Intelligent Automation & Soft Computing

Tech Science Press

DOI: 10.32604/iasc.2023.032084

ARTICLE

Coverless Image Steganography System Based on Maze Game Generation

Al Hussien Seddik Saad[1], Mohammed S. Reda[2], Gamal M. Behery[2], Ahmed A. El-harby[2], Mohammed Baz[3], Mohamed Abouhawwash[4,5,*] and Ahmed Ismail Ebada[6]

[1]Department of computer science, Faculty of Science, Minia University, Shalaby, Minia, Egypt

[2]Department of computer science, Faculty of Computers and Artificial Intelligence, Damietta University, New Damietta, Damietta, Egypt

[3]Department of Computer Engineering, College of Computer and Information Technology, Taif University, P. O. Box 11099, Taif, 21994, Saudi Arabia

[4]Department of Mathematics, Faculty of Science, Mansoura University, Mansoura, 35516, Egypt

[5]Department of Computational Mathematics, Science, and Engineering (CMSE), College of Engineering, Michigan State University, East Lansing, MI, 48824, USA

[6]Department of Information Systems, Faculty of Computers and Artificial Intelligence, Damietta University, New Damietta, Damietta, Egypt

*Corresponding Author: Mohamed Abouhawwash. Email: abouhaww@msu.edu

Received: 06 May 2022 Accepted: 29 June 2022 Published: 05 February 2024

ABSTRACT

The trend of digital information transformation has become a topic of interest. Many data are threatening; thus, protecting such data from attackers is considered an essential process. Recently, a new methodology for data concealing has been suggested by researchers called coverless steganography. Coverless steganography can be accomplished either by building an image database to match its image subblocks with the secret message to obtain the stego image or by generating an image. This paper proposes a coverless image steganography system based on pure image generation using secret message bits with a capacity higher than the other traditional systems. The system uses the secret message to generate the stego image in the form of one of the Intelligence Quotient (IQ) games, the maze. Firstly, a full grid is generated with several specific rows and columns determined from the number of bits of the secret message. Then, these bits are fed to the full grid to form the maze game stego image. Finally, the generated maze game stego image is sent to the recipient. The experimental results, using the Bit Error Rate (BER), were conducted, and confirmed the strength of this system represented by a high capacity, perfect performance, robustness, and stronger hiding system compared with existing coverless steganography systems.

KEYWORDS

Coverless data hiding; digital image steganography; intelligence quotient games; maze game

1 Introduction

The technology of invisibility of secret information during its transmission in a hidden manner has been discussed by researchers. Due to the extended utilization of digital data (images, audios, and

videos) in practical daily life and the exchange of these data through different social media platforms. The transmission of such secret sensitive data in a digital form is urgently needed a security system to protect these data from being pirated. Recently, a new terminology has appeared on the scene called coverless steganography. Coverless does not necessarily mean that a cover may be used to accomplish the data hiding process [1]. The problem here is that the attackers can easily modify, destroy, or add a new bitstream to the original secret message, this requires designing a new robust steganography system to protect the secret message bits and deliver this message to the receiver.

Meng et al. [2] proposed an algorithm for coverless steganography based on direct current coefficients. A video was used as a cover file to conceal the transmitted data. A process of coding the video using a gaussian distribution model had been developed to investigate the changes of these coefficients by establishing a hash function to accelerate searching for videos with index structure. This process matched the hash sequences of the secret message and videos based on their index structure.

Edhah et al. [3] developed a coverless image steganography scheme using invariant feature transform and a bag of features different from other methods. Matching between secret messages and natural images was used. These natural images which contain the secret message with an inverted index were sent to the recipient to retrieve the concealed secret message. This method does not modify the original natural images.

Zhang et al. [4] used discrete cosine transforms and latent Dirichlet allocation to build a coverless image steganography algorithm. Images were classified according to their topic. Then, these images were selected to apply discrete cosine transforms with a block size of 8×8. Consequently, a feature sequence was obtained through the relationship between coefficients of the next blocks. After that, feature sequence coordinate locations and image paths were represented with an inverted index. Furthermore, the matching process was done to examine whether a feature sequence of the secret message and selected images were identical or not based on its index.

Zhou et al. [5] retrieved a set of partial duplicates of a given image selected from a natural images database instead of changing the cover image pixels' intensities values. Each image in the database was segmented into a set of non-overlapping blocks indexed according to their features. The stego image was obtained by matching similar blocks with the secret message.

Govindasamy et al. [6] employed Haar wavelet transforms to improve the payload capacity by building a coverless image steganography system. An image of size 256×256 was divided into submatrices of size 1024. Then, for each submatrix, coefficients were created using the wavelet transform. These coefficients were transformed into binary values to compare these values with the current value to know which value is greater than others to obtain an array of size 67032. The secret data were divided into bytes to test them with the block and starting index of the array to obtain the stego image.

Tan et al. [7] used video motion analysis to develop a coverless steganography technique. Robust histograms of oriented optical flow were generated for all videos which were indexed. These indices and hash sequences of robust histograms were transmitted to the recipient as a mapping. The hash sequences of robust histograms were computed from the sent video to retrieve the secret message. All used videos as covers did not lose any of their contents through the transmission and retrieval processes.

After showing some existing techniques and methods of coverless steganography and their characteristics, it was clear that some of them had the following shortcomings:

- Some current steganography techniques were vulnerable to image processing assaults because they extracted characteristics in the spatial domain [4].
- Some of the current steganography techniques had low robustness and security [4].
- Almost all of them had low embedding capacity [8].
- Several images were necessary to depict the hidden information [4,8].
- A large image database was required. This database was scanned for images encoding secret message bits similar to image retrieval [9,10].
- Cover images were selected randomly resulting in a substantial disparity in the contents of these photographs. It could raise suspicions and significantly decrease the safety level of coverless steganography [4].

The main contribution of this paper is to propose a new coverless image steganography system that is based on maze game image generation driven by secret messages. The cover image pixels' values have not been altered as LSB; instead, it generates a maze game image based on the secret message bits.

The paper is organized as follows: Section 2 shows the proposed method. Section 3 shows the performance evaluation using the bit error rate metric. Section 4 deducts the experimental results and comparisons. Finally, Section 5 presents the conclusion.

2 The Proposed Method

The proposed system has two terminals that communicate together through any software application as a communication channel. The first terminal is called the transmitter while the second terminal is the recipient. The main problem that faces transferring secret information between different terminals is how this information is protected and secured from penetrating, altering, or stealing. Thus, a secure coverless steganography system based on a maze game generation has been presented to transfer this information in a secured form. Moreover, the proposed system secures the secret message as a trusted natural image by sending it through any communication channel. Also, the system solves the problems raised previously by existing systems. The system consists of two main components. The first component is the embedding algorithm which is used to generate the maze game stego image (MGSI) from the secret message, see Section 2.2. The second component is the extraction algorithm that is used to retrieve the transmitted secret message from MGSI, see Section 2.3. All details of this system will be discussed in depth in the following subsections.

2.1 Maze Definition, Construction, and Generation

A maze is a path or collection of paths, typically from an entrance to a goal. The word is used to refer both to branching tour puzzles through which the solver must find a route. Mazes have been built with walls and rooms. Maze generation is the act of designing the layout of passages and walls within a maze. There are many different approaches to generating mazes, with various maze generation algorithms for building them, either by hand or automatically by computer.

There are two main mechanisms used to generate mazes. In "carving passages", one marks out the network of available routes. In building a maze by "adding walls", one lays out a set of obstructions within an open area. Most mazes drawn on paper are done by drawing the walls, with the spaces in between the markings composing the passages [11]. The maze of the proposed system consists of a group of vertical and horizontal walls. Fig. 1 shows the maze structure of size 11 rows and 12 columns consisting of visible walls (vertical and horizontal). Ones and zeros are represented by visible and

invisible walls (vertically and horizontally). These walls (bits) are arranged randomly on the maze according to the secret message bits.

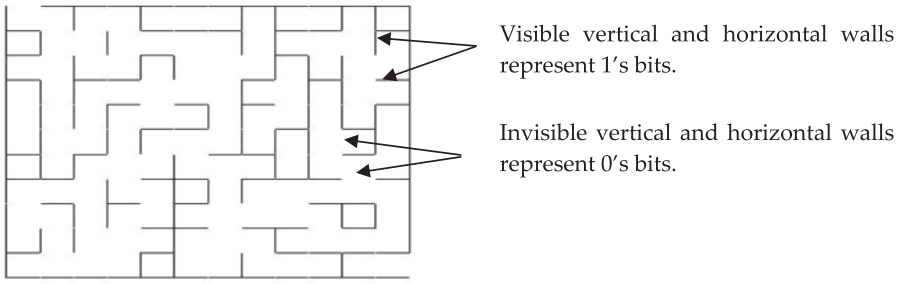

Figure 1: Structure of the generated maze image

2.2 The Embedding Algorithm

As mentioned above, this system generates a maze game using only the secret bits. The embedding process and pseudocode are described in detail below.

2.2.1 Embedding Process

This subsection explains how the first component of the system works. Only the secret message is inserted as an input to this system (See Embedding Pseudocode). This message is translated into a stream of binary (i.e., zeros and ones). After that, a grid of n rows and m columns will be created from the number of bits of the inserted secret message. Then, these bits are fed to the generated grid in sequential order for all rows and columns until all secret bits are finished. If the target bit is 0, remove the current wall; else, leave the current wall as it is in the grid. This procedure is repeated until all target bits are represented by visible or invisible walls. Finally, the maze image (i.e., the stego image) is generated and sent to the recipient. Fig. 2 illustrates an embedding structure for a real example to embed the bits "010100110111 . . . " and create MGSI.

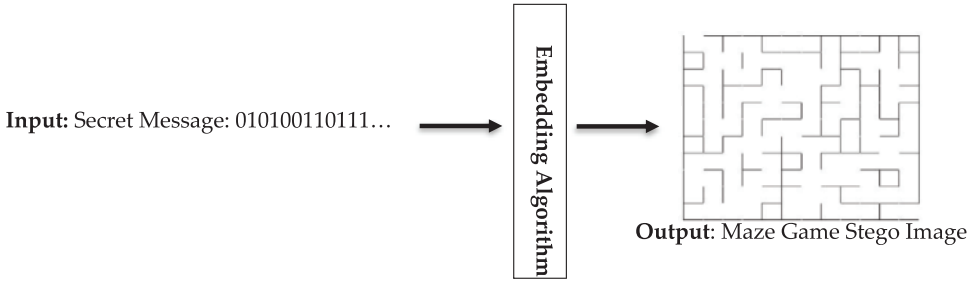

Figure 2: Embedding framework structure

2.2.2 Embedding Pseudocode

The first component of the proposed system is embedding which focuses on the mechanism of concealing the secret data. The input secret message is separated into characters, Sec_Mess = {sm_1, sm_2, sm_3, . . . , sm_n}. Then, bits are derived from these characters, SMB = {smb_1, smb_2, smb_3,..., smb_n}. After that, a grid of size nxm, where n is the number of rows and m is the number of columns, is created

according to the number of secret message bits. This grid is used to conceal the secret message inside it and obtain the maze image. Finally, two stages are constructed to hide the secret message bits. The first stage traverses the grid rows in sequential order, row after another starting from the first row. Zeros' bits are represented by removing current vertical walls to be invisible while ones' bits are represented by the existing vertical walls without change. The second stage is similar to the first stage but it uses the columns of the grid instead of rows for traversing. The result of the two stages produces MGSI which is sent to the receiver, as seen in Algorithm 1.

Algorithm 1: Secret Message Embedding Algorithm

BEGIN
Input: Secret_Message (Sec_Mess).
Output: Maze_Game_Stego_Image (MGSI).
For k=1 to length(SMB)
//The first stage.
  For i =1 to n
    For j =1 to m
      If (smb_k == 0)
        Remove the current wall, wall (i, j), from the grid to be an invisible wall.
      Elseif (smb_k == 1)
        Leave the current wall to be a visible wall.
      End if
    End for
  End for
//The second stage.
  For i2 =1 to m
    For j2 =1 to n
      If (smb_k == 0)
        Remove the current wall, wall (i2, j2), from the grid to be an invisible wall.
      Elseif (smb_k == 1)
        Leave the current wall to be the visible wall.
      End if
    End for
  End for
End for
Save MGSI.
END

2.3 The Extraction Algorithm

The following subsections describe in detail how the secret message is extracted from the sent MSGI.

2.3.1 Extraction Process

After the maze game stego image is delivered to the receiver through the communication channel. This image is inputted into the system to retrieve the secret message from it (see Algorithm 2). The maze game stego image was firstly scanned row by row and column by column to identify all vertical and horizontal walls, respectively. After that, these bits (walls) will be ordered, organized, and sorted

together to produce the bitstream. Finally, these sorted bits will be separated to form bytes, translated into characters, and joined together to obtain the secret message (Sec_Mess).

2.3.2 Secret Message Extraction Pseudocode

This algorithm is used to retrieve the secret message. The MGSI is transformed into black/white image, BW_MGSI. A cropping operation is done for the maze image by removing all white spaces surrounding the maze. Then, a grid of size nxm is created. This grid is scanned to locate the positions of all horizontal and vertical walls and save them in two arrays. They are called Rows_Pos and Cols_Pos and are used as mapping legends for the retrieval process. Finally, the algorithm for retrieving the secret message consists of two stages. The first stage scans the maze image rows in sequential order, row by row, starting from the first row to identify and detect all visible and invisible vertical walls. The second stage is similar to the first stage but it uses the columns of the maze image instead of rows for scanning to identify and detect all visible and invisible horizontal walls. The bitstream obtained from the second stage is concatenated to the bitstreams of the first stage to achieve the target of the extraction process for retrieving the secret message, as presented in Algorithm 2.

Algorithm 2: Secret Message Extraction Algorithm

BEGIN
Input: Maze_Game_Stego_Image (MGSI).
Output: Secret_Message (Sec_Mess).
//The first stage.
For i = 1 to n
  For j = 1 to m
    If (BW_MGSI (r_i, $Cols\_Pos_j$) == 0)
      Sec_Mess_Bits=Sec_Mess_Bits + ‘1’
    Else if (BW_MGSI (r_i, $Cols\_Pos_j$) == 1)
      Sec_Mess_Bits = Sec_Mess_Bits + ‘0’
    End if
  End for
End for
//The scond stage.
For i2 = 1 to m
  For j2 = 1 to n
    If (BW_MGSI ($Rows\_Pos_{j2}$, c_{i2}) == 0)
      Sec_Mess_Bits = Sec_Mess_Bits + ‘1’
    Else if (BW_MGSI ($Rows\_Pos_{j2}$, c_{i2}) == 1)
      Sec_Mess_Bits = Sec_Mess_Bits + ‘1’
    End if
  End for
End for
Return the secret message “Sec_Mess”.
END

3 Performance Evaluation

BER is the most important metric for evaluating the performance of the coverless steganography system because it measures the similarity between the original secret message and the retrieved bits, bit by bit. The error percentage of the retrieved message is calculated. Different attacks may detect system failure and make the system work with a small shortage. These attacks may be image scaling or image format changing. The system evaluation is conducted to know to the extent the system is successful. Eq. (1) defines the BER measurement [12].

$$\text{BER} = \frac{e}{n}, \text{e} = \sum_{i=1}^{n} pi \oplus qi \tag{1}$$

where e, n, p, and q are the number of detected errors, the total number of bits of the original secret message, original bitstream, and retrieved bitstream, respectively. The relationship between BER value and system quality is an inverse relation. A zero value of BER is an indication that there are no errors; otherwise, it means that some or all of the retrieved bits have been modified during the extraction process and the system does not achieve satisfactory success in that attack.

4 Results and Discussion

The system efficiency is checked when the system is executed and verified. The results of other coverless steganography systems are compared. MATLAB was used to build the system and obtain results [4,5,9]. All experiments are applied on a capacity of 1736 bits.

4.1 Embedding Capacity

Table 1 and Fig. 3 show that the proposed system achieved the highest embedding capacity compared with other steganography methods. The system payload hiding capacity is 1736 bits as shown in Fig. 4. This indicates that this system has improved the capacity problem mentioned in Section 1.

Table 1: The embedding capacity comparison

Method	Embedding capacity (bits)
Iris image [13]	8
Anime characters [14]	14
DCT and LDA [4]	1∼15
Bag of words [15]	16
Image hashing [9]	18
Motion analysis of videos [7]	32
Molecular structure images of materials [8]	36
Dynamic content selection [16]	68
Average pixel value [17]	80
Partial duplicate image retrieval [5]	384
Jigsaw puzzle [18]	760
Image block matching and dense convolutional network [19]	800
The proposed system	1736

As shown in the previous table, the proposed system achieved better embedding capacity than other coverless systems. This means that the system enhances the challenge of the limited capacity. The system can send and receive up to 10784 bits accurately 100% in a secure form without losing the bits during the extraction process.

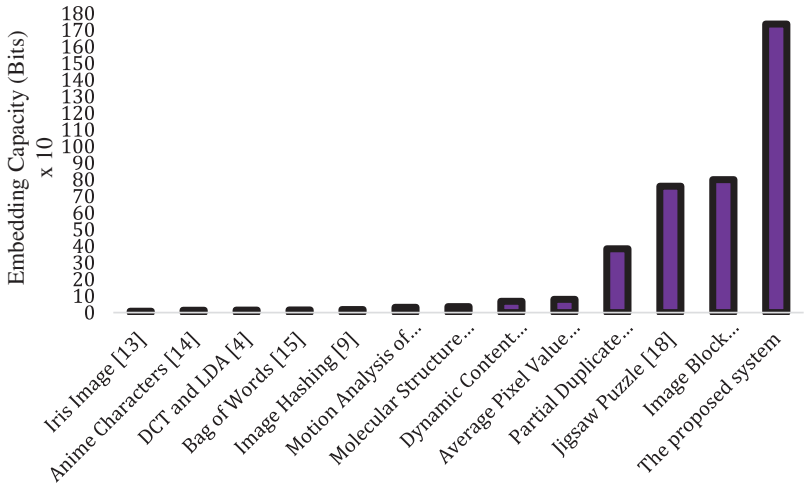

Figure 3: Embedding capacity compared with other methods

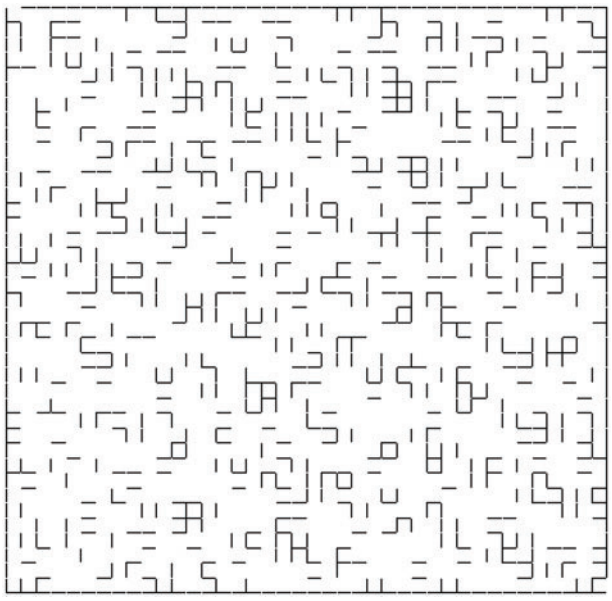

Figure 4: Generated MGSI with embedding capacity of 1736 bits

Table 2 and Fig. 5 show the comparison with other systems for embedding different secret messages sizes. These sizes are 1, 10, 100, and 1000 bytes.

As shown, the proposed system needs one image to send different data sizes but roughly five images are required to send a message of size 1000 bytes. This shows the level of efficiency of the system compared with other systems.

4.2 System Durability

Suppose that the attackers have noticed that there is a communication process between two peers. In that case, they could not read the secret message from the sent MGSI image as it is represented in the form of one of the Intelligence Quotient games, i.e., the maze. The generated MGSI does not have any fixed shapes. It is considered blind; therefore, there are no changes in the image pixels, confirming that the proposed system achieves a high level of security compared with other systems that use different secret message concealing procedures [10]. Instead of changing the values of the selected image pixels to insert the payload, MGSI is generated according to the bits of the sensitive transmitted message directly. The created MGSIs are suitable for the secret message representation as they are well-known games.

Table 2: The number of required images for embedding different sizes of data

Method	Secret message length			
	Byte	10 Byte	100 Byte	1000 Byte
Steganography without embedding [10]	1	10	100	1024
SIFT and BoF [20]	1	10	100	1024
DCT and LDA [4]	2∼9	7∼81	55∼801	548∼8193
Image hashing [9]	2	6	46	457
Motion analysis of videos [7]	1	3	25	256
Jigsaw puzzle [18]	1	1	1.05	10.7
The proposed system	1	1	1	4.6

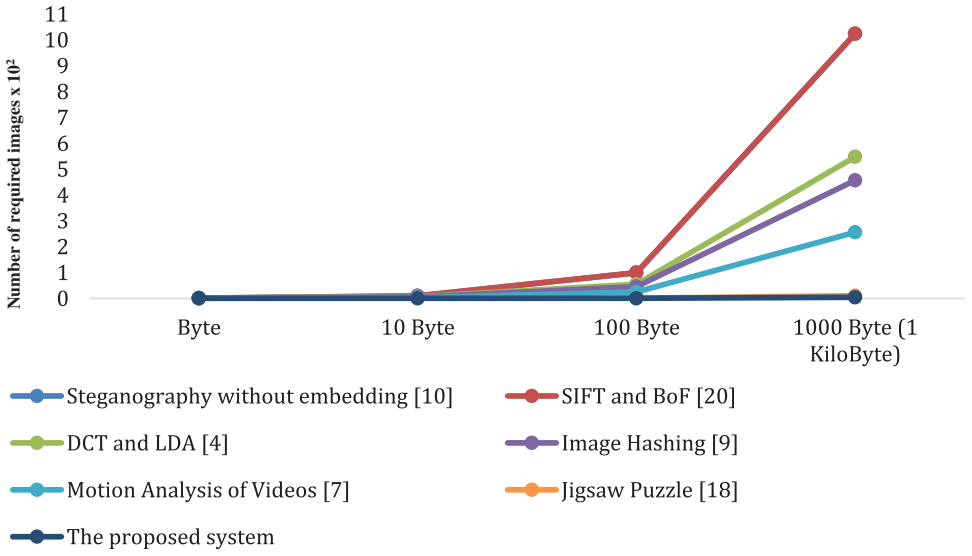

Figure 5: The number of needed images to hide different sizes of data

4.2.1 Image Scaling Attack

Image scaling plays a significant role as an important attack because it can affect the image, i.e., hidden data. BER values for image scaling attacks are compared with others at different scaling ratios are demonstrated. The obtained results confirmed that when the image was scaled with a ratio of 30%, a system failure occurred, and the retrieved data were distorted. The reason was that the maze was hazy at this ratio and the system could not identify the representation of the wall. However, 10 and 0.0058 were the number of incorrect retrieved bits and their BER value, respectively, at a 50% image scaling ratio. As a result, 100% efficiency was obtained at a 75% image scaling ratio and higher up to 150%. Thus, this is considered a great success for this system to escape from scaling attacks, as shown in Table 3.

Table 3: BER values for image scaling attack comparison

Image scaling ratio	DCT and LDA [4]	Grayscale gradient co-occurrence matrix [12]	The proposed system
30%	0.146	0.015	Failed
50%	0.057	0.009	0.0058
75%	0.039	0.002	0
150%	0.016	0.025	0

As shown in the previous table, a zero BER value was obtained when the system was tested at scaling ratios 75%n and 150%, this is an indication for better performance among all almost other systems. On the other hand, the system achieved 99.9942% as a success rate while the system could not recover the original secret message correctly.

4.2.2 JPEG Compression Attack

Many systems transfer the sent image into JPEG. Lossy compression allows data to be lost during the transmission process [12]. MSGI loses some/all of its data if the compression has been performed. After the JPEG compression attack, BER has been inducted into the system at different image qualities ranging from 90% to 50%. To make a meaningful comparison with others. It was found that they applied the system to the following image compression qualities: 90%, 80%, 70%, 60%, and 50%. The proposed system was applied to an obtained image called MGSI (generated in subsection 2.2.2) using the above qualities.

Table 4 and Fig. 6 present a comparison among the proposed system and other systems using different image qualities. The system produced zero BER for all used image qualities of the MGSI to achieve the best performance. This achievement enables this system to not lie in the trap of that attack, i.e., the message was retrieved with 100% accuracy. The tested MGSI was off. PNG format and its original size were 62 kilobytes.

4.2.3 Image Rotation Attack

Image rotation plays an important role as an attacker toward the system which may effect on recovery of the secret message. Rotating the image by θ degrees counter-clockwise may affect the shape of the maze inside the image. Rotation is 2D transform mapping the point of coordinate (x, y) to a new point (x′, y′) [21]. The proposed system was designed to set the maze shape in its normal shape

inside the image before extracting the secret message from the maze game image, so the system was able to escape from that attack achieving zero BER value.

Table 4: BER of JPEG image compression attack comparison

Image qualities	Chaos based DCT [12]	Chaos based zero steganography [12]	Chaos sequences and DCT [12]	Coverless information hiding based on robust image hashing [12]	Grayscale gradient co-occurrence matrix [12]	The proposed system
90%	2.2%	4.8%	0.2%	0%	0%	0%
80%	–	–	–	–	–	0%
70%	3.8%	8%	0.9%	8%	0.2%	0%
60%	–	–	–	–	–	0%
50%	15.1%	9.8%	14.6%	–	0.7%	0%

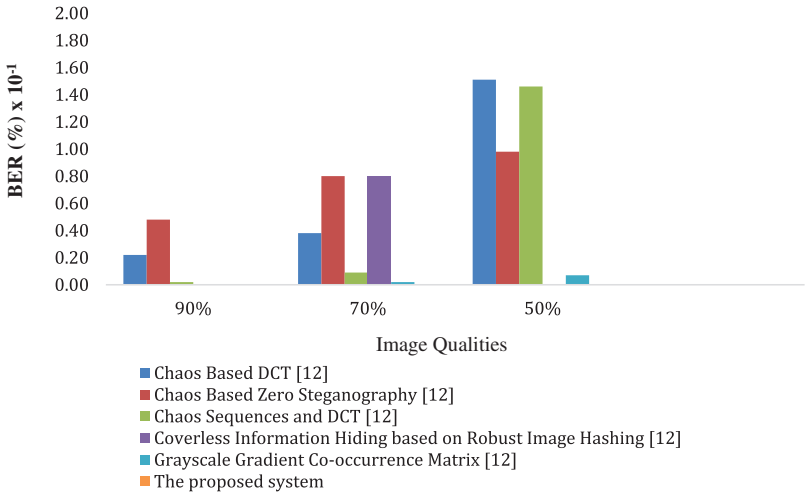

Figure 6: BER values when JPEG compression attack occurred

4.2.4 Other Different Attacks

Yahoo, Facebook, WhatsApp, color space conversion, and image file format are considered dangerous attackers because they can damage some/all of the transmitted data. However, these communication channels change the sent message during data transmission to the receiver such as Facebook [3]. For example, Facebook and WhatsApp compress the sent images. The format of the produced image, MGSI, is PNG. This image was transformed into JPEG, 256 color bitmap, GIF, TIFF, and BMP format. These transformations are considered attackers. The proposed system succeeded in escaping from the trap of these attackers. Accordingly, all values of BER were zeros, this means that there were no errors and the secret message had been retrieved correctly with 100% accuracy.

5 Conclusion

This paper presented a coverless image steganography system that depends on a maze game image generation with high durability, security, and large embedding capacity. The main idea of the system construction was that the secret message bitstreams were used as an input to the system to generate the MGSI. This image was sent to a receiver to extract the secret message.

The system was tested to evaluate the performance using a set of experimental measurements through a comparison with the others. These measurements were embedding capacity and the system attacks. These attacks were image scaling, JPEG compression, color space conversion, file format transformation, image rotation, and communication channel attacks. It was found that the payload capacity was doubled, 1736. The number of required images for embedding different sizes of data was less than half, roughly 5 images are required to hide 8000 bits while one image is required for embedding other sizes. In addition, the system was not affected by all studied types of attacks. It was noticed that the proposed system has the following advantages:

- Dependability: It automatically generates an image effectively hiding the secret message rather than using a cover.
- Optimization: From the experiments, the system achieved the highest embedding capacity and is considered a malicious fugitive from the attackers' trap.
- Game theory: It is considered one of the used techniques for building the maze as a game.

As a future work, more intelligence quotient games can be used to build a coverless steganography system rather than using a cover to embed the secret message. Also, more attacks may be used to check the robustness of the steganography system. New systems can be designed to generate any natural images based on the secret message bits. Most of the efforts in the field of text classification for data hiding have focused on the English language, while research on the Arabic language, which has numerous challenges is scarce [22–24]. This motivates the researchers to focus on hiding a secret message written in the Arabic language. A new coverless steganography system can be designed based on an image gallery to build a stego image [25–28]. Finally, one of the drawbacks of the proposed system is that the attackers can have the ability to read the sent secret message. Just the secret message has been viewed, it may be destroyed or altered before transmitting to the receiver.

Acknowledgement: Taif University Researchers Supporting Project, Taif University, Taif, Saudi Arabia.

Funding Statement: Taif University Researchers Supporting Project Number (TURSP-2020/239), Taif University, Taif, Saudi Arabia.

Author Contributions: Conceptualization, A-H.S.S., M.S.R., G.M.B., A.I.E., M.A.; methodology, A.A.E-H., M.B., M.A.; software, A-H. S.S., G.M.B., A.I.E.; validation, A-H.S.S., H.A., M.B., A.I.E., M.A.; formal analysis, A-H.S.S., M.S.R., M.A.; investigation, A-H.S.S., G.M.B., A.I.E., M.B., M.A.; resources, A.A.E-H., M.B., M.A.; data curation, A-H.S.S, G.M.B., A.I.E., M.A.; writing—original draft preparation, A-H.S.S., M.S.R., G.M.B., A.I.E., M.A.; writing—review and editing, M.B., A-H.S.S.., A.I.E., M.A.; visualization, G.M.B., A.I.E., M.A.; supervision, A.A.E-H., G.M.B.; project administration, M.A.; funding acquisition, M.B. All authors have read and agreed to the published version of the manuscript.

Availability of Data and Materials: None.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] D. Nashat and L. Mamdouh, "An efficient steganographic technique for hiding data," *Journal of the Egyptian Mathematical Society*, vol. 27, no. 1, pp. 1–14, 2019.

[2] L. Meng, X. Jiang, Z. Zhang, Z. Li and T. Sun, "Coverless video steganography based on maximum DC coefficients," *Journal of Latex Class Files*, vol. 14, no. 8, pp. 1–13, 2015.

[3] B. S. Edhah, D. M. Alghazzawi and L. Cheng, "Secret communication on facebook using image steganography: Experimental study," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 14, no. 10, pp. 428–435, 2016.

[4] X. Zhang, F. Peng and M. Long, "Robust coverless image steganography based on DCT and LDA topic classification," *IEEE Transactions on Multimedia*, vol. 20, no. 12, pp. 3223–3238, 2018.

[5] Z. Zhou, Y. Mu and Q. M. J. Wu, "Coverless image steganography using partial-duplicate image retrieval," *Soft Computing*, vol. 23, no. 13, pp. 4927–4938, 2019.

[6] V. Govindasamy, A. Sharma and V. Thanikaiselvan, "Coverless image steganography using haar integer wavelet transform," in *2020 Fourth Int. Conf. on Computing Methodologies and Communication (ICCMC)*, Erode, India, pp. 885–890, 2020.

[7] Y. Tan, J. Qin, X. Xiang, C. Zhang and Z. Wang, "Coverless steganography based on motion analysis of video," *Security and Communication Networks*, vol. 2021, no. 3, pp. 1–16, 2021.

[8] Y. Cao, Z. Zhou, X. Sun and C. Gao, "Coverless information hiding based on the molecular structure images of material," *Computers, Materials & Continua*, vol. 54, no. 2, pp. 197–207, 2018.

[9] S. Zheng, L. Wang, B. Ling and D. Hu, "Coverless information hiding based on robust image hashing," in *Int. Conf. on Intelligent Computing*, Cham, Springer, vol. 1, pp. 536–547, 2017.

[10] Z. Zhou, H. Sun, R. Harit, X. Chen and X. Sun, "Coverless image steganography without embedding," in *Int. Conf. on Cloud Computing and Security*, Cham, Springer, vol. 1, pp. 123–132, 2015.

[11] A. Cressant and S. Granon, "Definition of a new maze paradigm for the study of spatial behavior in rats," *Brain Research Protocols*, vol. 12, no. 2, pp. 116–124, 2003.

[12] J. Wu, Y. Liu, Z. Dai, S. Rahbar and Y. Jia, "A coverless information hiding algorithm based on grayscale gradient co-occurrence matrix," *IETE Technical Review*, vol. 35, no. 4, pp. 23–33, 2018.

[13] S. Li, X. Chen, Z. Wang, Z. Qian and X. Zhang, "Data hiding in iris image for privacy protection," *IETE Technical Review*, vol. 35, no. 1, pp. 34–41, 2018.

[14] Y. Cao, Z. Zhou, Q. M. J. Wu and C. Yuan, "Coverless information hiding based on the generation of anime characters," *EURASIP Journal on Image and Video Processing*, vol. 36, no. 1, pp. 1–15, 2020.

[15] Z. L. Zhou, Y. Cao and X. M. Sun, "Coverless information hiding based on bag-of-words model of image," *Journal of Applied Sciences*, vol. 34, no. 5, pp. 527–536, 2016.

[16] Y. Cao, Z. Zhou, C. Yang and X. Sun, "Dynamic content selection framework applied to coverless information hiding," *Journal of Internet Technology*, vol. 19, no. 4, pp. 1179–1185, 2018.

[17] L. Zou, J. Sun, M. Gao, W. Wan and B. B. Gupta, "A novel coverless information hiding method based on the average pixel value of the sub-images," *Multimedia Tools and Applications*, vol. 78, no. 4, pp. 7965–7980, 2019.

[18] A. H. S. Saad, M. S. Mohamed and E. H. Hafez, "Coverless image steganography based on jigsaw puzzle image generation," *Computers, Materials & Continua*, vol. 67, no. 3, pp. 2077–2091, 2021.

[19] Y. Luo, J. Qin, X. Xiang, Y. Tan, Q. Liu *et al.,* "Coverless real-time image information hiding based on image block matching and dense convolutional network," *Journal of Real-Time Image Processing*, vol. 17, no. 3, pp. 1–11, 2020.

[20] C. Yuan, Z. Xia and X. Sun, "Coverless image steganography based on SIFT and BOF," *Journal of Internet Technology*, vol. 18, no. 2, pp. 435–442, 2017.

[21] C. W. Tang and H. M. Hang, "A feature-based robust digital image watermarking scheme," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 950– 959, 2003.
[22] A. Omar, T. M. Mahmoud, T. A. Hafeez and A. Mahfouz, "Multi-label arabic text classification in online social networks," *Journal of Information Systems*, vol. 100, no. 2021, pp. 101785, 2021.
[23] M. Abouhawwash, K. Deb and A. Alessio, "Exploration of multi-objective optimization with genetic algorithms for PET image reconstruction," *Journal of Nuclear Medicine*, vol. 61, no. 1, pp. 572, 2020.
[24] A. Garg, A. Parashar, D. Barman, S. Jain, D. Singhal *et al.,* "Autism spectrum disorder prediction by an explainable deep learning approach," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 1459–1471, 2022.
[25] W. Sun, X. Chen, X. R. Zhang, G. Z. Dai and P. S. Chang, "A multi-feature learning model with enhanced local attention for vehicle re-identification," *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3549–3560, 2021.
[26] S. Mahajan, A. Raina, M. Abouhawwash, X. Z. Gao and A. K. Pandit, "COVID-19 detection from chest X-ray images using advanced deep learning techniques," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 1541–1556, 2022.
[27] A. Omar, T. M. Mahmoud and T. Abd-El-Hafeez, "Building online social network dataset for arabic text classification," in *Int. Conf. on Advanced Machine Learning Technologies and Applications (AMLTA2018)*, Cairo, Egypt, Springer, vol. 723, pp. 486–495, 2018.
[28] A. Omar, T. M. Mahmoud and T. Abd-El-Hafeez, "Comparative performance of machine learning and deep learning algorithms for arabic hate speech detection in OSNs," in *Int. Conf. of Artificial Intelligence and Computer Vision (AICV2020)*, Cham, Cairo, Egypt, Springer, vol. 1153, pp. 247–257, 2018.