the future.

**Acknowledgement.** We thank King Abdulaziz University and Arab Open University for supporting carrying out this work.

**REFERENCES**

[1] Einstein, A., B. Podolsky, and N. Rosen, 1935, "Can quantum-mechanical description of physical reality be considered complete?", Phys. Rev. 47, 777-780.

[2] Kamali, S. H., Shakerian, R., Hedayati, M., & Rahmani, M. (2010, August). A new modified version of advanced encryption standard based algorithm for image encryption. In *Electronics and Information Engineering (ICEIE), 2010 International Conference On* (Vol. 1, pp. V1-141). IEEE.

[3] Amador, J. J., & Green, R. W. (2005). Symmetric-key block cipher for image and text cryptography. *International Journal of Imaging Systems and Technology*, *15*(3), 178-188.

[4] Maniccam, S. S., & Bourbakis, N. G. (2004). Image and video encryption using SCAN patterns. *Pattern Recognition*, *37*(4), 725-737.

[5] Maniccam, S. S., & Bourbakis, N. G. (2004). Image and video encryption using SCAN patterns. *Pattern Recognition*, *37*(4), 725-737.
[6] Zeghid, M., Machhout, M., Khriji, L., Baganne, A., & Tourki, R. (2007). A modified AES based algorithm for image encryption. *International Journal of Computer Science and Engineering*, *1*(1), 70-75.
[7] Hoang, T. (2012, February). An efficient FPGA implementation of the Advanced Encryption Standard algorithm. In *Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2012 IEEE RIVF International Conference on* (pp. 1-4). IEEE.
[8] Rachh, R. R., Anami, B. S., & Mohan, P. A. (2009, January). Efficient implementations of S-box and inverse S-box for AES algorithm. In *TENCON 2009-2009 IEEE Region 10 Conference*(pp. 1-6). IEEE.
[9] Jun, Y., Jun, D., Na, L., & Yixiong, G. (2010, March). FPGA-based design and implementation of reduced AES algorithm. In *Challenges in Environmental Science and Computer Engineering (CESCE), 2010 International Conference on* (Vol. 2, pp. 67-70). IEEE.
[10] Lu, C. F., Kao, Y. S., Chiang, H. L., & Yang, C. H. (2003, October). Fast implementation of AES cryptographic algorithms in smart cards. In *Security Technology, 2003. Proceedings. IEEE 37th Annual 2003 International Carnahan Conference on* (pp. 573-579). IEEE.
[11] Su, C. P., Horng, C. L., Huang, C. T., & Wu, C. W. (2005, January). A configurable AES processor for enhanced security. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference* (pp. 361-366). ACM.
[12] Kaur, S., & Vig, R. (2007, December). Efficient implementation of aes algorithm in fpga device. In *Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on* (Vol. 2, pp. 179-187). IEEE.
[13] Reddy, M. S., & Babu, M. Y. A. (2013). Evaluation Of Microblaze and Implementation Of AES Algorithm using Spartan-3E. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, *2*(7), 3341-3347.
[14] AlRababah, A. A. (2017). A New Model of Information Systems Efficiency based on Key Performance Indicator (KPI). *management*, *4*, 8.
[15] Al-Rababah, A. A., & Biswas, R. (2008). Rough vague sets in an approximation space.
[16] AlRABABAH, A. A. (2017). Implementation of Software Systems Packages in Visual Internal Structures. Journal of Theoretical and Applied Information Technology, 95(19).
[17] Al-Rababah, A. A., & Al-Rababah, M. A. (2007). Functional Activity Based Comparison Study for Neural Network Application. *IJCSNS*, *7*(1), 153.
[18] Al Ofeishat, H. A., & Al-Rababah, A. A. (2009). Real-time programming platforms in the mainstream environments. *IJCSNS*, *9*(1), 197.
[19] Al-rababah, A. A., & Al-rababah, M. A. (2007). Module Management Tool in Software Development Organizations 1.
[20] Al-Rababah Ahmad, A. (2009). UML–Models Implementations in Software Engineering System Equipments Representations. *International Journal of Soft Computing Applications*, (4), 25-34.