A new approach for steganography: bit shifting operation of encrypted data in LSB (SED-LSB)

Araştırma Makalesi/Research Article

Serdar SOLAK, Umut ALTINIŞIK

Department of Informatics, Kocaeli University, Kocaeli, Turkey
serdars@kocaeli.edu.tr, umuta@kocaeli.edu.tr

(Geliş/Received:21.06.2018; Kabul/Accepted:28.01.2019)
DOI: 10.17671/gazibtd.435437

Abstract— In this article, a new method is presented by adding bit shift of encrypted data in LSB method (SED-LSB). The message to be embedded in the cover image is encrypted by the key received from the user. Then, secret data are obtained by bit shifting operation using this encrypted text. Finally, the secret data is hidden inside the cover image using the LSB method. SED-LSB is performed by preparing GUI in MATLAB. The quality of the cover image is measured by Peak Signal-to-Noise Ratio (PSNR), besides the similarity between the cover and stego image is determined by the Structural Similarity Index (SSIM). The 512 x 512 size of colored images as Lena, Peppers, Fruits and Baboon are used to hide 87373 byte data. PSNR values are between 51.615 – 51.656 and SSIM values are between 0.99937 – 0.99982. In the proposed method, when data are hidden at a higher capacity, stego image is found not only secure but also high quality.

Keywords— LSB, SED-LSB, Steganography, Cryptography, Security, PSNR, SSIM

Steganografi için yeni bir yaklaşım: (SED-LSB) LSB ile şifrelenmiş verilerin bit kaydırma işlemi

Özet— Makalede, veri gizlemede yaygın kullanılan LSB yöntemine bit kaydırma özelliği eklenerek yeni bir yöntem sunulmaktadır. Örtü görüntüye saklanacak olan metinler ilk olarak kullanıcıdan alınan anahtar sayesinde şifrelenmektedir. Bu işlemin ardından bit kaydırma özelliği kullanılarak gizli veri oluşturulmaktadır. Gizli veri, LSB yöntemi kullanılarak örtü görüntü içerisine gizlenmektedir. Çalışma MATLAB ortamında GUI hazırlanarak gerçekleştirilmiş olup, elde edilen PSNR ve SSIM değerleri makalede sunulmaktadır. 512 x 512 boyutlarında kullanılan Lena, Peppers, Fruits ve Baboon resimlerine 87373 bayt veri gizlenmiştir. PSNR değerlerinin 51,615 ile 51,656 arasında, SSIM değerlerinin ise 0,99937 ile 0,99982 arasında olduğu gözlenmiştir. Önerilen yöntem ile resimlerin içerisine yüksek kapasitede veriler güvenli bir şekilde gizlenmekte ve örtü görüntü ile stego görüntünün benzerliği yüksek çıkmaktadır.

Anahtar Kelimeler— LSB, SED-LSB, Steganografi, Kriptografi, Güvenlik, PSNR, SSIM

1. INTRODUCTION

Throughout history, the confidentiality and security of communication between people have become a very important factor. In recent years, the importance of information security has increased dramatically because digital communications are made over the Internet. Cryptography and Steganography are the science branches used for securing information security and confidentiality.

Cryptography is the whole of mathematical methods, such as identity control, integrity to ensure confidentiality and information security. In cryptography, it is necessary to use the same secret key to encrypt and decrypt [1-3]. In cryptology, public key systems [1,4], Symmetric-key [5-9] are widely used encryption techniques.

Steganography is a science, in which information is conveyed unnoticed by others. The message to be sent is hidden in a text, audio, image, video, or protocol file. Steganography consists of media carrier, secret data, secret key and stego media. Steganography performance is evaluated using the criteria of capacity, security, imperceptibility and computational complexity [10-13].

Image Steganography is divided into two basic classes as spatial and transform domain. Although Spatial domain has important advantages as high hiding capacity, less computational time and high manageable imperceptibility, vulnerable to geometric attacks. Transform domain is robustness for geometric attacks and compression, but high computational time, low capacity and limited manageable imperceptibility [13,14].

The Least Significant Bit (LSB) substitution [15], widely used in Steganography, is a method with less computational time and high-capacity data hiding. In this method, the least significant bits of the pixel channels of the image are changed. Wang et al. proposed genetic algorithm-based method to provide high data security and low computational time in LSB [16]. Chan and Cheng proposed an Optimal Pixel Adjustment Method (OPAM) to achieve high image quality and low computational time in LSB substitution method [17]. Muhammad et al. presented Stego Key-directed Adaptive Least Significant Bit (SKA-LSB) to increase security of cover image [18].

Kocak proposed a Couple Layered Security Model (CLSM) to hide high-capacity data using two-bit LSB method [19]. Aydoğan et al. presented a LSB-based method to hide the data with minimal changes to medical images [20].

In this paper, we proposed a new method SED-LSB, has high data security, less computational time, high data capacity and manageable imperceptibility. The rest of the article is organized as follows. In section 2, we present a new method SED-LSB. In Section 3, the proposed SED-LSB method is tested using a number of images. Finally, this article concludes in Section 4.

2. PROPOSED METHOD: SED-LSB

In this article, a more secure algorithm for data hiding is proposed by adding key generator and bit shift operation to the LSB method. The block diagram of the proposed SED-LSB is presented in Figure 1.

In the beginning, the message to be hidden is changed using the keyword, so the encrypted data is obtained. Encrypted data is passed through shifting operation process to obtain secret data. Finally, the secret data is hidden in the cover image with LSB method and stego image is created.

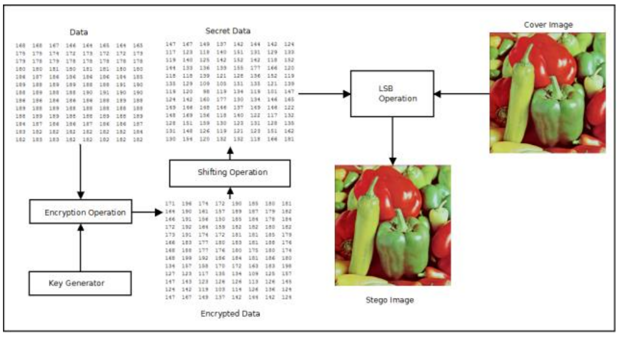


Figure 1. Block diagram of proposed SED-LSB method

2.1. Encryption Operation

In this section, encryption data is created by using the input data and the key value values generated by the key generator. The key received from user in text format is the input of the key generator. The text key consists of characters between ‘a’ and ‘z’. When a different character is entered, the key value cannot be obtained. Figure 2 shows the steps of the key generator process.

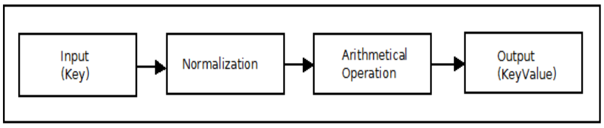


Figure 2. Block diagram of key generator

Normalization and arithmetical operation are applied respectively on the text key, and the key value output in binary format is obtained. In the normalization process, text key is converted to ASCII values, and then the letter ‘a’ is set to 1. The new values obtained at the end of this process are processed to the mathematical operations given in Equations 1, 2 and 3. The final value obtained is converted to the binary system and the key value is obtained.

$$S = \sum_{x=1}^{n} N_x \quad (1)$$

$$d = \max(N_1, N_n) - \min(N_1, N_n) \quad (2)$$

$$\mathrm{KeyValue} = \mathrm{ASCII2Binary}(\mathrm{mod}(S, d) * r) \quad (3)$$

In Equation 1, S is the sum of the normalized ASCII text key values, n is the text key size, and the normalized ASCII values of each character are called N_x. In Equation 2, d is the difference between the letter with the greatest and smallest ASCII value. In Equation 3, r is the number of characters repeated most in the text key. Figure 3 shows an example of obtaining the key value. In the example, key value (00000101) is obtained for 'pwd' text key value.

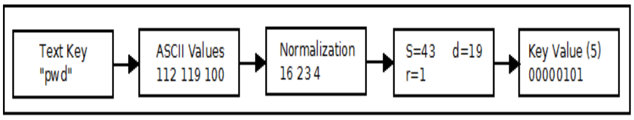


Figure 3. Key value creation process sample

The creation of the key generator is intended to make the cover text more secure. In addition, the data to be hidden in the cover image with the obtained value is subjected to XOR processing to obtain encrypted data. Figure 4 shows that the letter E and the key value are processed with XOR.

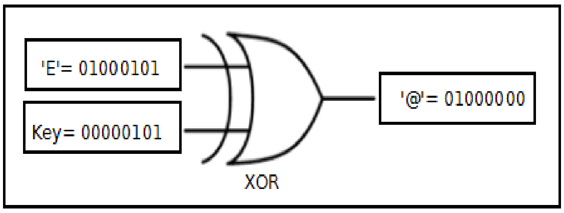


Figure 4. Encrypted data creation process sample

2.2. Shifting Operation

In this subsection, shifting is performed on the encrypted data. Thus, the encrypted data is changed again at the end of the shifting operation to provide extra security. Since shifting operation can be done at the desired amount, secret data is not figure out by someone else.

The proposed shifting operation is performed as binary data is moved from right to left by the desired number of bits. In the study, a four-bit shifting operation is carried out owing to perform inverse operation.

Figure 5 presents a sample application of the shifting operation. In the example, the '@' character is inverted to '♦' by inverse operation.

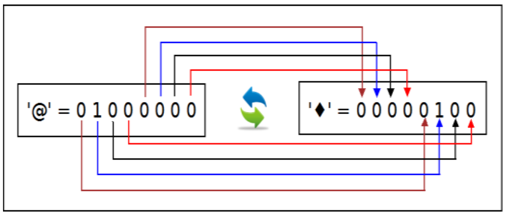


Figure 5. Shifting operation process sample

2.3. LSB Operation

LSB is one of the most common methods used in data hiding. Each pixel that produces a colored cover image has red, green, and blue (RGB) values. In the study, data hiding has been performed using three channels within the pixel. When data hiding is performed, the RGBBGRRG sequence is used. Three pixels are used in the process of hiding an eight-bit data. In the proposed study, LSB operation takes place in three steps. These steps are presented below.

1. Get pixel numeric values: The RGB data of each pixel value in the cover image is taken as decimal and converted to binary value.
2. Get secret data numeric values: It is the binary retrieval of the numeric or alphanumeric values that make up the secret data.
3. Hide secret data: It is at this stage that each character of the secret data is hidden in the cover image. This step is applied to perform until the all secret data is hidden. Data hiding occurs in three stages as follows.
    a. The binary value in the secret data is taken from the most significant bit to the least significant bit.
    b. If the received data is 0, the pixel value of the cover image and the binary value (11111110) are processed by the AND operation. At the end of this process, the pixel value of the cover image is updated.
    c. If the received data is 1, the pixel value of the cover image and the binary value (00000001) are processed by the OR operation. At the end of this process, the pixel value of the cover image is updated.

Figure 6 shows that the most significant three bits of 10100100 sample data is hiding on the one pixel of cover image by the SED-LSB method. The most significant bit of the sample is assigned to the R channel of the first pixel of the cover image, the most significant second bit is hidden in G, and the third bit is hidden in B channels. The bits 4, 5 and 6 in the data are respectively assigned to the channels B, G and R of the second pixel, 7 and 8 bits are hidden in the R and G channels of the third pixel.

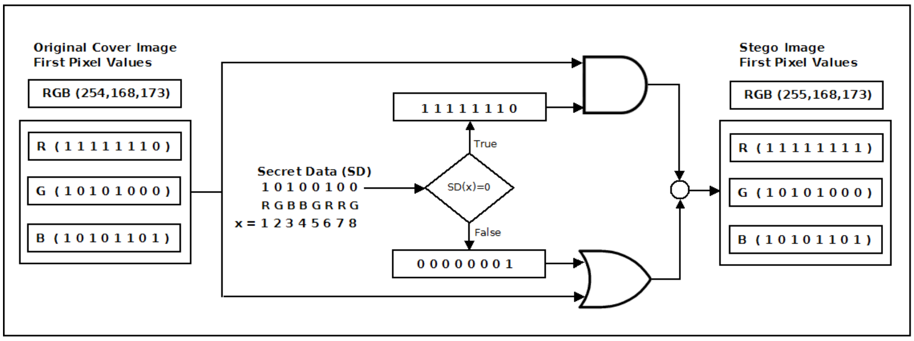


Figure 6. An example of hiding the most significant three bits of 10100100

3. EXPERIMENTAL STUDIES

The proposed study is performed in MATLAB, beside a program is prepared to provide graphical user interface (GUI). Experimental studies are performed in four stages using GUI. These steps are given as follows.

1. Image Capture (Cover Image): In this stage, the cover image is uploaded from the camera or file by using the GUI.

2. Text and Key Load: The user uploads the message that he wants to hide in the cover image by manual or file. In addition, users should write keyword to ensure the security of the data they want to hide. This keyword is also used in the decryption phase of the data.

3. Hide data with SED-LSB: In this phase, the loaded keyword is first converted into numerical data with the encryption operation. When the uploaded data is subjected to XOR operation using numeric key value, encryption data is obtained. Secret data is generated, as a result of applying encryption data to shifting operation. Secret data is hidden on cover image with SED-LSB method.

4. Analysis: This section provides various data about stego image. The PSNR [21] value is calculated to measure the quality of the cover image. The high PSNR value represents the stego image quality, while the low PSNR value indicates the high pixel difference between the cover and the stego image. The calculation of the PSNR value is given in Equation 4.

$$\text{PSNR} = 10\log_{10}\left(\frac{255^2}{\text{MSE}}\right) \tag{4}$$

SSIM shows the similarity rate of cover and stego image. The calculation of the SSIM value is shown in Equation 5 [22].

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x{}^2 + \mu_y{}^2 + c1)(\sigma_x{}^2 + \sigma_y{}^2 + c2)} \tag{5}$$

In the analysis phase of the study, the histogram comparing the cover image and the stego image, the size of the hidden message, the pixel change rates in each channel, PSNR and SSIM values are presented. Lena, Peppers, Fruits, Baboon 512 x 512 color images are used in experimental studies.

Figure 7 presents these images used in experimental studies. In Figure 8, a sample image of the GUI Steganography is presented.

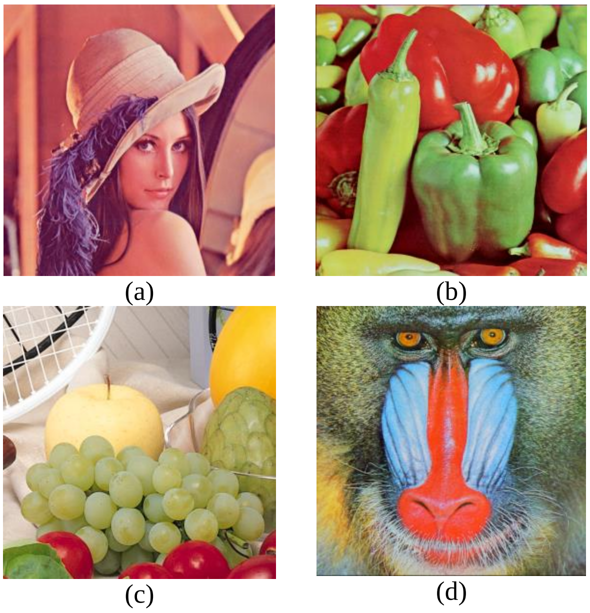
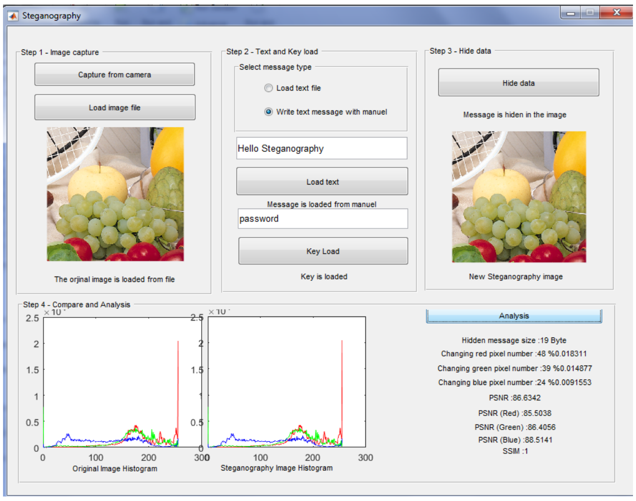

(a) (b)
(c) (d)

Figure 7. Images (a) Lena (b) Pepper (c) Fruits (d) Baboon



Hello Steganography

Message is loaded from manuel

password

Key is loaded

The orjinal image is loaded from file

Message is hiden in the image

New Steganography image

Hidden message size :19 Byte
Changing red pixel number :48 %0.018311
Changing green pixel number :39 %0.014877
Changing blue pixel number :24 %0.0091553
PSNR :86.6342
PSNR (Red) :85.5038
PSNR (Green) :86.4056
PSNR (Blue) :88.5141
SSIM :1

Figure 8. GUI for Steganography

In different sizes messages are hidden inside the cover images. Table 1 presents the values of the characters of "Hello Steganography" before they are hidden in the cover image.

The ascii and binary values of the characters of the message are given respectively in this table as result of original, encryption and shifting operations. The key value used in the encryption operation is obtained from the keyword "password".

As seen in Table 1, the message "Hello Steganography" is hidden in the cover image as “$÷ffVóò□÷ÍÂFVÍçÂğ&7”.

Table 1. An example conversion of "Hello Steganography" message into secret data

Letters	Original		Encryption Operation			Shifting Operation		
	ASCII	Binary	ASCII	Binary	Letter	ASCII	Binary	Letter
H	72	01001000	66	01000010	B	36	00100100	$
e	101	01100101	111	01101111	o	246	11110110	÷
l	108	01101100	102	01100110	f	102	01100110	f
o	111	01101111	101	01100101	e	86	01010110	V
space	32	00100000	42	00101010	*	162	10100010	ó
S	83	01010011	89	01011001	Y	149	10010101	ò
t	116	01110100	126	01111110	~	231	11100111	□
g	103	01100111	109	01101101	m	214	11010110	Í
a	97	01100001	107	01101011	k	182	10110110	Â
n	110	01101110	100	01100100	d	70	01000110	F
r	114	01110010	120	01111000	x	135	10000111	ç
p	112	01110000	122	01111010	z	167	10100111	ğ
h	104	01101000	98	01100010	b	38	00100110	&
y	121	01111001	115	01110011	s	55	00110111	7

Table 2 shows the PSNR and SSIM values calculated as a result of hiding 19-byte "Hello Steganography" message in different cover images. In this sample, the PSNR value of the cover images ranges from 86.015 to 87.545, also the SSIM values are calculated as 1.

Table 2. Simple message hidden (19 Byte Hello Steganography)

Cover Image (512*512)	PSNR				SSIM
	Red	Green	Blue	RGB	
Baboon	87.001	87.545	88.167	87.545	1.000
Friuts	85.504	86.406	88.514	86.634	1.000
Lena	85.982	87.001	88.699	87.087	1.000
Peppers	86.188	85.414	86.518	86.015	1.000

Table 3 presents results of the approximate maximum data (87373 bytes - 698984 bits) that the different cover images can be hidden using the proposed SED-LSB. The PSNR value of the cover images ranges from 51.615 to 51.656. According to the color channels, the PSNR values are 51.090 - 51.160 in the red, 51.073 – 51.162 in the green and 52.849 – 52.901 in the blue. The PSNR value of the blue channel is higher than the red and green channels, because fewer bits are hidden in the blue channel. Also, the SSIM values range from 0.99937 to 0.99982. The study shows that the similarity rate between the cover image and the stego image is high because the SSIM value is very close to 1 despite the maximum data hiding.

Table 3. Maximum text hidden 87373 Byte (698984 bit)

Cover Image (512*512)	PSNR				SSIM
	Red	Green	Blue	RGB	
Baboon	51.143	51.162	52.880	51.656	0.99973
Friuts	51.090	51.102	52.888	51.615	0.99937
Lena	51.131	51.161	52.901	51.656	0.99982
Peppers	51.160	51.073	52.849	51.621	0.99981

Figure 9 shows the PSNR values using the proposed method for Lena cover image. The embedding capacity of the proposed method is performed between 50.000 bits and maximum value with a step of 50.000 bits.

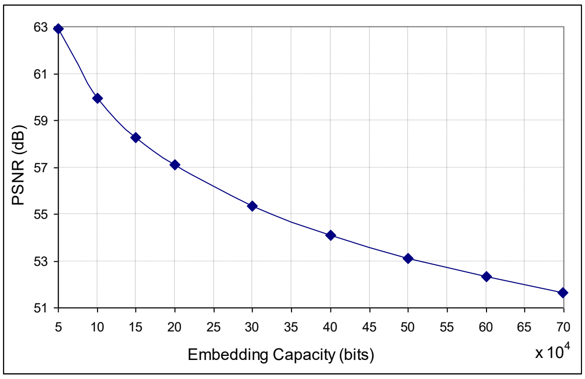


Figure 9. Performance of proposed method for Lena

Table 4 shows PSNRs of some studies in literature and our approach of 50.000 bits capacity. The PSNR values of SED-LSB, Sachnev et al. [23], Hong's [24] and Ou et al. [25] are given in the ranges 63.06-63.11, 50.33-59.26, 49.75-59.38 and 50.83-61.63 dB, respectively. Our approach PSNR values are found to give better results than these reversible data hiding methods.

Table 4. Comparison of proposed method and other methods [23-25]

Images	Sachnev et al. [23]	Hong [24]	Ou et al.[25]	Proposed SED-LSB Method
Lena	54.31	54.18	56.23	63.09
Baboon	50.33	49.75	50.83	63.11
Peppers	52.96	52.50	54.03	63.06
Airplane	57.49	58.67	60.13	63.11
House	59.26	59.38	61.63	63.09
Barbara	55.02	54.69	57.01	63.07

Table 5 shows PSNRs of Bhardwaj and Sharma [26] studies in literature and our approach for the different bits capacity. From the table, the PSNRs varies on the three 512 x 512 grayscale images of the proposed method. Our approach PSNR values give better results than varied LSBs data hiding methods.

Table 5. Comparison of proposed method and Bhardwaj and Sharma [26]

Methods	**4225 Bits**			**16384 Bits**			**24964 Bits**		
	Pepper	**Lena**	**Baboon**	**Pepper**	**Lena**	**Baboon**	**Pepper**	**Lena**	**Baboon**
Simple LSB	59.07	59.66	59.14	53.22	53.80	53.28	51.39	51.98	51.43
Random LSB	59.08	59.70	59.09	53.21	53.80	53.23	51.37	51.98	51.40
Inverted LSB	59.14	59.71	59.10	53.23	53.81	53.25	51.36	51.98	51.43
Comp. Inverted LSB	59.04	59.73	59.12	53.23	53.81	53.28	51.39	51.99	51.43
Proposed SED-LSB	69.04	69.03	69.08	63.16	63.19	63.21	61.33	61.35	61.37

4. CONCLUSION

In this paper, a secure and high capacity algorithm is proposed to hide data in the cover image. Key generators and shifting operations are added to the algorithm to achieve more effective data hiding security. In this case, a double layer security system is integrated into the LSB algorithm. The other contribution of the article is more effective data hiding capacity. For this purpose, three colors channel in each pixel are used to hide data different sequence as RGBBGRRG in this process. The sizes of 512 x 512 Lena, Peppers, Baboon, Fruit cover images are utilized in experimental studies. PSNR values are obtained between 51.615 and 51.656 to these standard cover images for maximum data hiding. In addition, SSIM values are found between 0.99937 and 0.99982 for maximum data. These values mean that the cover image and the stego image are similar.

REFERENCES

[1] W. Diffie, M. Hellman, “New directions in cryptography”, *IEEE transactions on Information Theory*, 22(6), 644-654, 1976.

[2] C. E. Shannon, “Communication theory of secrecy systems”, *Bell Labs Technical Journal*, 28(4), 656-715, 1949.

[3] N. Topaloğlu, M. H. Calp, B. Türk, “Bilgi Güvenliği Kapsamında Yeni Bir Veri Şifreleme Algoritması Tasarımı ve Gerçekleştirilmesi”, *Bilişim Teknolojileri Dergisi*, 9 (3), 291-301, 2016.

[4] A. Coşkun, Ü. Ülker, “Ulusal Bilgi Güvenliğine Yönelik Bir Kriptografi Algoritması Geliştirilmesi ve Harf Frekans Analizinine Karşı Güvenirlik Tespiti”, *Bilişim Teknolojileri Dergisi*, 6 (2), 31-39, 2013.

[5] V. Rijmen, J. Daemen, B. Preneel, A. Bosselaers, E. De Win, “The cipher SHARK”, **In International Workshop on Fast Software Encryption**, Springer, Berlin, Heidelberg, 99-111, February, 1996.

[6] R. Anderson, E. Biham, “Two practical and provably secure block ciphers: BEAR and LION”, **In International Workshop on Fast Software Encryption**, Springer, Berlin, Heidelberg, 113-120, February, 1996.

[7] M. T. Ahvanooey, Q. Li, J. Hou, H. D. Mazraeh, J. Zhang, “AITSteg: An Innovative Text Steganography Technique for Hidden Transmission of Text Message via Social Media”, *IEEE Access*, 6, 65981-65995, 2018.

[8] J. Daemen, V. Rijmen, “AES proposal: Rijndael”, 1996.

[9] J. R. Padilla-López, A. A. Chaaraoui, F. Flórez-Revuelta, “Visual privacy protection methods: A survey”, *Expert Systems with Applications*, *42*(9), 4177-4195, 2015.

[10] I. C. Lin, Y. B. Lin, C. M. Wang, “Hiding data in spatial domain images with distortion tolerance”, *Computer Standards & Interfaces*, 31(2), 458-464, 2009.

[11] S. Solak, U. Altınışık, “LSB Substitution and PVD performance analysis for image steganography”, *International Journal of Computer Sciences and Engineering*, 6(10), 1-4, 2018.

[12] S. Solak, U. Altınışık, “En Düşük Anlamlı Son Üç Bitin Değiştirilmesi Yöntemi Kullanılarak Renkli Görüntülere Verilerin Gizlenmesi”, **Uluslararası Marmara Fen ve Sosyal Bilimler Kongresi (IMASCON)**, 344-347, Kocaeli, Turkey, 2018.

[13] M. Hussain, A. W. A. Wahab, Y. I. B. Idris, A. T. Ho, K. H. Jung, “Image steganography in spatial domain: A survey”, *Signal Processing: Image Communication*, 65, 46-66, 2018.

[14] H. Al-Dmour, A. Al-Ani, “A steganography embedding method based on edge identification and XOR coding”, *Expert systems with Applications*, 46, 293-306, 2016.

[15] W. Bender, D. Gruhl, N. Morimoto, A. Lu, “Techniques for data hiding”, *IBM systems journal*, 35(3.4), 313-336, 1996.

[16] R. Z. Wang, C. F. Lin, J. C. Lin, “Image hiding by optimal LSB substitution and genetic algorithm”, *Pattern recognition*, 34(3), 671-683, 2001.

[17] C. K. Chan, L. M. Cheng, “Hiding data in images by simple LSB substitution”, *Pattern recognition*, 37(3), 469-474, 2004.

[18] K. Muhammad, J. Ahmad, N. U. Rehman, Z. Jan, M. Sajjad, “CISSKA-LSB: color image steganography using stego key-directed adaptive LSB substitution method”, *Multimedia Tools and Applications*, 76(6), 8597-8626, 2017.

[19] C. Kocak, “Clsm: Couple Layered Security Model A High-Capacity Data Hiding Scheme Using With Steganography”, *Image Analysis & Stereology*, 36(1), 15-23, 2017.

[20] T. Aydoğan, C. Bayilmiş, “A new efficient block matching data hiding method based on scanning order selection in medical images”, *Turkish Journal of Electrical Engineering & Computer Sciences*, 25(1), 461-473, 2017.

[21] A. Hore, D. Ziou, “Image quality metrics: PSNR vs. SSIM”, **In Pattern recognition (icpr) 20th international conference on IEEE**, 2366-2369, 2010.

[22] Z. Wang, E. P. Simoncelli, A. C. Bovik, “Multiscale structural similarity for image quality assessment”, **In Signals, Systems and Computers Conference Record of the Thirty-Seventh Asilomar Conference on**, 2, 1398-1402, 2004.

[23] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, Y. Q. Shi, “Reversible watermarking algorithm using sorting and prediction”, *IEEE Transactions on Circuits and Systems for Video Technology*, 19(7), 989-999, 2009.

[24] W. Hong, “Adaptive reversible data hiding method based on error energy control and histogram shifting”, *Optics Communications*, 285(2), 101–108, 2012.

[25] B. Ou, X. Li, Y. Zhao, R. Ni, “Efficient color image reversible data hiding based on channel-dependent payload partition and adaptive embedding”, *Signal Processing*, 108, 642-657, 2015.

[26] R. Bhardwaj, V. Sharma, “Image steganography based on complemented message and inverted bit LSB substitution”, *Procedia Computer Science*, 93, 832-838, 2016.