International Journal of Electrical and Computer Engineering (IJECE)
Vol. 12, No. 1, February 2022, pp. 358~364
ISSN: 2088-8708, DOI: 10.11591/ijece.v12i1.pp358-364

A combination of least significant bit and deflate compression for image steganography

Huda Kadhim Tayyeh[1], Ahmed Sabah Ahmed Al-Jumaili[2]
[1]Department of Informatics Systems Management (ISM), College of Business Informatics, University of Information Technology and Communications, Baghdad, Iraq
[2]Department of Bioinformatics (BI), College of Biomedical Informatics, University of Information Technology and Communications, Baghdad, Iraq

Article Info

Article history:

Received Mar 22, 2021
Revised May 29, 2021
Accepted Jun 12, 2021

Keywords:

Data compression
Deflate algorithm
Image steganography
LSB

ABSTRACT

Steganography is one of the cryptography techniques where secret information can be hidden through multimedia files such as images and videos. Steganography can offer a way of exchanging secret and encrypted information in an untypical mechanism where communicating parties can only interpret the secret message. The literature has shown a great interest in the least significant bit (LSB) technique which aims at embedding the secret message bits into the most insignificant bits of the image pixels. Although LSB showed a stable performance of image steganography yet, many works should be done on the message part. This paper aims to propose a combination of LSB and Deflate compression algorithm for image steganography. The proposed Deflate algorithm utilized both LZ77 and Huffman coding. After compressing the message text, LSB has been applied to embed the text within the cover image. Using benchmark images, the proposed method demonstrated an outperformance over the state of the art. This can proof the efficacy of using Deflate as a data compression prior to the LSB embedding.

Corresponding Author:

Huda Kadhim Tayyeh
Department of Informatics Systems Management (ISM), College of Business Informatics, University of Information Technology and Communications
Baghdad, Iraq
Email: haljobori@uoitc.edu.iq[1]; asabahj@uoitc.edu.iq[2]

1. INTRODUCTION

Information security represents a vital role in modern technologies. The evolution of data communications and networking has enabled a wide range of data transmission, exchange, and storage [1]. Such an opportunity for communications comes with various risks and vulnerabilities that might threaten information security. Thus, an imperative demand to secure such communications by providing authentic and confidential data access and exchange [2], [3]. Here comes the role of cryptography in which a secure channel for exchanging information can be provided effectively.

Steganography is one of the cryptography techniques where secret information can be hidden through multimedia files such as images and videos [4]. Steganography can offer a way of exchanging secret and encrypted information in an untypical mechanism where communicating parties can only interpret the secret message [5], [6]. Consider the case of image steganography, a cover image will be used to embed a secret message with a secret key. Anyone outside the communicating parties would have no clue to interpreting the secret message within the stego image [7]. However, the key success of any image steganography technique lies in its ability to minimize the variations among the cover and stego images [8].

Embedding the secret message to the cover image would impact its pixel representation. Hence, a higher variation between the original image (i.e., cover image) and the embedded image (i.e., stego image) would risk violating the secret message. In this regard, both secret message size and encryption would play an essential role in maintaining the security of the message [9]. This can be depicted where a higher size of secret message would expose the variations between the cover and stego images. As well, trivial message encryption would contribute toward exposing the original text of the secret message [10].

The literature showed great progress in proposing techniques for image steganography. In particular, least significant bit (LSB) was the most common technique for steganography, for instance, Ashwini and Komal [11] presented an LSB method combined with a linked list to embed a secret message within a color image. In addition, Kumar *et al.* [12] presented a combination of LSB and advanced encryption standard (AES) method for image steganography. The authors have carried out the AES encryption method on the secret message before the LSB embedding to improve the security. Chakraborty *et al.* [13] presented an enhanced edge detection technique for image steganography. The authors have utilized the proposed technique to identify the edge regions that would carry the secret message. Consequentially, the authors have embedded the secret message to such edge regions. Taouil and Ameur [14] presented a matching technique combined with the LSB to accommodate image steganography. The authors have utilized the traditional LSB to embed the secret message to a cover image. Then, the wavelet Faber-Schauder matching technique has been used to substitute the corresponding bits when retrieving the original secret message from the cover image. Similarly, Elmasry [15] attempted to manipulate the secret message before the LSB embedding. This has been depicted by utilizing an encryption method known as Fisher-Yates Shuffle. Such a method is intended to generate unbiased permutations of the secret message text. Ahmed *et al.* [16] exploited the image's color channels to perform the steganography. The authors have exploited the green channel to embed the secret message. Then, the most significant bits (MSB) has been used to determine the intended bits for message embedding.

On the other hand, some studies have exploited the meta-heuristic approaches to find near-optimal solutions for image steganography. For instance, Shukur and Jabbar [17] utilized particle swarm optimization (PSO) to identify optimal pixels that were intended to be embedded before applying the LSB. Similarly, Khan and Bianchi [18] proposed an ant colony algorithm to determine optimal pixels' regions to embed a secret message within a cover image. Once, the positions are being declared the LSB is applied to embed the secret message.

Thahab [19] utilized a compression approach to compress the secret message before the embedding through LSB. The authors used a compression method known as burrows-wheeler transform (BWT) to compress the secret message. Lastly, the LSB has been used to embed the secret message to the cover image. Duan *et al.* [20] presented a segmentation method through a special architecture of neural network (NN) known as U-Net architecture. Such a segmentation method has been utilized for identifying pixels' regions for secret message embedding. proposed a new plan to hide information depend on U-Net architecture. Qu *et al.* [21] examined the LSB for image steganography using microscopic particles of the image pixel (i.e., quantum level). Hence, a quantum bit of the cover image's pixel has been exploited to embed the secret message. Such a quantum bit is known as qubit thus, a least significant qubits (LSQ) has been presented in such a study. Lastly, Neamah *et al.* [22] presented an LSB steganography mechanism along with an XNOR gate. Such a gate has been used to exploit the image color channel as an encryption key.

As noticed from the literature, most of the attempts aim at determining the best path of pixel embedding or utilizing a manipulation mechanism over the secret message to improve the image steganography. However, there is still room for addressing a compression technique that might reduce the secret message size which indeed would lead to better results of steganography. Although LSB showed a stable performance of image steganography yet, many works should be done on the message part. This can be represented by addressing adequate text encryption and compression methods for minimizing the variations and protecting the security of the secret message.In this paper, the role of data compression will be examined along with the LSB technique in image steganography. This can be represented by proposing the Deflate algorithm with is a combination of LZ77 and Huffman coding algorithms. Consequentially, the LSB will be used to embed the compressed secret message in which an XNOR gate is being utilized to determine the color channel to be encoded. The paper is organized as; section 2 discusses the proposed data compression algorithm along with embedding the message through LSB. Section 3 analyzes the experimental results and establishes a comparison against the state-of-the-art. Lastly, section 4 provides a conclusion.

2. PROPOSED METHOD

The framework of the proposed method consists of multiple phases. First, both the cover image and the secret message would be prepared for the embedding process. In terms of the messaging aspect, the text will be undergone a compression task where the proposed Deflate algorithm is applied. Deflate algorithm

consists of two sub-algorithms including LZ77 and Huffman coding. On the other hand, the image's binary representation will be acquired where the least significant bit is utilized upon the color channel of the image via an XNOR gate. Lastly, the secret message will be embedded through LSB. Figure 1 shows the framework of the proposed method.

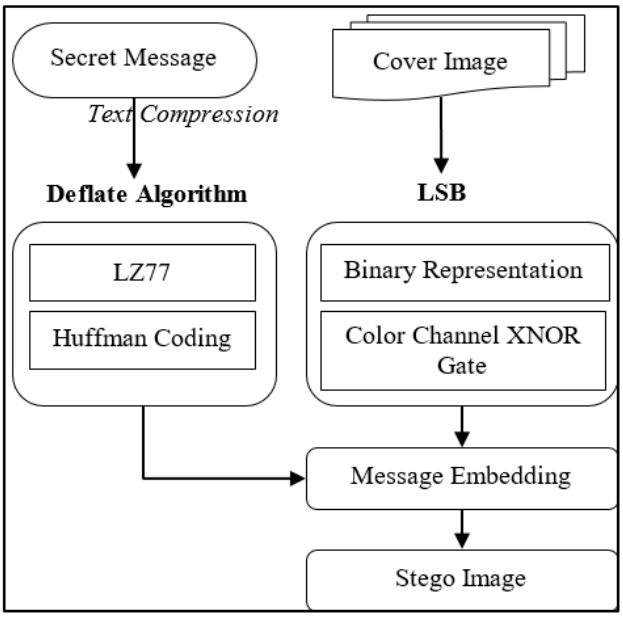

Figure 1. The framework of the proposed method

2.1. Cover image

For the images that will be used within the experiments, four benchmark images are being tackled including Airplane, Lina, Peppers, and Baboon. These images can be downloaded through (http://sipi.usc.edu/database/database.php?volume=misc). These images have been considered toward providing a consistent comparison against the baseline of Neamah *et al.* [22]. These images are colored with a size of 512×512.

2.2. Secret message

For the secret message, different sizes of text have been considered within the experiments including 4700 bits, 14500 bits, and 24250 bits. Again, these sizes have been considered toward providing a consistent comparison against the baseline of Neamah *et al.* [22].

2.3. Deflate compression algorithm

Deflate is a lossless compression algorithm that is composed of two main techniques including LZ77 and Huffman coding [23]. To address such an algorithm, it is necessary to illustrate the sub-techniques which can be depicted in the following subsections.

2.3.1. LZ77

LZ77 is a compression technique that belongs to the lossless algorithms where a dictionary-based mechanism is utilized to record any possible matches among the characters [24]. LZ77 contains three main parameters which determine the type and position of matches including offset, length, and codeword. Offset refers to the position of the previous match for a certain character/s. Whereas, length refers to the number of matches. Lastly, the codeword refers to the particular characters that have been matched. Let a text be as follow: *This message is secret.*

The encoding of such a text using LZ77 can be depicted in Table 1 where offset, length, and codeword along with the encoding output is illustrated. As shown in Table 1, the first seven characters (i.e., this me) reveal no matching therefore both codeword and length were set to zero. However, once a matching occurred in offset 8, the position of the previously matched character along with how many times the matching occurred would be represented as (4, 1) where the 's' character has been occurred at offset 4 and

happened for one time. In addition, at offset 14 a match for three characters has been depicted where the subset 'is' has occurred previously before 11 steps and the matching occurred three times thus, the output was (11, 3). The final output of encoding can be seen as:

$$(0,0)T(0,0)h(0,0)i(0,0)s(0,0)(0,0)m(0,0)e(4,2)(0,0)a(0,0)g(7,1)(5,1)(3,3)(4,1)(7,1)(0,0)$$
$$c(0,0)r(7,1)(0,0)t$$

To decode the LZ77 output, Table 2 depicts the process of retrieving the original characters.

Table 1. LZ77 encoding

Offset	Codeword	Length	Byte	Output
1	-	0	T	(0,0)T
2	-	0	h	(0,0)h
3	-	0	i	(0,0)i
4	-	0	s	(0,0)s
5	-	0	*blank*	(0,0)*blank*
6	-	0	m	(0,0)m
7	-	0	e	(0,0)e
8	s	1	-	(4,2)
9	-	0	a	(0,0)a
10	-	0	g	(0,0)g
11	e	1	-	(7,1)
12	*blank*	1	-	(5,1)
13	is *blank*		-	(3,3)
14	s		-	(4,1)
15	e		-	(7,1)
16	-		c	(0,0)c
17	-		r	(0,0)r
18	e		-	(7,1)
19	-		t	(0,0)t

Table 2. LZ77 decoding

Offset	LZ77 Encoding	Retrieved Byte	Stream
1	(0,0)T	T	T
2	(0,0)h	h	Th
3	(0,0)i	i	Thi
4	(0,0)s	s	This
5	(0,0)*blank*	*blank*	This_
6	(0,0)m	m	This m
7	(0,0)e	e	This me
8	(4,2)	ss	This mess
9	(0,0)a	a	This messa
10	(0,0)g	g	This messag
11	(7,1)	e	This message
12	(5,1)	*blank*	This message_
13	(3,3)	Is *blank*	This message is_
14	(4,1)	s	This message is s
15	(7,1)	e	This message is se
16	(0,0)c	c	This message is sec
17	(0,0)r	r	This message is secr
18	(7,1)	e	This message is secre
19	(0,0)t	t	This message is secret

2.3.2. Huffman coding

Huffman coding is another lossless compression algorithm that utilizes a special character frequency table and generated tree [25]. The frequency table aims at formulating the number of occurrences for each character within the text. Consequentially, the Huffman tree will be generated in which the characters are depicted as nodes with their frequencies. The smallest frequencies will start to generate edges until all the nodes are being connected. Consider the output of the LZ77 encoding which depicted as:

$$(0,0)T(0,0)h(0,0)i(0,0)s(0,0)(0,0)m(0,0)e(4,2)(0,0)a(0,0)g(7,1)(5,1)(3,3)(4,1)(7,1)(0,0)$$
$$c(0,0)r(7,1)(0,0)t$$

The Huffman table can be initiated by considering only the unique characters along with their frequencies as shown in Table 3. After that, the Huffman tree is generated in which the unique characters are forming the nodes [26], [27]. Then, the smallest frequency nodes will start to initiate the edges as shown in Figure 2. Hence, the left-hand edges will be marked as '0', while the right-hand edges will be marked as '1'. Therefore, if we consider the code for the character 'T', it will be '0000000'. Based on such a mechanism, the coding for each character will be represented in Table 4.

Table 3. Unique characters and frequency

Unique Characters	Frequency	Unique Characters	Frequency
0	24	h	1
(	19	i	1
)	19	s	1
,	19	*blank*	1
1	5	m	1
7	3	e	1
3	2	a	1
4	2	g	1
5	1	c	1
2	1	r	1
T	1	t	1

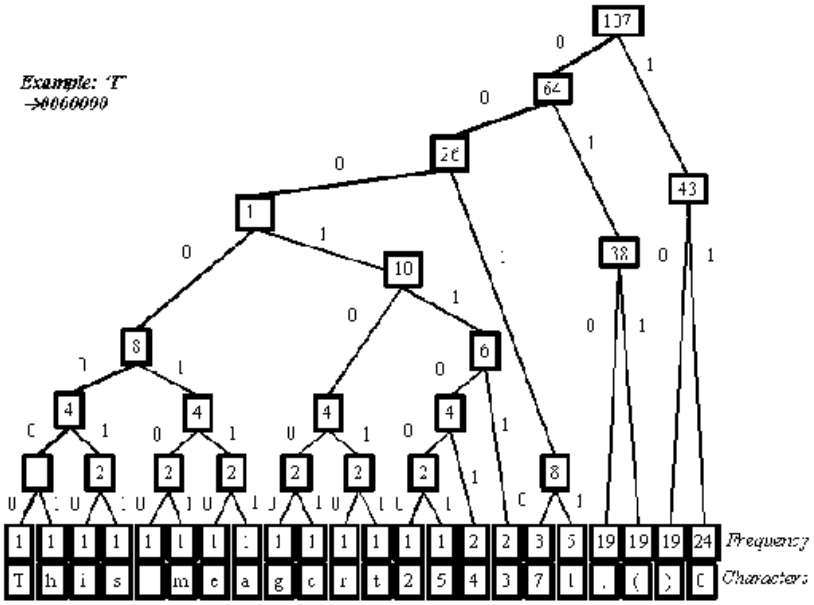

Figure 2. Huffman encoding

Table 4. Huffman coding

Character	Frequency	Coding
0	24	11
(	19	011
)	19	10
,	19	010
1	5	0011
7	3	0010
3	2	000111
4	2	0001101
5	1	00011001
2	1	00011000
T	1	0000000
h	1	0000001
i	1	0000010
s	1	0000011
blank	1	0000100
m	1	0000101
e	1	0000110
a	1	0000111
g	1	0001000
c	1	0001001
r	1	0001010
t	1	0001011

2.4. LSB message embedding

After compressing the secret message text, the cover image will be prepared for the message embedding process. For this purpose, the binary representation of each color channel within the image's pixels will be articulated. Hence, LSB will be applied where an XNOR gate is being used to determine whether a color channel is used for the embedding. The same mechanism in the baseline of Neamah *et al*. [22] is followed. This is depicted in which if the image pixel index mode 3 equals 0 then, the embedding will take place at the red channel, otherwise, if the image pixel index mode 3 equals 1 then, the green channel is selected. Finally, if the image pixel index mode 3 equals 2 then, the blue channel is selected.

3. RESULTS AND DISCUSSION

To evaluate the performance of the proposed steganography method, the two common metrics of mean-squared error (MSE) and the peak signal-to-noise ratio (PSNR) will be considered. MSE aims at calculating the total quadratic mistake between the cover image and the stego image. It can be computed as [28]:

$$MSE = \frac{1}{mn}\sum_{x=0}^{m-1}\sum_{y=0}^{n-1}[I(x,y) - K(x,y)]^2 \quad (1)$$

where m and n denote the height and width of the cover image and stego image respectively. Whereas, $I(x,y)$and $K(x,y)$ denote pixel values of the two images respectively. The low the value of MSE, the better steganography is acquired.On the other hand, PSNR aims at assessing the quality of an image statistically, it can be computed as in (2):

$$PSNR = 10\ log_{10}\frac{Max^2}{MSE} \quad (2)$$

where Max denotes the size of the image which is 255. The greater value of PSNR indicates better steganography. Based on the aforementioned metrics, the proposed method will be evaluated. In addition, a comparison of a baseline study of Neamah *et al.* [22] is considered. Table 5 depicts these results.

Table 5. Experimental results

Color Images with size 512×512	Secret text 4700bits				Secret text 14500bits				Secret text 24250bits			
	Baseline		Proposed		Baseline		Proposed		Baseline		Proposed	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
Airplane	0.0457	60.44	0.0016	76.03	0.0448	55.54	0.0046	71.49	0.0445	52.84	0.0067	69.82
Lina	0.0336	61.74	0.0017	75.81	0.0347	56.75	0.0046	71.47	0.0339	53.65	0.0067	69.84
Peppers	0.0879	57.65	0.0017	75.77	0.0865	42.84	0.0046	71.49	0.0862	39.99	0.0066	69.89
Baboon	0.0740	58.39	0.0017	75.80	0.0732	42.84	0.0044	71.60	0.0729	40.89	0.0066	69.90

As shown in Table 5, the results of both MSE and PSNR for all images showed an outperformance acquired by the proposed method over the baseline's results. This has been depicted for all text sizes in which the proposed method was able to get better results for both MSE and PSNR. Figure 3 depicts such an outperformance of the proposed method. As shown in Figure 3, the proposed method demonstrated a higher performance of steganography over the baseline study. This can prove the efficacy of using data compression especially through Deflate algorithm which utilizes two approaches of LZ77 and Huffman coding. The compression of the text leads to a significant reduction of text size which reflects higher PSNR and lowers MSE using LSB.

Figure 3. Outperformance of the proposed method

4. CONCLUSION

This paper proposed a combination of LSB and Deflate compression algorithm. Deflates has been able to reduce the message text size considerably through LZ77 and Huffman coding. After compressing the message text, the traditional LSB method has been used to embed the message through a cover image. Results of MSE and PSNR reveals an outperformance for the proposed method compared to the baseline. For future directions, examining the pixel path within the embedding to determine better regions would contribute toward improving both PSNR and MSE.

REFERENCES

[1] M. Sammour, B. Hussin, M. F. I. Othman, M. Doheir, B. AlShaikhdeeb, and M. S. Talib, "DNS tunneling: a review on features," *International Journal of Engineering and Technology*, vol. 7, no. 20, pp. 1-5, 2018, doi: 10.14419/ijet.v7i3.20.17266.

[2] M. N. Magableh, and B. Alshaikhdeeb, "A comparative study of encryption methods for cloud query processing," *Journal of Computer Science*, vol. 15, no. 11, pp. 1585-1594, 2019, doi:10.3844/jcssp.2019.1585.1594.

[3] H. K. Tayyeh, and A. S. A. Al-Jumaili, "Classifying confidential data using SVM for efficient cloud query processing," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 17, no. 6, pp. 3155-3160, 2019, doi: 10.12928/telkomnika.v17i6.13059.

[4] X. Duan, D. Guo, N. Liu, B. Li, M. Gou, and C. Qin, "A new high capacity image steganography method combined with image elliptic curve cryptography and deep neural network," *IEEE Access*, vol. 8, pp. 25777-25788, 2020, doi: 10.1109/ACCESS.2020.2971528.

[5] A. Gutub, and M. Al-Ghamdi, "Hiding shares by multimedia image steganography for optimized counting-based secret sharing," *Multimedia Tools and Applications*, vol. 79, pp. 7951-7985, 2020, doi: 10.1007/s11042-019-08427-x.

[6] H. K. Tayyeh, M. S. Mahdi, and A. A. AL-Jumaili, "Novel steganography scheme using Arabic text features in Holy Quran," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 3, pp. 1910-1918, 2019, doi: 10.11591/ijece.v9i3.pp1910-1918.

[7] A. S. Ansari, M. S. Mohammadi, and M. T. Parvez, "A multiple-format steganography algorithm for color images," *IEEE Access*, vol. 8, pp. 83926-83939, 2020, doi: 10.1109/ACCESS.2020.2991130.

[8] I. J. Kadhim, P. Premaratne, and P. J. Vial, "High capacity adaptive image steganography with cover region selection using dual-tree complex wavelet transform," *Cognitive Systems Research*, vol. 60, pp. 20-32, 2020, doi: 10.1016/j.cogsys.2019.11.002.

[9] W. Liu *et al.*, "Secure halftone image steganography with minimizing the distortion on pair swapping," *Signal Processing*, vol. 167, 2020, Art. no. 107287, doi: 10.1016/j.sigpro.2019.107287.

[10] Q. Li *et al.*, "A novel grayscale image steganography scheme based on chaos encryption and generative adversarial networks," *IEEE Access*, vol. 8, pp. 168166-168176, 2020, doi: 10.1109/ACCESS.2020.3021103.

[11] B. Ashwini and B. Komal, "Hybrid approach for embedding text or image in cover images," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 5, no. 5, 2016.

[12] M. Kumar, G. Yadav, A. K. Keshari, and S. Katiyar, "Image processing using steganography," *International Journal of Engineering Science and Computing*, vol. 7, pp. 10619-10624, 2017.

[13] S. Chakraborty, A. S. Jalal, and C. Bhatnagar, "LSB based non blind predictive edge adaptive image steganography," *Multimedia Tools and Applications*, vol. 76, no. 6, pp. 7973-7987, 2017, doi: 10.1007/s11042-016-3449-4.

[14] Y. Taouil, and E. B. Ameur, "Steganographic scheme based on message-cover matching," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 5, pp. 3594-3603, 2018, doi: 10.11591/ijece.v8i5.pp3594-3603.

[15] W. Elmasry, "New LSB-based colour image steganography method to enhance the efficiency in payload capacity, security and integrity check," *Sādhanā*, vol. 43, no. 5, pp. 1-14, 2018, Art. no. 68, doi: 10.1007/s12046-018-0848-4.

[16] S. Ahmed, R. Jaffari, and L. A. Thebo, "Data hiding using green channel as pixel value indicator," *International Journal of Image Processing (IJIP)*, vol. 12, no. 3, pp. 90-100, 2018.

[17] W. A. Shukur, and K. K. Jabbar, "Information hiding using LSB Technique based on developed PSO algorithm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 2, pp. 1156-1168, 2018, doi: 10.11591/ijece.v8i2.pp1156-1168.

[18] S. Khan, and T. Bianchi, "Ant colony optimization (ACO) based data hiding in image complex region," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 1, pp. 379-389, 2018, doi: 10.11591/ijece.v8i1.pp379-389.

[19] A. T. Thahab, "A secure image steganography based on burrows wheeler transform and dynamic bit embedding," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, pp. 460-467, 2019, doi: 10.11591/ijece.v9i1.pp460-467.

[20] X. Duan, K. Jia, B. Li, D. Guo, E. Zhang, and C. Qin, "Reversible image steganography scheme based on a U-Net structure," *IEEE Access*, vol. 7, pp. 9314-9323, 2019, doi: 10.1109/ACCESS.2019.2891247.

[21] Z. Qu, Z. Cheng, and X. Wang, "Matrix coding-based quantum image steganography algorithm," *IEEE Access*, vol. 7, pp. 35684-35698, 2019, doi: 10.1109/ACCESS.2019.2894295.

[22] R. M. Neamah, J. A. Abed, and E. A. Abbood, "Hide text depending on the three channels of pixels in color images using the modified LSB algorithm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 1, pp. 809-815, 2020, doi: 10.11591/ijece.v10i1.pp809-815.

[23] S. Oswal, A. Singh, and K. Kumari, "Deflate compression algorithm," *International Journal of Engineering Research and General Science*, vol. 4, no. 1, pp. 430-436, 2016.

[24] A. S. Sitio, "Text message compression analysis using the LZ77 algorithm," *INFOKUM*, vol. 7, no. 1, pp. 16-21, 2018.

[25] A. Moffat, "Huffman coding," *ACM Computing Surveys (CSUR*), vol. 52, no. 4, pp. 1-35, 2019, Art. no. 85, doi: 10.1145/3342555.

[26] A. S. A. AL-Jumaili, and H. K. Tayyeh, "A hybrid method of linguistic and statistical features for arabic sentiment analysis," *Baghdad Science Journal*, vol. 17, no. 1, pp. 385-390, 2020, doi: 10.21123/bsj.2020.17.1(Suppl.).0385.

[27] A. S. AL-Jumaili, H. K. Tayyeh, and R. J. Kadhem, "A backpropagation neural network for splitting identifiers," *Journal of Computer Science*, vol. 14, no. 10, pp. 1412-1419, 2019, doi: 10.3844/jcssp.2018.1412.1419.

[28] A. K. Sahu, and G. Swain, "Reversible image steganography using dual-layer LSB matching," *Sensing and Imaging*, vol. 21, pp. 1-21, 2020, Art. no. 1, doi: 10.1007/s11220-019-0262-y.