Avrupa Bilim ve Teknoloji Dergisi
Özel Sayı 36, S. 91-95, Mayıs 2022

Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
Special Issue 36, pp. 91-95, May 2022

Research Article

A Color Image Scrambling Method Based on Zigzag Transform and Cross-channel Permutation

Mehmet Demirtaş[1*]

[1*] Necmettin Erbakan University, Faculty of Engineering, Departmant of Electrical and Electronics Engineering, Konya, Turkey, (ORCID: 0000-0002-9018-3124), mdemirtas@erbakan.edu.tr

(1st International Conference on Engineering and Applied Natural Sciences ICEANS 2022, May 10-13, 2022)

(**DOI:** 10.31590/ejosat.1106823)

ATIF/REFERENCE: Demirtaş, M. (2022). A Color Image Scrambling Method Based on Zigzag Transform and Cross-channel Permutation. *European Journal of Science and Technology*, (36), 91-95.

Abstract

It is a challenging task to maintain the privacy and security of color images that are shared over public networks. To address this challenge, image encryption methods can be used to protect the confidentiality of the shared image. As a part of spatial domain-based image encryption, the scrambling process aims to reduce the correlation between adjacent pixels. Furthermore, the correlation between different channels of an RGB color image can be decreased by applying a cross-channel permutation. To achieve both decorrelations, this paper presents a new image scrambling method that is based on zigzag transform and cross-channel permutation. Firstly, a color image's red, green, and blue channels are obtained, and the zigzag transform is applied to each channel, respectively. The transformed channels are concatenated for further shuffling. A chaotic tent map is employed to create the necessary parameters to perform cross-channel permutation. The cross-channel permutation ensures swapping pixels between different channels. Performance measures of the proposed method show that the correlation coefficients of an RGB image can be significantly reduced. The information entropy of each channel is also greatly increased with this method. The average execution time is found to be less than 0.08 s for a 256 x 256 color image, indicating that the proposed method can be used in image encryption applications.

Keywords: color image encryption, cross-channel permutation, image scrambling, tent map, zigzag transform.

Zigzag Dönüşümü ve Kanallar Arası Permütasyona Dayalı Bir Renkli Görüntü Karıştırma Yöntemi

Öz

Genel ağlar üzerinden paylaşılan renkli görüntülerin gizliliğini ve güvenliğini sağlamak zorlu bir iştir. Paylaşılan görüntünün gizliliğini korumak için görüntü şifreleme yöntemleri kullanılabilir. Uzamsal alan tabanlı görüntü şifrelemenin bir parçası olarak, karıştırma işlemi, bitişik pikseller arasındaki korelasyonu azaltır. Ayrıca, bir renkli görüntünün farklı kanalları arasındaki korelasyon, kanallar arası bir permütasyon ile azaltılabilir. Bu makale zikzak dönüşümüne ve çapraz kanal permütasyonuna dayanan yeni bir görüntü karıştırma yöntemi sunmaktadır. İlk olarak renkli bir görüntünün kırmızı, yeşil ve mavi kanallarına zikzak dönüşümü uygulanır. Kanallar arası permütasyon gerçekleştirmek için gerekli parametreler kaotik bir çadır haritası ile elde edilir. Çapraz kanal permütasyonu, farklı kanallar arasında piksellerin değiştirilmesini sağlar. Önerilen yöntemin performans ölçümleri, bir RGB görüntüsünün korelasyon katsayılarının önemli ölçüde azaltılabileceğini göstermektedir. Bu yöntemle her kanalın bilgi entropisi de büyük ölçüde artırılır. 256 x 256 renkli bir görüntü için ortalama yürütme süresinin 0.08 s'den az olduğu tespit edilmiştir ve bu da önerilen yöntemin görüntü şifreleme uygulamalarında kullanılabileceğini göstermektedir.

Anahtar Kelimeler: çadır haritası, görüntü karıştırma, kanallar arası permütasyon, renkli görüntü şifreleme, zikzak dönüştürme.

[*] Corresponding Author: mdemirtas@erbakan.edu.tr

1. Introduction

Image encryption refers to the process of making a transmitted image unidentifiable to third parties. When both the location and value of pixels in an image change, this is called spatial domain-based image encryption (Kaur & Kumar, 2020). The spatial domain-based image encryption methods generally consist of two stages: scrambling and diffusion. In the scrambling stage, pixel values of the transmitted image are kept unchanged, but pixel positions are shuffled. Generally, a chaotic map (Muthu & Murali, 2021) can be employed to determine the shuffling positions so that correlation between adjacent pixels can be reduced. For a color image, correlations between R, G, and B channels should also be reduced (Z. Li, Peng, Tan, & Li, 2020).

Zigzag transform scans all elements of a matrix in zigzag form and creates a 1D array (Gao & Wang, 2021). This 1D array can be reshaped as a 2D matrix to be the same size as the original matrix. Therefore, a plain image, which can be represented as a 2D matrix, can be scrambled using a zigzag transform (Gao & Wang, 2021; Wang & Chen, 2021; Xingyuan, Junjian, & Guanghui, 2019; Zhang & Gong, 2022). An example of zigzag transformation is shown in Fig. 1 for a 3 x 3 matrix. Although the zigzag transform can be used to shuffle a matrix, it has some weaknesses. The elements in the matrix are scanned adjacently, which means that the correlation between neighboring elements cannot be reduced by the desired amount. Therefore, additional pixel scrambling is generally needed (Ramasamy, Ranganathan, Kadry, Damaševičius, & Blažauskas, 2019). In this paper, a chaotic tent map (C. Li, Luo, Qin, & Li, 2017) is utilized to calculate the required sequences for further scrambling.

In this study, as shown in Fig. 1, 1D arrays are obtained using a zigzag transform for each channel of the plain image. Subsequently, these 1D arrays are concatenated horizontally. The concatenated 1D matrix is resized as a 2D matrix. Finally, the columns of this 2D matrix are shuffled according to the parameters produced by the chaotic tent map. To sum up, in-channel scrambling is achieved by zigzag transformation; and cross-channel scrambling is performed using a chaotic tent map.

In Section 2, the proposed color image scrambling method is explained. Section 3 presents the performance evaluation of the proposed method. Finally, in Section 4, the overall study is summarized.

2. Material and Method

2.1. The Chaotic Tent Map

The chaotic tent map is a one-dimensional, discrete, and non-invertible map and it can be expressed as in Eq. (1).

$$x_{n+1} = \begin{cases} rx_n & \text{if } x_n < 0.5, \\ r(1-x_n) & \text{otherwise} \end{cases}, \quad (1)$$

where r is the control parameter between 0 and 2. When the control parameter is selected sufficiently close to 2, this map shows fully chaotic behavior in (0,1) (Kanso, 2011). The map shown in (1) will produce completely different values for two slightly different initial values. If the initial value x_0 is selected as a plain color image dependent value, then entirely different chaotic sequences can be generated even if there are two very similar plain images. As a result, very similar plain images can produce totally different scrambled images.

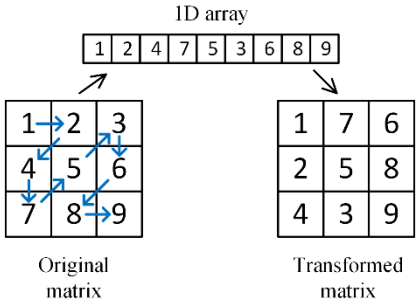


Figure 1. Zigzag transform of 3 x 3 matrix

2.2. The Proposed Color Image Scrambling Method

A color image P with a size of $M \times N$ is the input plain image. Three channels of the plain image are obtained as R, G, and B. The proposed color image scrambling method's steps are given as follows.

Step 1 R, G, and B channels are transformed using a zigzag transform to obtain three 1D arrays named R_z, G_z, and B_z.

Step 2 R_z, G_z, and B_z arrays are concatenated horizontally to create a 1×3MN array. This array is reshaped as a 2D matrix with a size of 3×MN named P_z.

Step 3 The chaotic tent map in (1) is iterated for $MN + L$ times with an initial condition x_0. The control parameter is selected as $r = 1.9999$ and the initial condition x_0, and L are calculated using the following equations.

$$x_0 = \text{mod}\left(\frac{\text{sum}(R+G+B)}{3MN}, 0.5\right) + k_1 \quad (2)$$

$$L = \text{mod}(\text{sum}(R+G+B), 1000) + 100k_2 \quad (3)$$

where $\text{sum}(R+G+B)$ represents the total sum of all pixel values in the plain image. The secret keys $k_1 \in [0,0.5]$ and $k_2 \in \mathbb{N}$ can be determined by the user.

Step 4 The first L values are discarded to get rid of the transient effects. A pseudo-random chaotic sequence is obtained as $S_x = \{x_1, x_2, \dots, x_{MN}\}$, where all elements range from 0 to 1. The elements in S_x are sorted in ascending order. The index vector $C = (c_1, c_2, \dots, c_{MN})$, which marks the positions of the sorted elements in the original sequence S_x, is obtained.

Step 5 The index vector is used to shuffle columns of the matrix P_z. The following swapping expression is used to perform this operation.

$$P_{zs}(:, i) = P_z(:, C(i)) \quad (4)$$

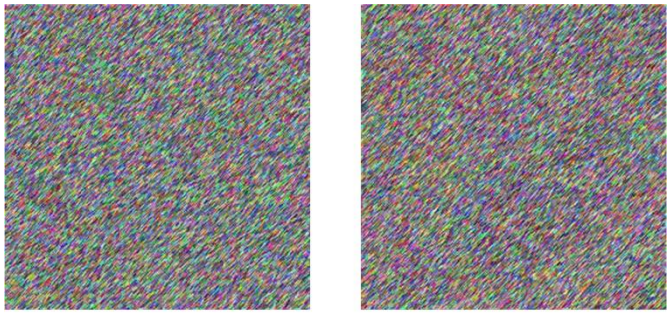
where P_{zs} is the shuffled matrix with a size of $3 \times MN$ and $i = 1,2 \dots, MN$.

Step 6 The first, second, and third rows of the P_{zs} represent shuffled red, green, and blue channels, respectively. Each row is reshaped as a 2D matrix to obtain R_s, G_s, and B_s. Finally, those three shuffled matrices are combined to obtain the shuffled color image P_s.

2.2. Descrambling Process

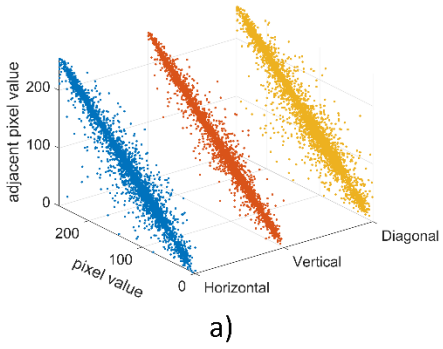
The descrambling of the shuffled image can be achieved by executing the scrambling method in reverse order

Figure. 2 Plain image and scrambled image

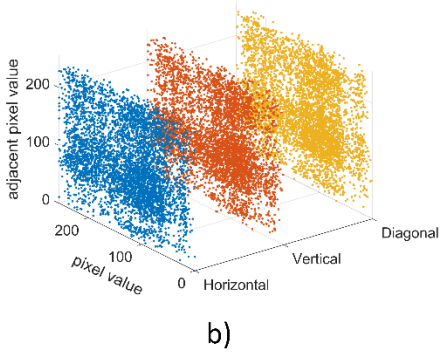


a) b)

Figure. 3 Key sensitivity analysis a) Descrambling with $k_1 + 10^{-16}$ b) Decrambling with $k_2 - 1$



a)



b)

Figure. 4 Correlation analysis a) Plain Lena b) Scrambled Lena

3. Results and Discussion

To test the proposed image scrambling method's performance, key sensitivity analysis, correlation analysis, entropy analysis, differential analysis, and execution time analysis are carried out. Two standard color Lena images (256 x 256 and 512 x 512) are used during the experimental work. In Fig.2., the plain 256x256 Lena image and its corresponding scrambled image are shown.

3.1. Key Sensitivity Analysis

There are two different parameters selected as the secret keys in this work: $k_1 \in [0,0.5]$ and $k_2 \in \mathbb{N}$. These parameters are selected as $k_1 = 0.25$ and $k_2 = 100$ for the scrambling process shown in Fig. 2. If these secret keys are known precisely, the plain image can be obtained from the scrambled image by the descrambling process. However, even if there is a slight change in a secret key, the descrambled image should be different from the plain image. This is called key sensitivity. To test the proposed method's key sensitivity, k_1 and k_2 are changed by 10^{16} and 1, respectively. The descrambled images for the slightly changed secret keys are shown in Fig 3. These images are entirely different from the plain image; therefore, the proposed image scrambling method is sensitively dependent on secret keys.

3.2. Correlation Analysis

A plain image's neighboring pixels have very similar pixel values, which causes a high correlation in horizontal, vertical, and diagonal directions. Image scrambling's objective is to reduce these high correlations. To visualize the correlation change, 5000 different pixel pairs are randomly chosen from the plain and scrambled Lena images, in three directions. The distribution of the selected pixel pairs is shown in Fig. 4. As is seen in Fig.4, the strong correlation observed in the plain image is greatly reduced with the help of the proposed scrambling process in all directions.

Table 1. Correlation coefficients for plain and scrambled Lena images with different sizes

	Correlation coefficient		
	Horizontal	**Vertical**	**Diagonal**
Plain Lena (256x256)	0.95624	0.98108	0.94217
Scrambled Lena (256x256)	0.00809	0.01753	0.01255
Plain Lena (512x512)	0.97795	0.98731	0.97136
Scrambled Lena (512x512)	-0.01313	0.01202	0.00954

The correlation coefficient c, which can be used to quantify the correlation change, is expressed as in (5).

$$c = \frac{\sum_{i=1}^{n}\left(x_i - E(x)\right)\left(y_i - E(y)\right)}{\sqrt{\sum_{i=1}^{n}\left(x_i - E(x)\right)^2 \sum_{i=1}^{n}\left(y_i - E(y)\right)^2}} \quad (5)$$

where x and y are the neighboring pixel values, n is the number of randomly chosen pixel pairs, and $E(x)$ and $E(y)$ are the means of x_i and y_i pixel values, respectively. The correlation coefficient should be close to 1 and 0 for the plain image and scrambled image, respectively.

5000 different pixel pairs are randomly chosen from the plain and scrambled Lena images, in horizontal, vertical, and diagonal directions. The correlation coefficients are computed using Eq. (5) and listed in Table 1 for two Lena images with sizes of 256 x 256 and 512 x 512. The calculated correlation coefficient values in all directions show that the strong correlations of the plain images are reduced by the scrambling method.

Table 2. Information entropies for plain and scrambled Lena images with different sizes

	Information entropy	
Lena(256x256)	**Plain**	**Scrambled**
Red channel	7.28331229	7.76153114
Green channel	7.58510891	7.75970589
Blue channel	7.04193705	7.76136489
Lena(512x512)	**Plain**	**Scrambled**
Red channel	7.25310236	7.75019948
Green channel	7.59403792	7.74912571
Blue channel	6.96842695	7.75010782

3.3. Information Entropy Analysis

Information entropy can be used as an indication of randomness for the plain and scrambling images. Its value should be larger for scrambled images. Ideally, it should be equal to 8 for every channel of an RGB color image. A signal's information entropy can be calculated as in (6).

$$H = -\sum_{i=0}^{255} P(s_i) \log_2 P(s_i) \tag{6}$$

where $P(s_i)$ is the probability of occurrence of s_i. Table 2 lists the information entropy values for plain and scrambled Lena images with different sizes. The increase in the randomness due to the scrambling method is demonstrated.

3.4. Differential Analysis

A good scrambling algorithm should produce two very different scrambled images for two slightly different plain images. Therefore, a scrambling method must be sensitive to the input plain image. The sensitivity to the plain image can be evaluated using a metric named Number of Pixel Change Rate (NPCR), which computes the ratio of the number of different pixels between two scrambled images. Theoretically, the NPCR value should be close to 99.61% (Y. Li, Wang, & Chen, 2017). NPCR can be calculated using the Eqs. (7) and (8).

$$NPCR = \frac{1}{M \times N} \sum_{j=1}^{N} \sum_{i=1}^{M} D(i,j) \times 100\ \% \tag{7}$$

$$D(i,j) = \begin{cases} 1 & \text{if } S_1(i,j) \neq S_2(i,j) \\ 0 & \text{if } S_1(i,j) = S_2(i,j) \end{cases} \tag{8}$$

where S_1 and S_2 are scrambled images, whose corresponding plain images have only one pixel difference. To find NPCR values, a pixel is randomly chosen from the plain Lena image and its value is changed by 1. Then, the modified plain image is scrambled using the proposed method to acquire S_1 and S_2. In this way, the NPCR value of each channel can be calculated using (7) and (8). In Table 3, NPCR values are listed for Lena images with different sizes. The calculated NPCR values are very close to the theoretical values, which means that this scrambling method can resist differential attacks.

3.5. Execution Time

The proposed algorithm is run on MATLAB 2017b. The scrambling time and descrambling time of the proposed algorithm are measured by a PC with an Intel Core 2.80 GHz processor and 16 GB RAM. Table 4 lists the average scrambling and descrambling times when the algorithm is executed 10 times. A color image with a size of 512x512 can be scrambled under 0.25 s with this method. Similarly, for a 256x256 color image, the scrambling time is less than 0.08 s. The execution times are short enough to use the suggested image scrambling method in real-time image encryption algorithms.

Table 3. NPCR values for Lena images with different sizes

	NPCR (%)		
	Red	**Green**	**Blue**
Lena (256x256)	99.50256	99.46899	99.48730
Lena (512x512)	99.48120	99.49150	99.48540

Table 4. Scrambling and descrambling times for Lena images with different sizes

	Time (s)	
	Scrambling	**Descrambling**
Lena (256x256)	0.0779	0.0417
Lena (512x512)	0.2430	0.1433

4. Conclusions and Recommendations

In this paper, a novel, fast and secure image scrambling method, which is based on zigzag transformation and cross-channel permutation, is proposed. The proposed method's several performance parameters are simulated for two color Lena images with sizes of 256x256 and 512x512. The scrambling method is shown to be very sensitive to the secret keys. Also, it can significantly reduce the correlation between adjacent pixels. The information entropy of the plain image is highly increased with the help of the scrambling method. Moreover, the differential analysis proves that the method can withstand differential attacks due to the NPCR values close to ideal ones. The method is also very fast as shown in the execution time analysis, which means that it can be applied to image encryption schemes.

References

Gao, H., & Wang, X. (2021). Chaotic Image Encryption Algorithm Based on Zigzag Transform With Bidirectional Crossover From Random Position. *IEEE Access, 9*, 105627-105640. doi:10.1109/ACCESS.2021.3099214

Kanso, A. (2011). Self-shrinking chaotic stream ciphers. *Communications in Nonlinear Science and Numerical Simulation, 16*(2), 822-836. doi:https://doi.org/10.1016/j.cnsns.2010.04.039

Kaur, M., & Kumar, V. (2020). A Comprehensive Review on Image Encryption Techniques. *Archives of Computational Methods in Engineering, 27*(1), 15-43. doi:10.1007/s11831-018-9298-8

Li, C., Luo, G., Qin, K., & Li, C. (2017). An image encryption scheme based on chaotic tent map. *Nonlinear Dynamics, 87*(1), 127-133. doi:10.1007/s11071-016-3030-8

Li, Y., Wang, C., & Chen, H. (2017). A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-

level permutation. *Optics and Lasers in Engineering, 90*, 238-246. doi:https://doi.org/10.1016/j.optlaseng.2016.10.020

Li, Z., Peng, C., Tan, W., & Li, L. (2020). A Novel Chaos-Based Color Image Encryption Scheme Using Bit-Level Permutation. *symmetry, 12*(9), 1497. doi:https://doi.org/10.3390/sym12091497

Muthu, J. S., & Murali, P. (2021). Review of Chaos Detection Techniques Performed on Chaotic Maps and Systems in Image Encryption. *SN Computer Science, 2*(5), 392. doi:10.1007/s42979-021-00778-3

Ramasamy, P., Ranganathan, V., Kadry, S., Damaševičius, R., & Blažauskas, T. (2019). An Image Encryption Scheme Based on Block Scrambling, Modified Zigzag Transformation and Key Generation Using Enhanced Logistic—Tent Map. *entropy, 21*(7). doi:10.3390/e21070656

Wang, X., & Chen, X. (2021). An image encryption algorithm based on dynamic row scrambling and Zigzag transformation. *Chaos, Solitons & Fractals, 147*, 110962. doi:https://doi.org/10.1016/j.chaos.2021.110962

Xingyuan, W., Junjian, Z., & Guanghui, C. (2019). An image encryption algorithm based on ZigZag transform and LL compound chaotic system. *Optics & Laser Technology, 119*, 105581. doi:https://doi.org/10.1016/j.optlastec.2019.105581

Zhang, X., & Gong, Z. (2022). Color image encryption algorithm based on 3D Zigzag transformation and view planes. *Multimedia Tools and Applications*. doi:10.1007/s11042-022-13003-x