Image Encryption Using Block-Based Transformation Algorithm

Mohammad Ali Bani Younes and Aman Jantan

***Abstract*—Encryption is used to securely transmit data in open networks. Each type of data has its own features, therefore different techniques should be used to protect confidential image data from unauthorized access. Most of the available encryption algorithms are mainly used for textual data and may not be suitable for multimedia data such as images. In this paper, we introduce a block-based transformation algorithm based on the combination of image transformation and a well known encryption and decryption algorithm called Blowfish. The original image was divided into blocks, which were rearranged into a transformed image using a transformation algorithm presented here, and then the transformed image was encrypted using the Blowfish algorithm. The results showed that the correlation between image elements was significantly decreased by using the proposed technique. The results also show that increasing the number of blocks by using smaller block sizes resulted in a lower correlation and higher entropy.**

***Index Terms*—Image correlation, Image encryption, Image entropy, Permutation.**

I. Introduction

The rapid growth of computer networks allowed large files, such as digital images, to be easily transmitted over the internet [1]. Data encryption is widely used to ensure security however, most of the available encryption algorithms are used for text data. Due to large data size and real time constrains, algorithms that are good for textual data may not be suitable for multimedia data [2]-[4].

In most of the natural images, the values of the neighboring pixels are strongly correlated (i.e. the value of any given pixel can be reasonably predicted from the values of its neighbors) [5]-[7]. In order to dissipate the high correlation among pixels and increase the entropy value, we propose a transformation algorithm that divides the image into blocks and then shuffles their positions before it passes them to the Blowfish encryption algorithm. By using the correlation and entropy as a measure of security, this process results in a lower correlation and a higher entropy value when compared to using the Blowfish algorithm alone, and thus improving the security level of the encrypted images. There are two main keys to increase the entropy; the variable secret key of the transformation process and the variable secret key of the Blowfish algorithm. The variable secret key of the transformation process determines the seed, which is used to build the secret transformation table with a variable number of blocks. If the key is changed, another seed will be generated, and then a different secret transformation table is obtained.

The variable secret key of the Blowfish algorithm is used to encrypt the transformed image. This encryption process decreases the mutual information among the encrypted image variables (i.e. high contrast) and thus increasing the entropy value. In this paper we propose a block-based transformation algorithm in order to increase the security level of the encrypted images. The rest of this paper is organized as follows. Section II gives a background about the current image encryption schemes. In Section III, the description of the proposed block-based transformation algorithm is presented. Section IV and V present the experimental results and discussion, while section VI presents a comparison between the proposed algorithm and those of other systems. Finally, section VII concludes the paper.

II. BACKGROUND

Encryption is the process of transforming the information to insure its security. With the huge growth of computer networks and the latest advances in digital technologies, a huge amount of digital data is being exchanged over various types of networks. It is often true that a large part of this information is either confidential or private. As a result, different security techniques have been used to provide the required protection [8].

The security of digital images has become more and more important due to the rapid evolution of the Internet in the digital world today. The security of digital images has attracted more attention recently, and many different image encryption methods have been proposed to enhance the security of these images [9].

Image encryption techniques try to convert an image to another one that is hard to understand [9]. On the other hand, image decryption retrieves the original image from the encrypted one. There are various image encryption systems to encrypt and decrypt data, and there is no single encryption algorithm satisfies the different image types.

Most of the algorithms specifically designed to encrypt digital images are proposed in the mid-1990s. There are two major groups of image encryption algorithms: (a) non-chaos selective methods and (b) Chaos-based selective or non-selective methods. Most of these algorithms are

Manuscript received 25 Jul 2007. Mohammad Ali Moh'd Bani Younes and Aman Jantan are with the Department of Computer Science, Universiti Sains Malaysia, and E-mail: mali.cod04@student.usm.my, aman@cs.usm.my.

(Advance online publication: 19 February 2008)

designed for a specific image format compressed or uncompressed, and some of them are even format compliant. There are methods that offer light encryption (degradation), while others offer strong form of encryption. Some of the algorithms are scalable and have different modes ranging from degradation to strong encryption [10].

Mitra A *et al.* [11] have proposed a random combinational image encryption approach with bit, pixel and block permutations.

Zhi-Hong Guan *et al.* [12] have presented a new image encryption scheme, in which shuffling the positions and changing the grey values of image pixels are combined to confuse the relationship between the cipher image and the plain image.

Sinha A. and Singh K. [13] proposed an image encryption by using Fractional Fourier Transform (FRFT) and JigSaw Transform (JST) in image bit planes.

Shujun Li *et al.* [14] have pointed out that all permutation-only image ciphers were insecure against known/chosen-plaintext attacks. In conclusion, they suggested that secret permutations have to be combined with other encryption techniques to design highly secured images.

Maniccam S.S. and Bourbakis N G. [10] proposed image and video encryption using SCAN patterns. The image encryption is performed by SCAN-based permutation of pixels and a substitution rule which together form an iterated product cipher.

Ozturk I. and Sogukpinar I. [15] proposed new schemes which add compression capability to the mirror-like image encryption MIE and Visual Cryptography VC algorithms to improve these algorithms.

Sinha A. and Singh K. [16] proposed a technique to encrypt an image for secure transmission using the digital signature of the image. Digital signatures enable the recipient of a message to authenticate the sender of a message and verify that the message is intact.

Droogenbroeck M.V. and Benedett R. [2] have proposed two methods for the encryption of an image; selective encryption and multiple selective encryption.

Maniccam S.S., Nikolaos G. and Bourbakis. [17] have presented a new methodology, which performs both lossless compression and encryption of binary and gray-scale images. The compression and encryption schemes are based on SCAN patterns generated by the SCAN methodology.

The proposed algorithm divides the image into random number of blocks with predefined maximum and minimum number of pixels, resulting in a stronger encryption and a decreased correlation.

III. THE PROPOSED TECHNIQUE

A. *Description of the Transformation Algorithm*

The transformation technique works as follows: the *original* image is divided into a random number of blocks that are then shuffled within the image. The generated (or transformed) image is then fed to the Blowfish encryption algorithm. The main idea is that an image can be viewed as an arrangement of blocks. The intelligible information present in an image is due to the correlation among the image elements in a given arrangement. This perceivable information can be reduced by decreasing the correlation among the image elements using certain transformation techniques.

The secret key of this approach is used to determine the seed. The seed plays a main role in building the transformation table, which is then used to generate the transformed image with different random number of block sizes. The transformation process refers to the operation of dividing and replacing an arrangement of the original image.

The image can be decomposed into blocks; each one contains a specific number of pixels. The blocks are transformed into new locations. For better transformation the block size should be small, because fewer pixels keep their neighbors. In this case, the correlation will be decreased and thus it becomes difficult to predict the value of any given pixel from the values of its neighbors. At the receiver side, the original image can be obtained by the inverse transformation of the blocks. A general block diagram of the transformation method is shown in Fig. 1.

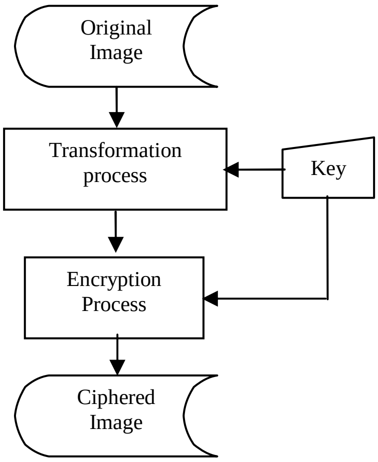


Fig. 1. General block diagram of the transformation algorithm

The transformation algorithm is presented below. It generates a transformation table that will be used to build a newly transformed image.

```
ALGORITHM CREATE_TRANSFORMATION_TABLE
1: Load Image
2: Input key
3: Get ImageWidth and ImageHeight
4:
   4.1: LowerHorizontalNoBlocks = Int(ImageWidth /10)
   4.2: LowerVerticalNoBlocks = Int(ImageHeight /10)
5: Randomize ()
6:
 6.1: HorizontalNoBlocks = RandomNum between
          (LowerHorizontalNoBlocks and ImageWidth)
6.2: VerticalNoBlocks = RandomNum between
          (LowerVerticalNoBlocks and ImageHeight)
7: NoBlocks = HorizontalNoBlocks * VerticalNoBlocks
8: Seed = | Hash value (Key) |
9: HashValue1 = |Hash value (first half of the Key)|
   HashValue2 = |Hash value (second half of the Key)|
10: Randomize using seed
   11: If HashValue1 > HashValue2 Then
```

```
            SEEDALTERNATE = 1
          Else
            SEEDALTERNATE = 2
        End If
12: I = 0
        Number-of-seed-changes (N) = 1
13: While I < NoBlocks
    R = RandomNum between (zero and NoBlocks -1)
    If R is not selected Then
      Assign location R to the block I
          I +=1
      Else
      If SEEDALTERNATE = 1 Then
        seed = seed + (HashValue1 Mod I) +1
        SEEDALTERNATE = 2
        Else
        seed = seed + (HashValue2 Mod I) + 1
        SEEDALTERNATE = 1
    Randomize (seed)
    End If
    Else
        Number-of-seed-changes += 1
    If Number-of-seed-changes > 500,000 then
      For K = 0 to NoBlocks -1
      If K not selected then
          Assign location K to Block I
            I=I+1
      End if
          Next K
    End if
End if
End While
END CREATE_TRANSFORMATION_TABLE

Input: BMP image file, a string Key
Output: Transformation table
```

```
ALGORITHM PERFORM_TRANSFORMATION
1: For I = 0 to NoBlocks -1
    1.1: Get the new location of block I from the
    transformation table
    1.2: Set block I in its new location
    END PERFORM_TRANSFORMATION

    Input: Original Image (BMP image file) and
    Transformation table
    Output: Transformed Image.
```

B. *Description of the combination technique*

The block-based transformation algorithm is based on the combination of image transformation followed by encryption (i.e. transformation algorithm followed by the Blowfish algorithm). The transformation algorithm and the Blowfish algorithm use the original image to produce three output images; (a) a ciphered image using Blowfish, (b) a transformed image using a transformation process and (c) a transformed image encrypted using Blowfish.

The correlation and entropy of the three images are computed and compared with each other. This technique aims at enhancing the security level of the encrypted images by reducing the correlation among image elements and increasing its entropy value.

Image measurements (correlation and entropy) will be carried out on the original image and the encrypted images with and without transformation algorithm the results are then analyzed. The overview model of the proposed technique is shown in Fig.2.

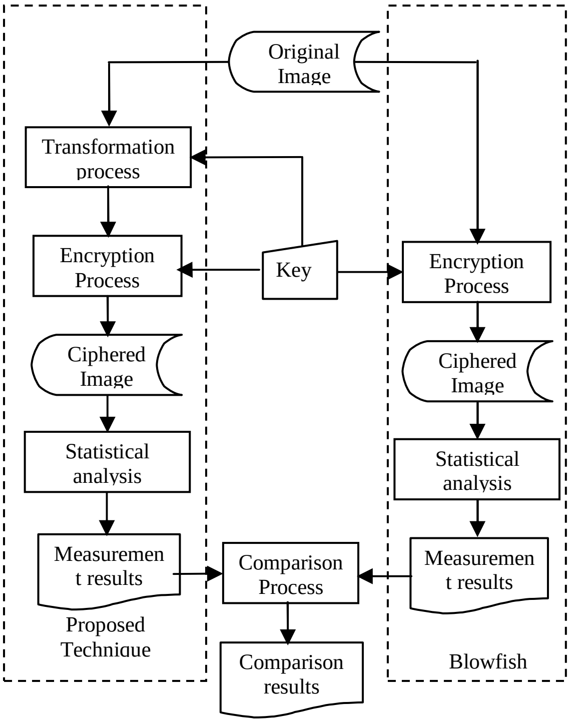


Fig. 2. An overview diagram of the proposed technique

IV. EXPERIMENTS

The method used to evaluate the present technique is described in Fig. 2. The algorithm was applied on a bit mapped (bmp) image that has the size of 300 pixels x 300 pixels with 256 colors. In order to evaluate the impact of the number of blocks on the correlation and entropy, three different cases were tested. The number of blocks and the block sizes for each case are shown in Table I.

Table I Different Cases to Test the Impact of the Number of Blocks on the Correlation and Entropy

Case Number	Number of blocks	Block size
1	30×30	10 pixels × 10 pixels
2	60×60	5 pixels × 5 pixels
3	100×100	3 pixels × 3 pixels

Each case produces three output images; (a) a ciphered image using the Blowfish algorithm, (b) a transformed image using the proposed algorithm, and (c) a ciphered image using the proposed algorithm followed by the Blowfish algorithm. For the rest of this paper, we use image A, image B, image C, and image D to denote the original image, the ciphered image using the Blowfish algorithm, the transformed image, and the ciphered image using the proposed algorithm followed by the Blowfish algorithm respectively.

Correlation and entropy are computed for each case according to equation (1) and equation (2).

$$r=\frac{n\sum(xy)-\sum x\sum y}{\sqrt{\left[n\sum(x^2)-(\sum x)^2\right]\left[n\sum(y^2)-(\sum y)^2\right]}} \qquad (1)$$

Where

r: correlation value

n: the number of pairs of data

$\sum xy$: sum of the products of paired data

$\sum x$: sum of x data

$\sum y$: sum of y data

$\sum x^2$: sum of squared x data

$\sum y^2$: sum of squared y data

Entropy defined as follows [18]-[19].

$$H_e = -\sum_{k=0}^{G-1} P(k)\log_2(P(k)) \qquad (2)$$

Where:

H_e: entropy.

G: gray value of input image (0... 255).

$P(k)$: is the probability of the occurrence of symbol k.

Case 1: The image is decomposed into 10 pixels × 10 pixels blocks. Fig.3. shows the resulted images.

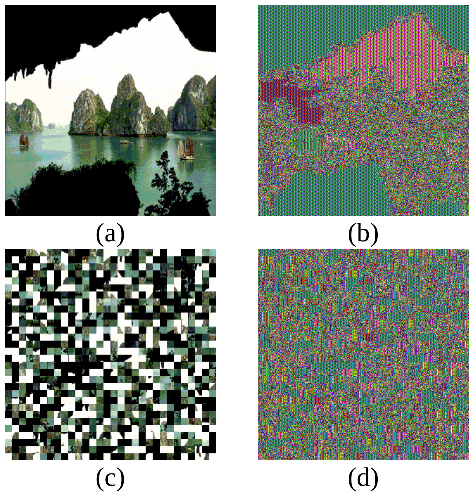
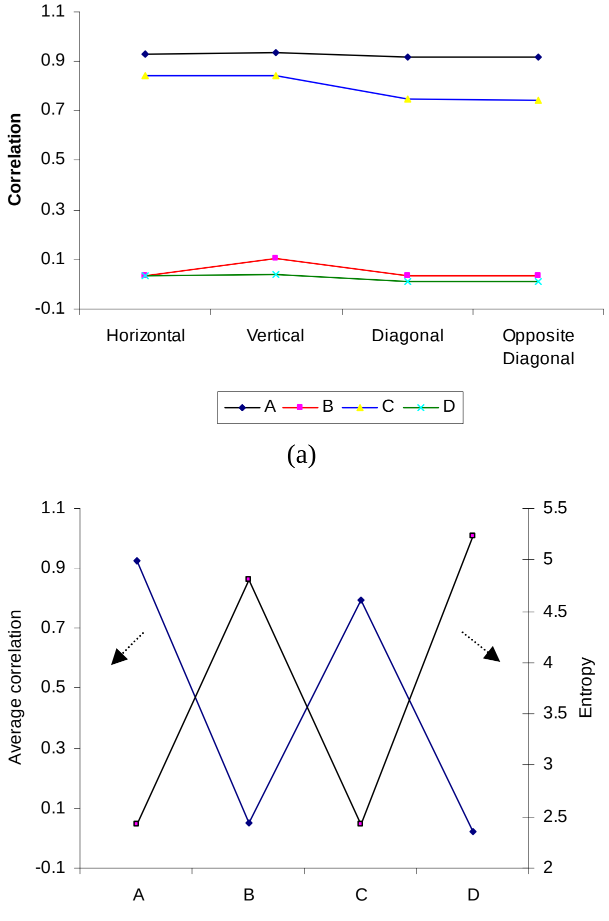

(a) (b)
(c) (d)

Fig. 3. Results of encryption by using 10 pixels × 10 pixels blocks. (a) Original image. (b) Encrypted image using Blowfish. (c) Transformed image. (d) Encrypted image using transformed followed by the Blowfish algorithm.

The correlation and entropy results of this case are summarized in Table II and Fig. 4

Table II Results of Correlation and Entropy values of Case 1.

Measurement		A	B	C	D
Correlation	Horizontal	0.933	0.035	0.843	0.034
	Vertical	0.936	0.107	0.844	0.038
	Diagonal	0.919	0.032	0.748	0.013
	Opposite Diagonal	0.916	0.034	0.745	0.009
	Average	0.926	0.052	0.795	0.023
Entropy value		2.431	4.799	2.431	5.231



(b)

Fig. 4. Correlation and entropy values of Case 1. (a) Correlation between directions. (b) Average correlation and entropy.

Case 2: The image is decomposed into 5 pixels × 5 pixels blocks. Fig. 5 shows the resulted images.

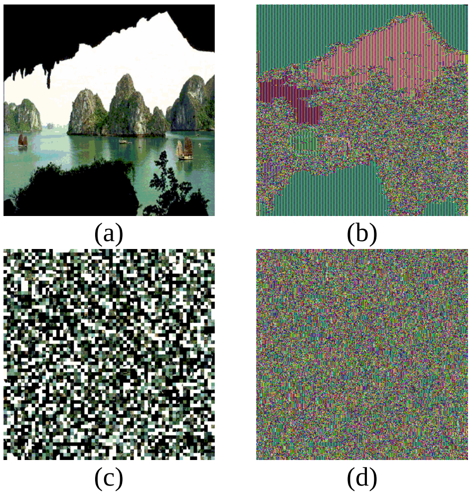
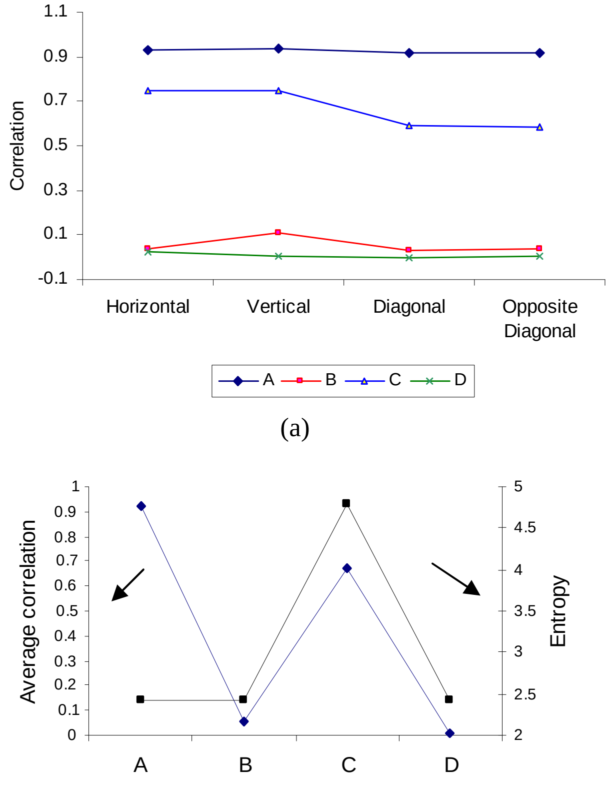

(a) (b)
(c) (d)

Fig. 5. Results of encryption by using 5 pixels × 5 pixels blocks. (a) Original image. (b) Encrypted image using Blowfish. (c) Transformed image. (d) Encrypted image using transformed followed by the Blowfish algorithm.

The correlation and entropy results of this case are summarized in Table III and Fig. 6.

Table III Results of Correlation and Entropy values of Case 2.

Measurement		A	B	C	D
Correlation	Horizontal	0.9325	0.0346	0.7469	0.0245
	Vertical	0.9362	0.1073	0.7497	0.0067
	Diagonal	0.9186	0.0321	0.5881	0
	Opposite Diagonal	0.9156	0.0337	0.5877	0.0056
	Average	0.9257	0.0519	0.6681	0.0092
Entropy value		2.431	2.4305	4.799	2.4305

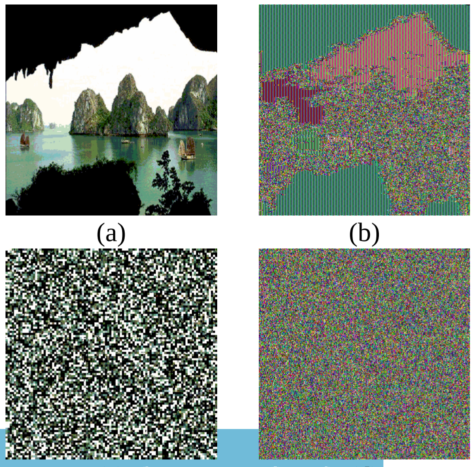


(a)

(b)

Fig. 6. Correlation and entropy values of Case 2. (a) Correlation between directions. (b) Average correlation and entropy.

***Case* 3:** The image is decomposed into 3 pixels × 3 pixels blocks. Fig. 7 shows the resulted images.



(a) (b)

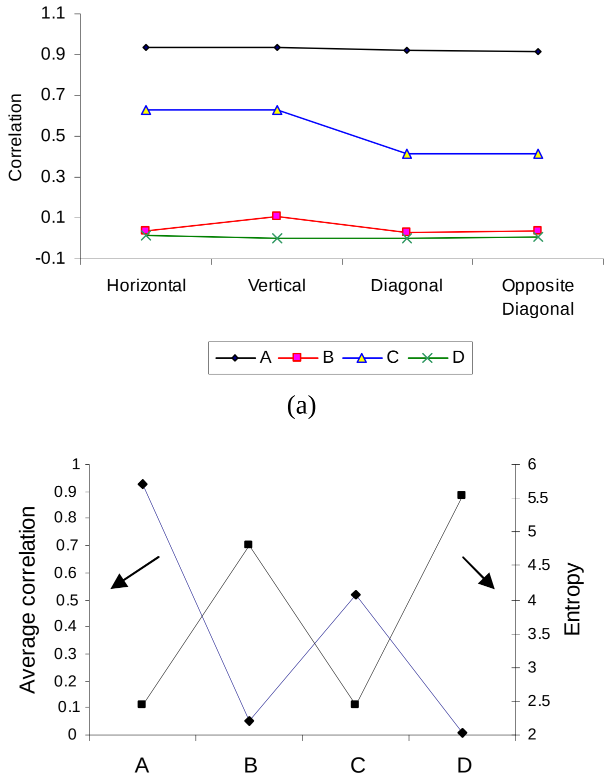
(c) (d)

Fig. 7. Results of encryption by using 3 pixels × 3 pixels blocks. (a) Original image. (b) Encrypted image using Blowfish. (c) Transformed image. (d) Encrypted image using transformed followed by the Blowfish algorithm.

The results of this case are summarized in Table IV and Fig. 8.

Table IV Results of Correlation and Entropy values of Case 3.

Measurement		A	B	C	D
Correlation	Horizontal	0.9325	0.0346	0.6289	0.0129
	Vertical	0.9362	0.1073	0.6265	0.0034
	Diagonal	0.9186	0.0321	0.4145	0.0014
	Opposite Diagonal	0.9156	0.0337	0.4146	0.0049
	Average	0.9257	0.0519	0.5211	0.0056
Entropy value		2.4305	4.799	2.4305	5.5281

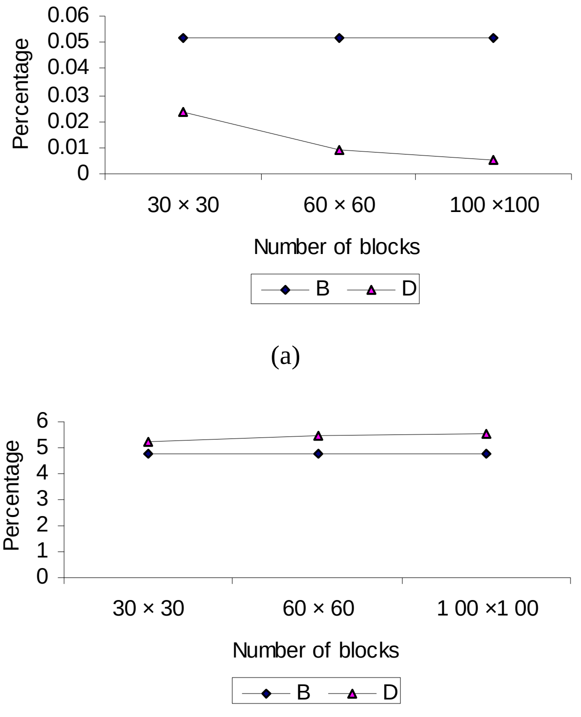


(a)

(b)

Fig. 8. Correlation and entropy values of Case 3. (a) Correlations between directions. (b) Average correlation and entropy.

(Advance online publication: 19 February 2008)



(a)

(b)

Fig. 9 Average correlation and entropy for image B and image D. (a) Correlation. (b) Entropy

As shown in Fig. 9 above, the three different cases showed that the proposed algorithm resulted in lower correlation and higher entropy than the Blowfish algorithm alone. The best performance occurs for the smallest image block size.

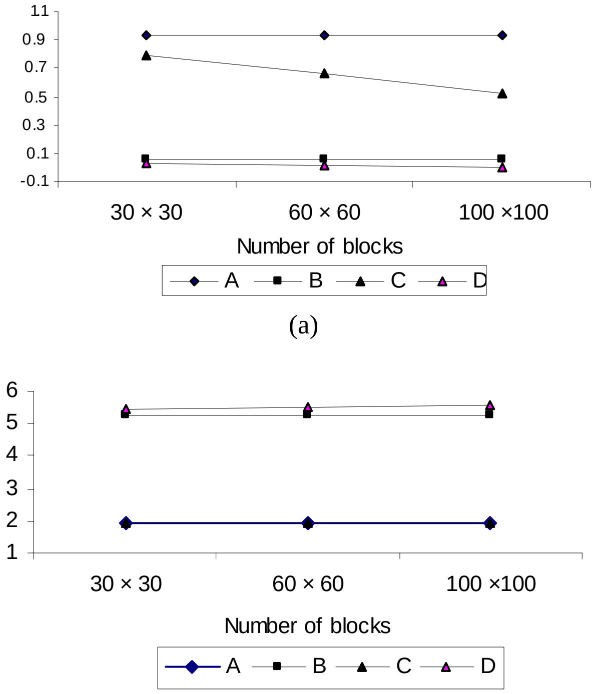
V. RESULTS AND DISCUSSION

The above cases show that using the proposed algorithm followed by the Blowfish algorithm resulted in a lower correlation and a higher entropy compared to using the Blowfish alone. Dividing the image into a larger number of blocks made the performance even better. The results showed that the correlation was reduced even further and the entropy was increased as the number of blocks is increased.

Table V summarizes the results of the different cases, while Fig. 10 shows the effect of block size on the correlation and entropy value.

Table V Results of Correlation and Entropy values of image (a) in Fig. 3.

Measurement	Number of blocks	A	B	C	D
Average correlation	30 × 30	0.9257	0.0519	0.7952	0.0234
	60 × 60			0.6681	0.0092
	100 ×100			0.5211	0.0056
Entropy	30 × 30	2.4305	4.799	2.4305	5.2305
	60 × 60				5.4737
	100 ×100				5.5281



(a)

(b)

Fig. 10. Correlation and entropy values versus the number of blocks. (a) Correlation. (b) Entropy.

VI. COMPARISON BETWEEN THE PROPOSED ALGORITHM AND OTHER ALGORITHMS

In this section, two experiments were carried out using the image of Fig. 11. In the first experiment, the image was divided into different blocks that are shuffled according to the proposed algorithm and then followed by one of four commonly used encryption algorithms; Blowfish, Twofish, RijnDael, or RC4. These algorithms are commercially available, so we applied them on the ciphered image that resulted from applying the proposed algorithm on the different block sizes of the original image. The correlation and entropy of each one was compared with applying the corresponding algorithm alone. The results of this experiment are shown in Table VI and Fig. 12.

Fig. 12 shows that using the proposed algorithm along with the other algorithms resulted in a better performance compared to using the other algorithms alone**.**

In the second experiment, commonly used algorithms in literature were used to compare with the proposed algorithm. Since those algorithms are not available for us, we compared the ciphered images generated by applying these algorithms with the ciphered images generated by applying the proposed algorithm on the original image. The correlation and entropy of the ciphered images were recorded for each algorithm. The results of this application are shown in Table VII and plotted in Fig. 13.

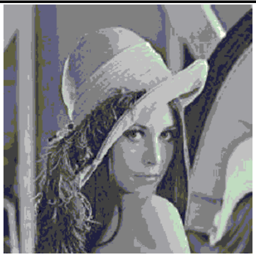


Fig. 11. Image used to compare the proposed algorithm with the commonly used ones in industry and literature

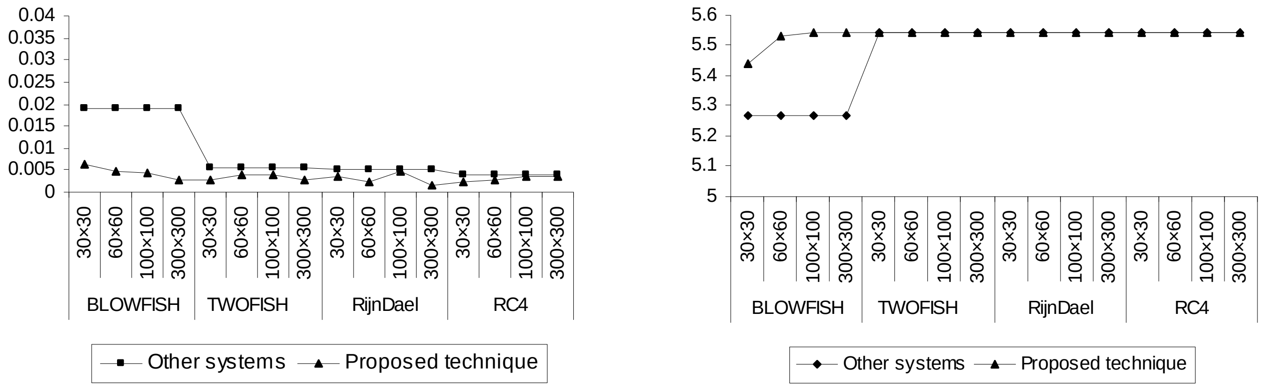
Table VI Correlation and Entropy when Appling. (a) Commercially available algorithms alone. (b) Commercially available algorithms preceded by the proposed algorithm when applied to the image of Fig. 11

Commercially available algorithms alone		
System	Correlation	Entropy
BLOWFISH(448)	0.0189	5.2703
TWOFISH (256)	0.0054	5.5437
RijnDael (AES256)	0.0051	5.5436
RC4(2048)	0.0038	5.5437

(a)

Commercially available algorithms preceded by the proposed algorithm			
System	Number of blocks	Correlation	Entropy
BLOWFISH(448)	30×30	0.0063	5.4402
	60×60	0.0049	5.5286
	100×100	0.0044	5.5407
	300×300	0.0028	5.5438
TWOFISH (256)	30×30	0.0026	5.5437
	60×60	0.0040	5.5439
	100×100	0.0041	5.5438
	300×300	0.0029	5.5438
RijnDael (AES256)	30×30	0.0034	5.5440
	60×60	0.0024	5.5439
	100×100	0.0049	5.5438
	300×300	0.0016	5.5439
RC4(2048)	30×30	0.0024	5.5438
	60×60	0.0026	5.5437
	100×100	0.0034	5.5439
	300×300	0.0034	5.5438

(b)

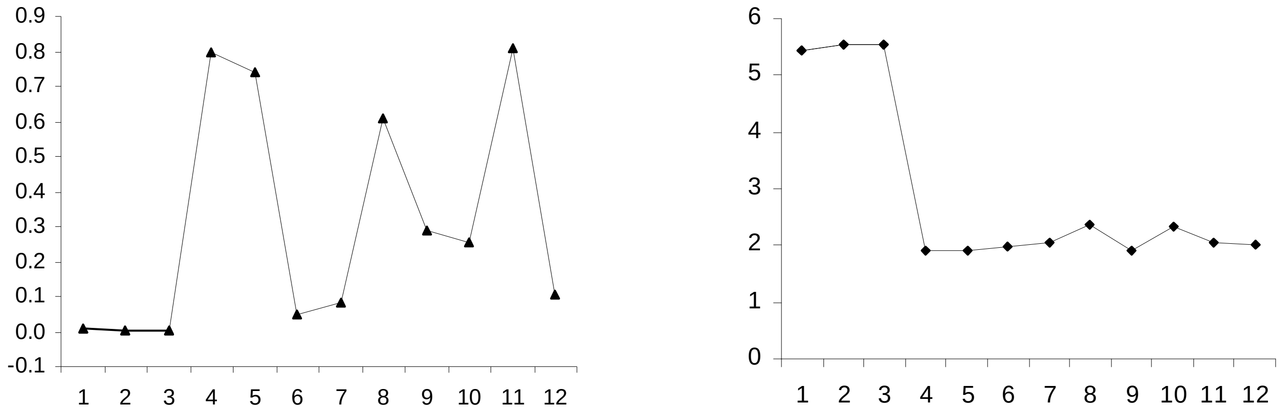


(a) (b)

Fig. 12. Correlation and entropy values of the algorithms shown in Table VI. (a) Correlation. (b) Entropy.

Table VII Correlation and Entropy values of commonly used Encryption Algorithms in Literature when applied to the image of Fig. 11.

Number	Image encryption algorithm	Measurement	
		Correlation	Entropy
1	Proposed technique 30 × 30	0.0063	5.4402
2	Proposed technique 60 × 60	0.0049	5.5286
3	Proposed technique 100 × 100	0.0044	5.5407
4	Block Permutation 8×8	0.7977	1.9216
5	Pixel Permutation	0.7418	1.9017
6	Bit Permutation	0.0479	1.9769
7	[Block, bit, pixel] Combination	0.0812	2.0631
8	3D Jigsaw transform	0.6096	2.3735
9	Arnold cat map	0.2882	1.9208
10	Chen's chaotic system	0.2569	2.3440
11	A naive selective encryption (3 bits encrypted)	0.8078	2.0635
12	A naive selective encryption (7 bits encrypted)	0.1037	2.0182



(a) (b)

Fig. 13. Correlation and entropy values of the algorithms shown in Table VII. (a) Correlation. (b) Entropy.

Fig. 12 and Fig. 13 show that the proposed algorithm resulted in lower correlation and higher entropy and thus in a better encrypted image than all other algorithms tested in this paper.

VII. CONCLUSION

In this paper a simple and strong method has been proposed for image security using a combination of block-based image transformation and encryption techniques. The cases showed that the correlation was decreased when the proposed algorithm was applied to them before the Blowfish algorithm. Experimental results of the proposed technique showed that an inverse relationship exists between number of blocks and correlation, and a direct relationship between number of blocks and entropy. When compared to many commonly used algorithms, the proposed algorithm resulted in the best performance; the lowest correlation and the highest entropy.

REFERENCES

[1] W. Lee, T. Chen and C. Chieh Lee, "Improvement of an encryption scheme for binary images," *Pakistan Journal of Information and Technology*. Vol. 2, no. 2, 2003, pp. 191-200. http://www.ansinet.org/

[2] M. V. Droogenbroech, R. Benedett, "Techniques for a selective encryption of uncompressed and compressed images," In ACIVS'02, Ghent, Belgium. *Proceedings of Advanced Concepts for Intelligent Vision Systems*, 2002.

[3] S. Changgui, B. Bharat, "An efficient MPEG video encryption algorithm," *Proceedings of the symposium on reliable distributed systems, IEEE computer society Press*, 1998, pp. 381-386.

[4] S. Fong, P.B. Ray, and S. Singh, "Improving the lightweight video encryption algorithm," *proceeding of iasted international conference, single* processing, pattern recognition and application, 2002, pp. 25-28.

[5] S. P. Nana'vati., P. K. panigrahi. "Wavelets:applications to image compression- I,". *joined of the scientific and engineering computing*, vol. 9, no. 3, 2004, pp. 4- 10.

[6] c. Ratael, gonzales, e. Richard, and woods, "Digital image processing," 2nd ed, Prentice hall, 2002.

[7] AL. Vitali, A. Borneo, M. Fumagalli and R. Rinaldo, "Video over IP using standard-compatible multiple description coding," *Journal of Zhejiang*

University- Science A, vol. 7, no. 5 ,2006, pp. 668-676.

[8] H. El-din. H. Ahmed, H. M. Kalash, and O. S. Farag Allah, "Encryption quality analysis of the RC5 block cipher algorithm for digital images," Menoufia *University, Department of Computer Science and Engineering, Faculty of Electronic Engineering, Menouf-32952, Egypt*, 2006.

[9] Li. Shujun, X. Zheng "Cryptanalysis of a chaotic image encryption method," Inst. of Image Process. Xi'an Jiaotong Univ., Shaanxi, This paper appears in: Circuits and Systems, ISCAS 2002. IEEE International Symposium on Publication Date: 2002, Vol. 2, 2002, page(s):708,711.

[10] S.S. Maniccam, N.G. Bourbakis, "Image and video encryption using SCAN patterns," *Journal of Pattern Recognition Society*, vol. 37, no. 4, pp.725–737, 2004.

[11] A. Mitra, , Y V. Subba Rao, and S. R. M. Prasnna, "A new image encryption approach using combinational permutation techniques," *Journal of computer Science*, vol. 1, no. 1, p.127, 2006, Available: http://www.enformatika.org

[12] G. Zhi-Hong, H. Fangjun, and G.Wenjie, "Chaos-based image encryption algorithm," *Department of Electrical and computer Engineering*, University of Waterloo, ON N2L 3G1, Canada. Published by: *Elsevier*, 2005, pp. 153-157.

[13] A. Sinha, K. Singh, "Image encryption by using fractional Fourier transform and Jigsaw transform in image bit planes," Source: *optical engineering, spie-int society optical engineering*, vol. 44, no. 5 , 2005, pp.15-18.

[14] Li. Shujun, Li. Chengqing, C. Guanrong, *Fellow., IEEE.*, Dan Zhang., and Nikolaos,G., Bourbakis *Fellow., IEEE.* "A general cryptanalysis of permutation-only multimedia encryption algorithms," 2004, http://eprint.iacr. Org/2004/374.pdf

[15] I. Ozturk, I.Sogukpinar, "Analysis and comparison of image encryption algorithm," *Journal of transactions on engineering, computing and technology December*, vol. 3, 2004, p.38.
http: //www.enformatika.org/

[16] A. Sinha, K. Singh, "A technique for image encryption using digital signature," Source: *Optics Communications*, vol.218, no. 4, 2003, pp.229-234.http://www.elsevier.com/

[17] S.S. Maniccam., G.Nikolaos, and Bourbakis, "Lossless image compression and encryption using SCAN," *Journal of: Pattern Recognition*, vol. 34, no. 6., , 2001, pp.1229– 1245.

[18] M. Sonka, V. Hlavac. and R. Boyle, "Digital image processing," in: image Processing, Analysis, and Machine Vision, 1998, 2nd ed. *http://www.pws.com*

[19] D. Feldman, "A brief introduction to: information theory, excess entropy and computational mechanics," college of the atlantic 105 eden street, bar harbor, me 04609, 2002, http://hornacek.coa.edu/

Mohammad. A. B. Younes received BSc in computer science from Yarmouk University in Irbid Jordan, MSc from Sudan University of science and technology in 1986 and 2003 respectively and currently enrolled in the PhD program in computer science in the Universiti Sains Malaysia.

Dr. Aman Jantan received BSc in computer science, Master in AI, PhD in Software Engineering from Universiti Sains Malaysia, penang in 1993, 1996 and 2002 respectively and currently Lecturer in School of Computer Sciences in Universiti Sains Malaysia, penang.