Journal of Engineering and Applied Sciences 13 (Special Issue 6): 5230-5237, 2018
ISSN: 1816-949X

Design and Implementation of a Secure System Based on Compression and Encryption Techniques

[1]Baheja Shukur, [2]Intissar hamid and [1]Sawsan Hadi
[1]Department of Information Network, Faculty of Information Technology,
[2]Department of Electrical Engineering, Faculty of Engineering,
University of Babylon, 51001 Hillah, Babylon, Iraq

Abstract: With the fast growth in multimedia and network technologies, the image compression and encryption become a very important part in the communication and storage of images. Image compression methods proposed by the authors have no concern of security. Similarly, image encryption methods proposed by the researchers have no concern of image size. This research presents a way to apply image encryption with compression using a secret key. In compression stage, Five Modulus Method (FMM) and FMM protocol apply to each band of image (R, G, B). The encryption stage consists of two-part diffusion and confusion operations. In diffusion part on all the bits of pixels in the image a sequential XOR operation is done and confusion part makes a circular rotate right of these bits. On the other hand, the suggested system uses geffe generator to generate an initial secret key. To evaluate our suggested system number of performance measurements are used like entropy, correlation, histogram, PSNR, MSE, AD, SC, MD, NAE and compression ratio to show the power of the suggested system. The suggested system is implemented in Vb6 language.

Key words: Image compression, FMM protocol, FMM, image encryption, diffusion, confusion, rotation, XOR, entropy, histogram, correlation coefficient, image quality metrics, Geffe generator

INTRODUCTION

Encryption an invertible conversion of information for hiding from of any extraneous persons, providing, While, authorized users are able outlet for information. Mainly, encryption task is the confidentiality of transmitting information. An important feature of any encryption algorithm is to use the key that approves the choice of conversion of the plurality of potential for this algorithm. Users are authorized if they have some authentic key (Zhou *et al*., 2016; Schneier, 1996). All the complexity and indeed, the encryption objective consists in how this process is realized. In general, encryption composed of two parts encryption and decryption. Encryption is used to hide information from unauthorized users in transmission or storage, prevent change of information in transit or storage and authenticate the source of information and to prevent of failure of the sender information by the fact that the data has been sent to them (Poorani and Manju, 2016).

Image compression is the applying data compression algorithm to the images that are stored in digital form. As a result of the compression, image size decreases because it reduces the image transmission time on the network and saves storage space (Jagadeesh and Rani, 2013; Uhl and Pommer, 2005). Image compression is divided into lossy compression quality and lossless compression. Lossless compression is often preferable to artificially construct images such as graphics programs, icons or for special occasions for example if the images are intended for further processing of image recognition algorithms. Lossy compression algorithms with an increase in compression ratio as a rule give rise to clearly visible to the human eye artifacts (Zhang, 2011). Algorithms for lossless compression are RLE is used in formats PCX as the primary method and in the formats BMP, TGA, TIFF as one of the available, LZW used in GIF format, LZ-Huffman used in PNG format. Algorithms Lossy compression is JPEG, on mobile platforms used image transfer to palletized format, JPEG 2000 of a fractal compression algorithm, DXTC - texture compression implemented in the graphics API DirectX and supported at the hardware level by modern graphics cards (Xiaoyong *et al*., 2017; Hoggs and Tanis, 2009; Johnson *et al*., 2004).

Corresponding Author: Baheja Shukur, Department of Information Network, Faculty of Information Technology, University of Babylon, 51001 Hillah, Babylon, Iraq

FMM technique: FMM it means five modulus method was presented by Jassim and Qassim (2012). The basic concept of this technique is to transform each pixel into multiples the number 5. This transformation neglects the portion signal which not seen on the receiving system (HVS). Since, adjacent pixels in the image matrix are correlated, so, the one of the very important tasks is determining minimally correlated image representation. The basic concept of image compression that the adjacent pixels usually has the same direct adjacent.

In FFM method the image is divided into 8×8 blocks, then in each block each pixel can be converted into multiples the number 5. The efficiency of this conversion will not be seen by the HVS (Jassim and Qassim, 2012). Thus, the value of each pixel multiples of 5, i.e., 0, 5, 10, 15, 20, ..., 255. The algorithm of FMM formulated as:

Algorithm 1; FMM formulated:

if the value of pixelmod 5 = 4
$Pixel_{after} = pixel_{before} + 1$
Else if the value of pixelmod 5 = 3
$Pixel_{after} = pixel_{before} + 2$
Else if the value of pixelmod 5 = 2
$Pixel_{after} = pixel_{before} - 2$
Else if the value of pixelmod 5 = 1
$Pixel_{after} = pixel_{before} - 1$

The conversion accurately shown in Table 1 also, we can observe that when divided the new values of pixels by 5 remainder are zero. Hence, the result 52 numbers are multiples of 5 between 0-255 which are (0.5, 10.15, 20, ..., 255). Consequently, again if we divide the result numbers by 5, we get numbers from 0-51. To illustrate FMM method we take an example, Table 2 is precisely 8×8 blocks were chosen from any one of R, G or B arrays in any BMP image.

To determine the variance between pixel can be computed the standard deviation as a (12.8285). That means each pixel is converted into multiple of five after applying FMM, a new block after applying FMM can be observed in Table 3.

After the new block dividing by 5, we obtain Table 4. In Table 4, we calculate the standard deviation as (2.572751) to measure the variance between the pixels. We can observe after the conversion using FMM that the variance between pixels becomes less. Where the standard deviation in the transformed block is lower than in the original block. Then we fined the minimum number in Table 4 which is 42 and then subtracted it from each pixel in this figur, after this we obtain Table 5.

As seen in Table 5, the number 8 is the maximum number and (1000) it is the binary code of 8, that means the length of each number in the block is 4 bits. So, the block length is 64×4 = 256 and each number in original block has 8 bits so the length of original block is 64×8 = 512. Hence, the ratio compression is 512/256 = 2.

Table 1: The values of the pixels before and after using FMM

Before	After	Before	After
0	0	100	100
1	0	101	100
2	0	102	100
3	5	103	105
4	5	110	110
8	10	111	110
9	10	112	110
10	10	113	115
13	15	114	115

Table 2: 8×8 block from any one of R, Y or G arrays

Pixel (row, colum)	Co l1	Co l2	Co l3	Co l4	Co l5	Co l6	Co l7	Co l8
Row 1	221	232	231	242	246	247	251	250
Row 2	220	227	231	236	242	241	250	251
Row 3	221	215	221	232	240	247	251	251
Row 4	217	216	216	225	237	241	245	247
Row 5	216	221	217	222	231	235	242	247
Row 6	220	216	222	215	227	231	242	247
Row 7	216	216	211	216	222	227	237	247
Row 8	217	216	211	216	217	222	237	235

Table 3: A new block after applying FMM

Pixel (row, Colum)	Co l1	Co l2	Co l3	Co l4	Co l5	Co l6	Co l7	Co l8
Row 1	220	230	230	240	245	245	250	250
Row 2	220	225	230	235	240	245	250	250
Row 3	220	215	220	230	240	245	250	250
Row 4	215	215	215	225	235	240	245	245
Row 5	215	220	215	220	230	235	240	245
Row 6	220	215	220	215	225	230	240	245
Row 7	215	215	210	215	220	225	235	245
Row 8	215	215	210	215	215	220	235	235

Table 4: Dividing new block by 5

Pixel (row, colum)	Co l1	Co l2	Co l3	Co l4	Co l5	Co l6	Co l7	Co l8
Row 1	44	46	46	48	49	49	50	50
Row 2	44	45	46	47	48	49	50	50
Row 3	44	43	44	46	48	49	50	50
Row 4	43	43	43	45	47	48	49	49
Row 5	43	44	43	44	46	47	48	49
Row 6	44	43	44	43	45	46	48	49
Row 7	43	43	42	43	44	45	47	49
Row 8	43	43	42	43	43	44	47	47

Table 5: Subtracting 42 from each pixel of the Table 4

Pixel (row, colum)	Co l1	Co l2	Co l3	Co l4	Co l5	Co l6	Co l7	Co l8
Row 1	2	4	4	6	7	7	8	8
Row 2	2	3	4	5	6	7	8	8
Row 3	2	1	2	4	6	7	8	8
Row 4	1	1	1	3	5	6	7	7
Row 5	1	2	1	2	4	5	6	7
Row 6	2	1	2	1	3	4	6	7
Row 7	1	1	0	1	2	3	5	7
Row 8	1	1	0	1	1	2	5	5

Table 6: The bit representation of each number between 0-51

Number	Bit stream	Number	Bit stream
0	000000	32	100000
1	000001	33	100001
2	000010	34	100010
3	000011	35	100011
4	000100	36	100100
5	000101	37	100101
6	000110	38	100110
7	000111	39	100111
8	001000	40	101000
9	001001	41	101001
10	001010	42	101010
11	001011	43	101011
12	001100	44	101100
13	001101	45	101101
14	001110	46	101110
15	001111	47	101111
16	010000	48	110000
17	010001	49	110001
18	010010	50	110010
19	010011	51	110011

FMM bit storage: Clearly, we needed 8 bits location to store each number between 0-255. Now, after applying FMM we obtain a number between 0-51. The bit representation of each number is shown in Table 6.

Clearly as shown from Table 6, we can observe that the length of each number between 0-51 is 6 bits where the number 51 has 6 bits. Which means that the bit stream of original block is more than a new block by 2 bits.

MATERIALS AND METHODS

Five modulus method protocol: Obviously, the length of each number from 0-255 is 8 bits where the number 255 has 8 bits. After using FMM method we get numbers from 0-51 and the length of each number become 6 bits where the number 51 has 6 bits. According that the FMM protocol can be created as the protocol of computer network. So, as FFM stream protocol consists of from header and stream (Loussert *et al*., 2008). The header contains6 bits for the minimum value of new block after conversion and one bit called repetition bit where if each pixel of the new block has the same value the repetition bit will be one otherwise will be zero. So, the repetition bit is used to reduce the storage for each new block when the whole block contains the same value. The purpose of FMM method is to reduce the variance of the same block. The repetition bit at the most time will be one because of The convergence between the value of the block. The third part of the header is 6 bits for maximum value of the resulting after subtracting minimum value from each pixel of a new block. In stream the number of bits are max

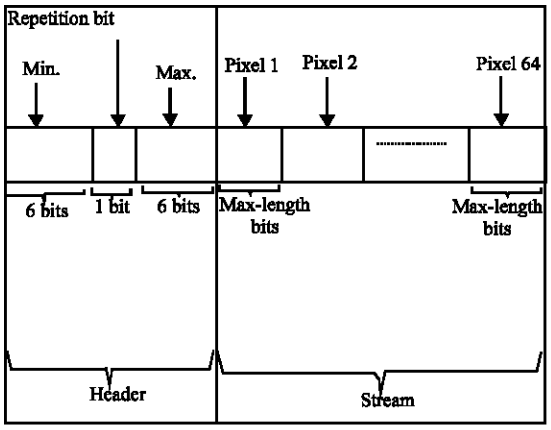


Fig. 1: FMM protocal

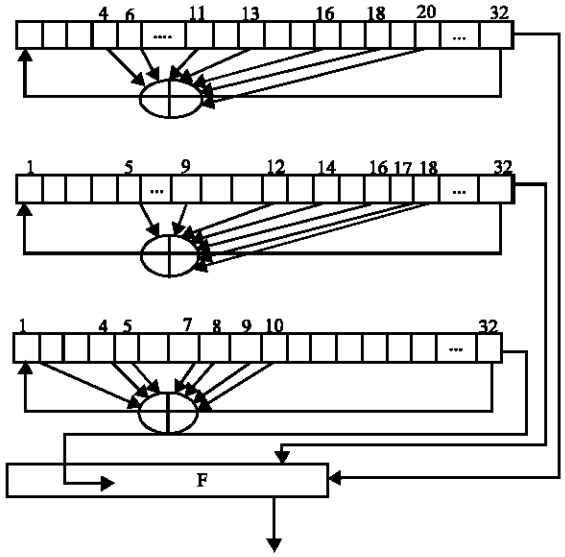


Fig. 2: The encryption key generator

length bits multiplying by 64 which means each pixel has max length bits, max length bits are the length of maximum value after subtracting (Hassan and Younis, 2013; Jakimoski and Subbalakshmi, 2007). So, the stream bits of pixels from pixel (1) to pixel (64) Fig. 1, the FMM protocol.

Encryption key generator: This generator consists of three Linear Feedback Registers (LFSR) (LFSR1, LFSR2,-LFSR3). The output of these generators combines by the Boolean function that given by:

$$F(X1, X2, X3) = (X1.X2) \otimes (X1.X3)$$

The architecture of this key generator can be shown in Fig. 2:

LFSR primitive polynomial:

$$1+x^4+x^6+x^{11}+x^{13}+x^{16}+x^{18}+x^{20}+x^{32}$$

LFSR2 primitive polynomial:

$$1+x+x^5+x^7+x^9+x^{12}+x^{14}+x^{16}+x^{17}+x^{18}+x^{32}$$

LFSR3 primitive polynomial:

$$1+x^2+x^4+x^5+x^7+x^8+x^9+x^{10}+x^{32}$$

Performance measures: To judge the quality of the reconstructed images compared with the original ones and evaluate the performance of the suggested system, it is needed to use some image quality parameters. Some of standard metrics have been established including the most usable metric such as: Histogram, Entropy, correlation, PSNR (PN) which are explained in Eq. 3 and 4. O_{xy} and R_{xy} indicate the pixel value of the original and reconstructed image and PP (v) is representing the probability of frequency of a pixel with gray level valuable (v) (Jawad and Fawad, 2013; David, 2007; Lu and Hao, 2011). Most decompression systems are intended to maximize the PSNR and minimize the MSE:

$$\text{Entropy} = -\sum_{v=0}^{N-1} PP(v)*\log_2 PP(v) \quad (1)$$

$$\text{Correlation} = \frac{\sum_{x=0}^{H-1}\sum_{y=0}^{W-1}(O_{xy}-\bar{O})(R_{xy}-\bar{R})}{\sqrt{\sum_{v=0}^{X-1}\sum_{w=0}^{Y-1}(O_{xy}-\bar{O})^2(R_{xy}-\bar{R})^2}} \quad (2)$$

$$\text{MSR} = \sqrt{\frac{1}{H^2}\sum_{x=0}^{H-1}\sum_{y=0}^{W-1}[O_{xy}-R_{xy}]^2} \quad (3)$$

$$\text{PN} = 10\log_{10}\frac{(LL-1)^2}{\frac{1}{(H\times W)}\times\sum_{x=0}^{H-1}\sum_{y=0}^{W-1}[O_{xy}-R_{xy}]^2} \quad (4)$$

Also, the Compression Ratio (CR) may be defined as the ratio between the reconstructed image to the original image (Samson and Sastry, 2012):

$$\text{CR} = \text{Size of Uncompressed file}/\text{Size of compressed file} \quad (5)$$

Another image quality metrics are used for assessing the power of the suggested system like Normalized Absolute Error (NAE), Structural Content (SC), Average Difference (AD) and Maximum Difference (MD) (Ganesan and Bhavani, 2013; Ghodake and Mendgudle, 2013):

$$\text{AD} = \sum_{x=1}^{H}\sum_{y=1}^{W}(O_{xy}-R_{xy})/HW \quad (6)$$

$$\text{SC} = \sum_{x=1}^{H}\sum_{y=1}^{W}(O_{xy}^2/\sum_{x=1}^{H}\sum_{y=1}^{W}(R_{xy}^2 \quad (7)$$

$$\text{MD} = \text{Max}(||O_{xy}-R_{xy}) \quad (8)$$

$$\text{NAE} = \sum_{x=2}^{H}\sum_{y=2}^{W}(O_{xy}-R_{xy})/\sum_{x=2}^{H}\sum_{y=2}^{W}|O_{xy}| \quad (9)$$

Structural design of the suggested system: The suggested system can be seen in Fig. 3:

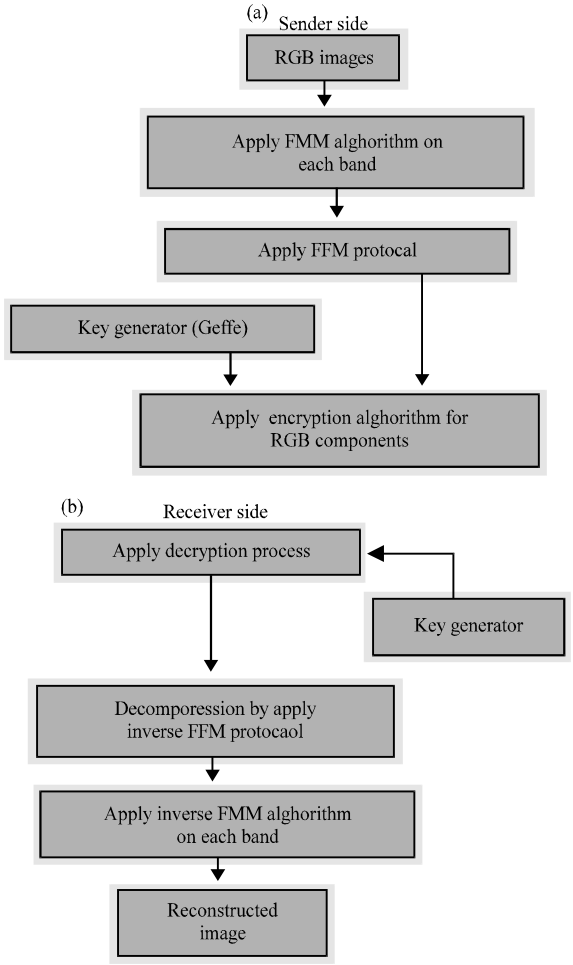


Fig. 3: Structural design of the suggested system: a) Sender side and b) Receiver side

Sender side

Compression part:

- The source image is converted into its RGB components
- Applied the FMM technique on each band of source image
- FMM protocol is applied on o each band

Encryption part:

- Generate a secret key of length k (number of bits) using geffe generator
- Create a list of binary vectors (vector list (no of vectors) by splitting the source image into a set of vectors. Each vector consists of k number of bits and their indices of vector list (0, ..., no of vectors-1)
- Set temp vector = vector list (0) and set vector list (0) = secret key
- Diffusion operation: XOR the bits of each vector with the bits of its neighbor vectors. The step applies to all vectors in vector list
- Confusion operation: circular rotate right the bits of each vector number of times equivalent to the number of 0's bits in It. The step is repeated on all vector in vector list
- Repeat diffusion and confusion operation number of times equivalent to the number of 1s bits in secret key
- Set vector list (0) = temp vector
- Repeat all above steps for each band of the image
- Restore the vector lists in encrypted image

Receiver side

Decryption part:

- Use the same secret key that is generated in encryption part
- Create a list of binary vectors vector list (no of vectors) by splitting the encrypted image into a set of vectors. Each vector consists of k number of bits and the indices of vectors in VectorList (0, ..., no of vectors-1)
- Set temp vector = vector list (0) and set vector list (0) = secret key
- Inv-confusion operation: circular rotate left the bits of each vector number of times equivalent to the number of 0s bits in it. The step is repeated on all vector in vector list
- Inv-diffusion operation: XOR the bits of each vector with the bits of its neighbor vector. The step applies to all vectors in vector list start from vector list (no of vectors-2) to vector list (0)
- Repeat diffusion and confusion operation number of times equivalent to the number of 1's bits in secret key
- Set vector list (0) = temp vector
- Repeat all above steps for each band of the image
- Restore the vector lists in decrypted imag
- The inverse FMM technique applies on each band of the image
- Reconstructed the source image by combining R, G, B

Decompression part: Decompression the image by applying inverse FMM protocol on o each band.

RESULTS AND DISCUSSION

Experimental result for suggested system: The suggested system was carried out on three images. In Tables 7, showed the compression ratio, Table 8, explain The Entropy of the original image, image after applying FMM algorithm and after subtracting the Minimum Value (MV) while Table 9, studied the effect of using image quality measurements for the images in Table 10, showed the original image, the image after applying FMM, the image after subtracting the minimum value and their histograms whereas Fig. 4, calculated the correlationb etween original image, reconstructed image as shown.

Table 7: Compression ratio for three images

Image	R (%)	G (%)	B (%)
1	60	59	59
2	63	61	62
3	77	80	79

Table 8: The Entropy of the original image, image after applying FMM and after subtracting the Minimum Value (MV)

Images	Original	After FMM	After subtracting MV
Image 1			
R	7.5702	5.2865	3.7039
G	7.6201	5.3611	3.7179
B	7.5519	5.2757	3.666
Image 2			
R	6.2578	4.2503	3.6417
G	7.3609	5.0647	3.8415
B	6.9256	4.6813	3.8421
Image 3			
R	7.3107	5.2862	2.3772
G	6.2834	4.2153	2.1557
B	5.9348	3.7865	2.3595

Table 9: Image quality metrics values

Iamges	MSE	PNSR	AD	SC	MD	NAE
Image 1						
R	2.0155	45.087	0.0029	1.0000	2	0.0076
G	1.9901	45.142	0.0052	0.9997	2	0.0100
B	2.0257	45.065	0.0042	0.9997	2	0.0115
Image 2						
R	1.7686	45.6546	0.0842	0.9995	2	0.0190
G	1.9893	45.1438	-0.0168	0.9997	2	0.0088
B	1.9701	45.1859	0.0028	1.0000	2	0.0063
Image 3						
R	2.0238	45.0691	-0.0018	1.0005	2	0.0076
G	1.9268	45.2824	0.1312	1.0007	2	0.0056
B	2.0228	45.0712	0.0070	0.9999	2	0.0075

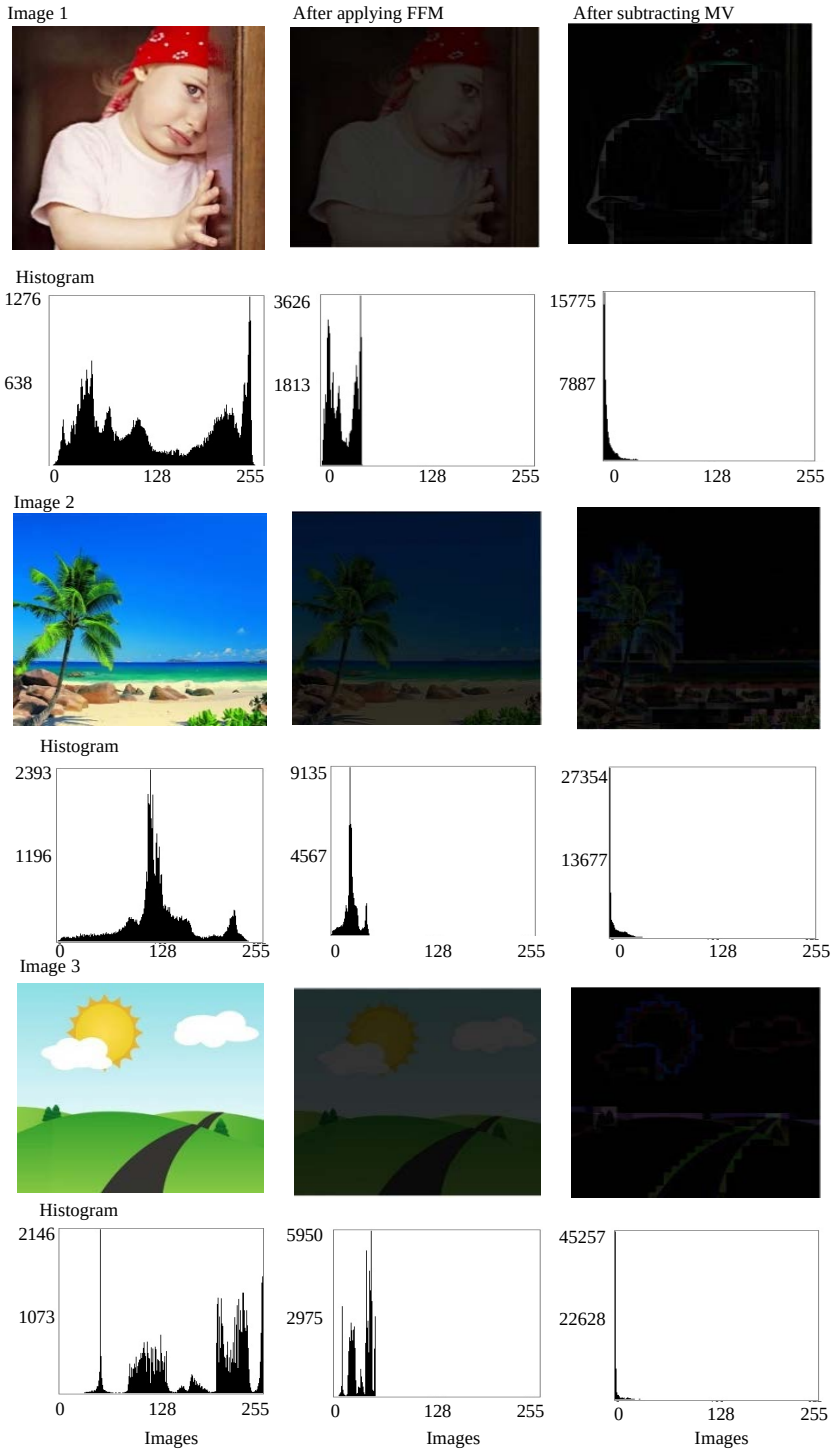
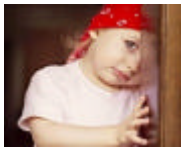
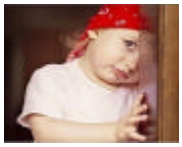


Fig. 4: The original image, the image after applying FMM, the image after subtracting the Minimum Value (MV) and their histograms

As shown in Table 8, the entropy values decreased after applying FMM and after subtracting the minimum value that means the number of pixels used to represent the image decreased. Also, we have seen that the values of quality metrics and correlation coefficient have not very high difference between original and reconstructed image. Therefore, one could accept this fair difference by taking into consideration compression ratio.

Table 10: Original image, reconst-ructed image and correlation coefficient between them

Images (1-3)		Correlation coefficient		
Reconst-ructed image	Original image	R	G	B
		0.9998	0.9999	0.9998
		0.9998	0.9996	0.9999
		0.9999	0.9996	0.9997

CONCLUSION

Using compression an encrypted image to reduce size of image and increase the secure transmission at the same time. Based on the experimental results, we conclude the suggested system has the best performance with lower MSE, higher PSNR and good security features. The suggested system gives good diffusion and confusion features by combining the two Boolean operations rotation and XOR. The compression ratio is increased when there is a higher correlation between pixels of the image. Using a big number of bits in the secret key gives the system more powerful and avoid the repetition in the encryption process. The variations between PSNR for the original image and reconstructed image are reasonable.

REFERENCES

David, S., 2007. Data Compression: The Complete Reference. 4th Edn., Springer, New York, USA., ISBN-10:1-84628-602-6, Pages: 1092.

Ganesan, P. and R. Bhavani, 2013. A high secure and robust image steganography using dual wavelet and blending model. J. Comput. Sci., 9: 277-284.

Ghodake, A.P. and S. Mendgudle, 2013. Security and privacy of image by encryption, lossy compression and iterative reconstruction. Intl. J. Comput. Appl., 62: 0975-8887.

Hassan, N.S. and H.A. Younis, 2013. Approach for partial encryption of compressed images. J. Babylon Univ. Pure Appl. Sci., 21: 775-784.

Hoggs, R.V. and E.A. Tanis, 2009. Probability and Statistical Inference. Vol.8, Macmillan Publishing Co., New York City, New York, USA.,.

Jagadeesh, S.V.V.D. and T.S. Rani, 2013. An effective approach of compressing encrypted. Intl. J. Comput. Commun. Technol., 2: 1108-1111.

Jakimoski, G. and K.P. Subbalakshmi, 2007. Security of compressing encrypted sources. Proceedings of the Forty-First Asilomar Conference on Signals, Systems and Computers ACSSC, November 4-7, 2007, IEEE, Pacific Grove, California, ISBN:978-1-4244-2109-1, pp: 901-903.

Jassim, F.A. and H.E. Qassim, 2012. Five modulus method for image compression. Intl. J. Signal Image Process., 3: 1-10.

Jawad, A. and A. Fawad, 2013. Efficiency analysis and security evaluation of image encryption schemes. Int. J. Video Image Process. Network Secur., 12: 18-31.

Johnson, M., P. Ishwar, V. Prabhakaran, D. Schonberg and K. Ramchandran, 2004. On compressing encrypted data. IEEE Trans. Signal Process., 52: 2992-3006.

Loussert, A., A. Alfalou, R.E. Sawda and A. Alkholidi, 2008. Enhanced System for image's compression and encryption by addition of biometric characteristics. Intl. J. Software Eng. Appl., 2: 111-118.

Lu, H. and X. Xiao, 2011. A novel color image encryption algorithm based on chaotic maps. Adv. Inf. Sci. Serv. Sci., 3: 28-35.

Poorani, B.A. and M.E. Manju, 2016. Secure image transformation using encryption and compression techniques. Intl. J. Adv. Res. Sci. Eng. Technol., 3: 2079-2084.

Samson, C. and V. Sastry, 2012. A novel image encryption supported by compression using multilevel wavelet transform. Intl. J. Adv. Comput. Sci. Appl., 3: 178-183.

Schneier, B., 1996. Applied Cryptography. Vol. 2, Wiley, New York, USA.,.

Uhl, A. and A. Pommer, 2005. Image and Video Encryption: From Digital Rights Management to Secured Personal Communication. Vol. 1, Springer, New York, USA., ISBN:0-387-23402-0, Pages: 125.

Xiaoyong, J., B. Sen, Z. Guibin and Y. Bing, 2017. Image encryption and compression based on the generalized knight's tour, discrete cosine transform and chaotic maps. Intl. J. Multimedia Tools Appl., 76: 12965-12979.

Zhang, X., 2011. Lossy compression and iterative reconstruction for encrypted image. IEEE. Trans. Inf. Forensics Secur., 6: 53-58.

Zhou, N., S. Pan, S. Cheng and Z. Zhou, 2016. Image compression-encryption scheme based on hyper-chaotic system and 2D compressive sensing. Opt. Laser Technol., 82: 121-133.