Computers, Materials & Continua
DOI:10.32604/cmc.2021.015329
Article

Tech Science Press

Coverless Image Steganography Based on Jigsaw Puzzle Image Generation

Al Hussien Seddik Saad[1,*], M. S. Mohamed[2,3] and E. H. Hafez[4]

[1]Department of Computer Science, Faculty of Science, Minia University, Minya, 61519, Egypt
[2]Department of Mathematics, College of Science, Taif University, Taif, 21944, Saudi Arabia
[3]Department of Mathematics, Faculty of Science, Al-Azher University, Nasr City, 11884, Egypt
[4]Department of Mathematics, Faculty of Science, Helwan University, Cairo, Egypt
[*]Corresponding Author: Al Hussien Seddik Saad. Email: al.hussien_seddik@mu.edu.eg
Received: 15 November 2020; Accepted: 10 December 2020

Abstract: Current image steganography methods are working by assigning an image as a cover file then embed the payload within it by modifying its pixels, creating the stego image. However, the left traces that are caused by these modifications will make steganalysis algorithms easily detect the hidden payload. A coverless data hiding concept is proposed to solve this issue. Coverless does not mean that cover is not required, or the payload can be transmitted without a cover. Instead, the payload is embedded by cover generation or a secret message mapping between the cover file and the payload. In this paper, a new coverless image steganography method has been proposed based on the jigsaw puzzle image generation driven by a secret message. Firstly, the image is divided into equal rows then further divided into equal columns, creating blocks (i.e., sub-images). Then, according to secret message bits and a proposed mapping function, each block will have tabs/blanks to get the shape of a puzzle piece creating a fully shaped jigsaw puzzle stego-image. After that, the generated jigsaw puzzle image is sent to the receiver. Experimental results and analysis show a good performance in the hiding capacity, security, and robustness compared with existing coverless image steganography methods.

Keywords: Coverless information hiding; Jigsaw puzzle image; image steganography; data hiding

1 Introduction

Transmission of sensitive and classified information over the internet has become one of today's main challenges. This sensitive data can be easily disclosed or exposed through unintentional, intentional, or illegal actions [1]. Growth in cyberattacks and cybercrimes [2] has led to significant attention in securing the transmission over public channels as the internet [3]. The security of transmission can be ensured by various methodologies of data hiding, such as steganography.

Steganography, an ancient practice [4], is one of the leading techniques for personal or sensitive data protection by achieving the principle of concealment [5]. It means hiding sensitive information in other innocent-looking media formats [3] such as video, audio, image, and DNA [6], creating the stego-file [7]. So that eavesdroppers can not notice that the communication is taking place [8]; consequently, the existence of the communication can not be diagnosed [9]. So, the goal is to provide secure and concealed data transmission [10].

Among all carriers, image files are the most popular covers for data hiding, as it is the most commonly used media file on the internet [11]. Traditional image steganography methods work by modifying image pixels' values to embed the payload. They are categorized into frequency domain and spatial domain steganography methods [12,13].

Firstly, in spatial domain methods, the payload is directly embedded into pixels of the cover image by modifying their values [3]. Typical spatial domain methods include "least significant bit" (LSB) and "pixel value difference" (PVD) methods [14]. While in frequency domain hiding methods, the secret data is embedded in image data after being transformed using a transformation function. These methods, such as data hiding in DCT, DWT, and DFT, which are "discrete cosine transform," "discrete wavelet transform," and "discrete Fourier transform" domains respectively [12,14–16].

The following key objectives are used to evaluate the image steganography method. The first objective is the embedding capacity measured in "bits per pixel" (bpp). It is defined as the maximum payload amount that can be concealed within a cover image and extracted correctly [17]. The second one is invisibility, which measures the visual similarity between stego and cover image. Invisibility is measured in the "Peak Signal-to-Noise Ratio" PSNR. The third objective is security, which can be defined as the resistance to unauthorized access attacks. Finally, robustness, a robust method, protects the payload from modification and destruction [18]. Therefore, the ideal image steganography method must fulfill all of these key objectives simultaneously [7,13].

As the embedding process is done by modifying the cover image, it is certain to leave some modification traces. Hence, we face such a dilemma: it can be detected by various existing steganalysis tools based on the modification traces [14]. So, the concept of hiding data coverlessly is proposed to avoid these left traces [14].

Coverless steganography or data hiding came into being in 2015. It does not mean that the carrier is not needed [19], or the secret information can be transferred without a carrier. Instead, the payload is embedded by carriers generation or by secret message mapping [3].

Coverless data hiding is classified into text steganography and image steganography according to carrier type. Coverless text steganography embeds the payload by creating a relation between texts and words; then, this information is hidden according to the mapping rfules and tags [3]. Whereas coverless image steganography can be categorized into two types. Image generation is driven by payload, such as data hiding by texture images generation. The other is to establish a mapping rule between the cover image and the payload to represent the secret message [15]. Therefore, coverless steganography methods have higher security than traditional steganography methods [14].

The main contribution of this paper is to propose a novel coverless image steganography method that is based on jigsaw puzzle image generation driven by secret message. It does not modify cover image pixels' values as LSB and PVD methods; instead, it generates a jigsaw puzzle image based on the secret message bits.

The remainder of the paper is arranged as follows: Section 2 presents some related works and studies on coverless image steganography. Section 3 introduces the proposed coverless Jigsaw puzzle image steganography in detail. Section 4 displays the results and analysis of the proposed method. Finally, Section 5 summarizes the paper.

2 Related Studies

In this section, some studies and related work on coverless image steganography is presented in detail.

Zhou et al. [16] proposed a method that works as follows: A set of images have been downloaded from the internet to create an image database. Then by using a robust algorithm, a hash sequence for each image will be generated. After that, according to generated image hashes that have to be shared between the sender and receiver, images are indexed to build an inverted index structure. Afterward, the secret message will be transmitted as follows: Firstly, the sender converts it into a bitstream. Secondly, segments it into equal-length segments. Thirdly, search for images that have hash sequences equal to the secret message segments. Finally, transmit these images (i.e., stego-images) to the receiver.

Zheng et al. [15] proposed a robust hash algorithm based on "Scale Invariant Feature Transform" (SIFT) that creates a binary hash sequence of 18-bits for each image to be used in their proposed coverless method. The proposed method starts with constructing an image database (local database) that contains each image and its corresponding 18-bits hash value. However, the authors found a problem that they need 218 images in the database with different hashes to have the ability to represent the whole 18-bits combinations, which is not practical. So, they represented each image with a different number of 18-bits hashes to minimize the required size of the database. Finally, the secret message is segmented into 18-bits segments, which equal to image hash value size, then stego-image of hash value equals to secret message segment is selected as a carrier to be sent to the recipient.

Zhang et al. [3] proposed a coverless algorithm based on "Latent Dirichlet Allocation" LDA topic classification and "discrete cosine transform" (DCT). Firstly, for database image classification, the "LDA model" has been used. Secondly, similar topic images are selected then DCT is performed on each 8×8 block in these images. Thirdly, based on the relation between block coefficients, a feature sequence is generated. After that, an inverted index has been created that consists of the following; image path, location coordinates, dc, and feature sequence. Finally, the secret message is converted into a bitstream and segmented into equal-length segments. Then the image that has a feature sequence equals to secret message segment will be selected as a stego image then all stego-images are sent to the recipient.

Cao et al. [19] proposed a coverless steganography method based on "molecular structure images of material." The method works in this way; first, the payload is represented by a stream of bits. Then, divided into equal-length segments. After that, the molecular image is divided into sub-images (i.e., $x \times y$ blocks). Finally, the average pixel values of these blocks represent secret message segments by a mapping function between secret message segments and pixel value intervals.

Duan et al. [12] proposed a method based on a generative model that works as follows; firstly, the sender has to feed the secret image as an input to the database that, in turn, creates another independent image that equals the secret image in its histogram distribution. Then, the histogram equally created image is sent to the receiver that has to feed it again as an input into the database

to create another image that is the same as the secret image visually. Finally, the authors said that both sides of communication have to share the same parameters and database.

Zhou et al. [9] proposed a coverless method based on partial-duplicate image retrieval, which works as follows; firstly, a vast image database has to be constructed by downloading images from the internet. Secondly, each image will be segmented into a set of non-overlapping blocks. Then, by using a hashing algorithm, each block of each image will have a label that will be used as the location. This location information marks which block of the image is used for secret message hiding, and it has to be shared at both communication sides. Afterward, the feature will be extracted from each block, and the inverted index structure is built. Finally, the hiding process works as follows. The payload will be hidden by dividing it into equal blocks. Then, for each block, by using the inverted index, a partial-duplicate image, which consists of similar blocks with the secret image, is retrieved. In the end, a lot of partial-duplicate images that can be considered as stego-images are gained. Those images are then sent to the recipient.

Wu et al. [20] proposed a method based on the "Grayscale Gradient Co-occurrence Matrix." This method works as follows; firstly, the payload is represented in a binary stream. Then, this binary stream will be divided into equal size segments, each of which is of 8-bits length. Secondly, using a turbo encoder, each 8-bit segment is coded to increase the segment's length to become 16 bit because of the data rate. Thirdly, according to the mapping function. Search for an image corresponds to the 16-bit length segment. Until all of the payload segments are represented. Fourthly, search for the image that corresponds to secret message length and put it after secret message images. Finally, send all images to the receiver side.

Zou et al. [14] proposed a coverless method based on the Chinese sentences that include subject, predicate, object, preposition. First of all, the payload is divided according to the sentence structure of the Chinese language. These segments are then marked as $\{I_1, I_2, \ldots, I_n\}$; n refers to segments number of the sentence in the Chinese language. Then, from the dictionary, each segment's position could be obtained, which is marked as $\{P_1, P_2, \ldots, P_n\}$. Afterward, According to the payload segment's position, label information of the hash sequence in each part of the hash array of images can be obtained, marked as $\{L_1, L_2, \ldots, L_n\}$. Finally, according to the label information, corresponding images can be indexed. Then, randomly selected stego images are transmitted to the recipient.

Finally, after presenting current coverless image steganography methods and their characteristics, it has been discovered that current methods are facing the following problems:

(1) Almost all of them are sensitive to image processing attacks as they extract features in the spatial domain [3].
(2) Current methods suffer from insufficient robustness and security [3].
(3) Moreover, they have low embedding capacity [19].
(4) Several images are required to represent the secret information [3,19].
(5) Large image database is required [15,16].
(6) Similar to image retrieval, they search in the database for the images representing secret message bits.
(7) Cover images are randomly selected, which results in a significant difference in the contents between these images. It may arouse the attacker's suspicion and greatly reduce coverless image steganography security [3].

(8) Current methods generate a hash sequence for each image/image block in the database at both sender and receiver, and this is not a practical solution as it costs time and resources [9,14–16].

3 Proposed Method

Assume that A and B are users that are communicating with each other using any application. The problem here is that unauthorized users can reach this personal information. So, secret information can be tampered with, lost, or manipulated during the transmission. Thus, it is clear that a secure and safe information transmission method is needed. To solve this issue, a novel coverless image steganography method based on the jigsaw puzzle image generation has been proposed.

The proposed method not only secured the information during transmission but also solved the problems stated in the previous section that are facing current coverless methods.

It contains two main sides: embedding, where the puzzle image will be created based on the payload, and extraction, where the secret message will be extracted. The proposed method will be described in detail in the next sections.

3.1 Information Hiding

As stated before, the proposed method is based totally on jigsaw puzzle image generation and secret message mapping. So the next subsection will briefly describe the jigsaw puzzle image and puzzle piece structure.

3.1.1 Jigsaw Puzzle

Learnersdictionary.com [21] defined a jigsaw puzzle as "a puzzle made of many small pieces that are cut into various shapes and can be fit together to form an image." See Fig. 1a. The shape of most jigsaw puzzle pieces is either rectangular or square, all of a similar shape, with tabs and blanks, see Fig. 1b. These tabs and blanks are arranged randomly on each piece. Usually, Each piece has four sides, so the total number of tabs and blanks is up to four, as shown in Fig. 1b [22].

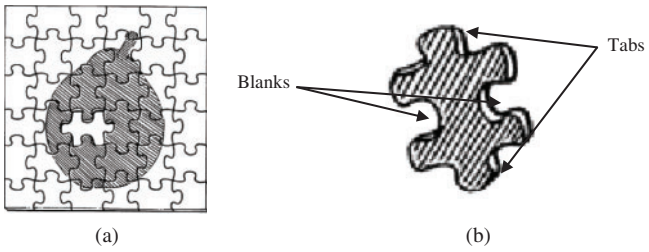

Figure 1: (a) Jigsaw puzzle image [21]. (b) Jigsaw puzzle piece structure (4-sides) [21]

3.1.2 Information Hiding Process

As shown in Fig. 2, the proposed coverless information hiding method works as follows: The selected image and the secret message (payload) are fed as inputs to the system (check Algorithm 1). Then the payload is converted into a stream of bits (i.e., 0's and 1's). On the other hand, the selected image will be divided into rows of similar heights by adding horizontal lines, see "Divided into Rows" step. Then the resulted image will be again divided into columns of similar width by

adding vertical lines, and the resulted image will be segmented into equal size blocks (i.e., sub-images) as shown in the "Divided into Blocks" step. These blocks are the puzzle pieces without tabs and blanks introduced above (i.e., without representing secret bits). Finally, the mapping function, which works as follows: first, scans the generated image blocks row by row then column by column to visit each block from all sides. Then, it takes the secret bit to be represented, which will be either 1 or 0. If the secret bit to be represented is 1, then a tab will be added to the current block side (i.e., top, bottom, left, or right) depending on the block location and the scanning algorithm. Otherwise, if the secret bit is 0, a blank will be added to the current block side. This process will be repeated until tabs and blanks represent the whole secret bits; see the "Puzzle Generation" step. In the end, the lines between the blocks and tabs/blanks will be removed, generating the shape of a jigsaw puzzle image (i.e., stego image), see "Removing Lines" step, which will be finally sent to the receiver.

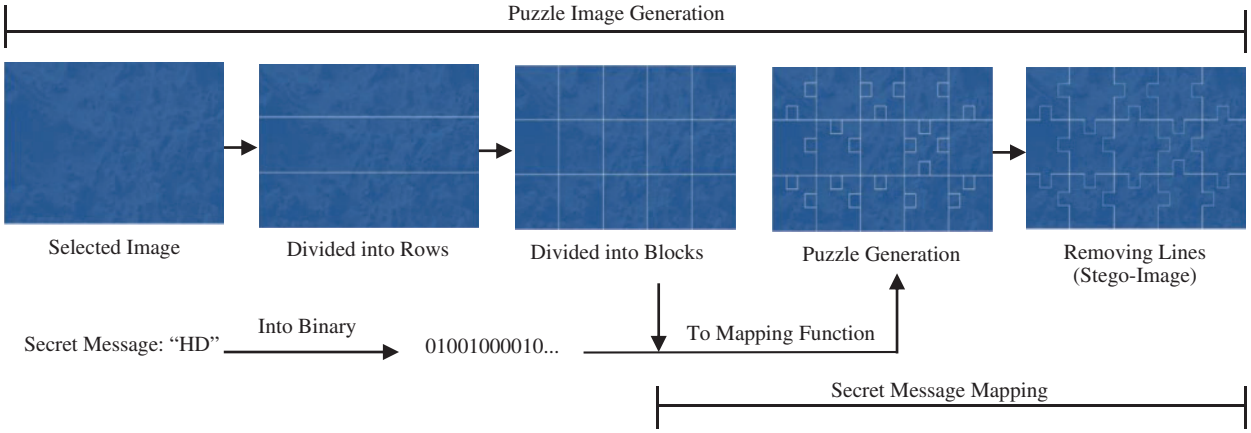

Figure 2: Information hiding framework

As an example, see Fig. 3. If the secret message bits '010010000100' and the selected image are fed as inputs to the hiding algorithm, then the jigsaw puzzle image will be generated based on the secret message bits, as shown in the figure.

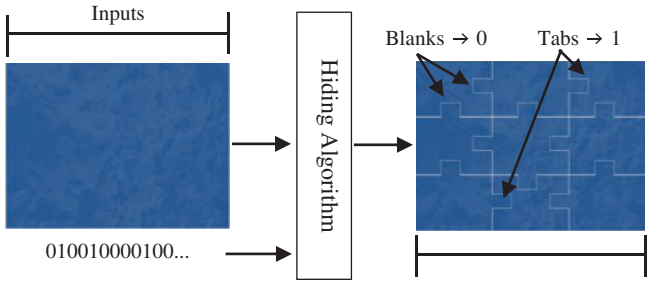

Figure 3: Information hiding example

3.1.3 Information Hiding Algorithm

Algorithm 1: Hiding algorithm

Input: Selected Image (SI), Secret Message (SM)
Output: Generated Jigsaw Puzzle Image (GJPI)

1) Divide Secret Message 'SM' into characters; $SM = \{sm_1, sm_2, sm_3, \ldots, sm_n\}$.
2) Convert characters of 'SM' into Secret Message Bits; $SMB = \{smb_1, smb_2, smb_3, \ldots, smb_n\}$.
3) Divide Selected Image 'SI' into equal height rows by 2-pixels width horizontal lines 'SI_R'
4) Divide 'SI_R' into equal columns by 2-pixels vertical lines creating Blocks; $IB = \{ib_1, ib_2, \ldots, ib_n\}$.
5) If (length(SMB) <= length(IB)
6) For $i = 1$ to length(SMB)
7) If ($smb_i == 1$)
8) Add 'Tab' to the current Image Block side ib_i.
9) Else if ($smb_i == 0$)
10) Add 'Blank' to the current Image Block side ib_i.
11) End if
12) End For
13) Else
14) Print("Secret Message is larger than image capacity")
15) End if
16) Return GJPI.

3.2 Information Extraction

3.2.1 Information Extraction Process

The extraction process, which is simpler than the hiding process, works as follows: The receiver inputs the generated jigsaw puzzle image into the extraction system (check Algorithm 2). First, the system scans the puzzle image row by row to visit each block's left and right sides. Then, for each block, each tab will represent a secret bit 1, and each blank will represent a secret bit 0. Secondly, the same process will be done column by column to visit each block's top and bottom sides. Afterward, all of these secret bits are collected and concatenated to form the secret bitstream. Finally, the obtained bitstream is segmented into chunks of 8-bits, converted into characters, joined together, forming the secret message.

3.2.2 Information Extraction Algorithm

Algorithm 2: Extraction algorithm

Input: Generated Jigsaw Puzzle Image (GJPI)
Output: Secret Message (SM)

1) //Initializing SecretMessage Variable
2) SM=" "
3) //Convert Generated Jigsaw Puzzle Image Into Binary Image

(Continued)

```
4) Gray_GJPI = Convert GJPI Into Grayscale
5) BW_GJPI = Convert Gray_GJPI into Binary
6) //Puzzle Lines Duplication, To be easily detected.
7) If (BW_GJPI Lines Need Duplication)
8)    Duplicate Lines
9) End if
10) //Detecting Puzzle Rows and Columns Pixels
11) Scan image rows of BW_GJPI, R = {r1, r2, r3, ..., rn}
12) Scan image columns of BW_GJPI, C = {c1, c2, c3, ..., cn}
13) //Scan the image row by row to get blocks' tabs and blanks of left and right sides
14) For i = 1:Length(R)
15)         For j = 1:Length(C)
16)            If (Block(ri, cj) has a Tab)
17)               SM = SM + '1'
18)            Else if (Block(ri, cj) has a Blank)
19)               SM = SM + '0'
20)            End if
21)         End for
22) End for
23) //Scan image column by column to get blocks' tabs and blanks of top and bottom sides
24) For i = 1: Length(C)
25)         For j = 1:Length(R)
26)            If (Block(cj, ri) has a Tab)
27)               SM = SM + '1'
28)            Else if (Block(cj, ri) has a Blank)
29)               SM = SM + '0'
30)            End if
31)         End for
32) End for
33) Return 'SM'
```

4 Evaluation and Comparisons

Solving the insufficiency that has been found in current methods was the main focus of this work. The proposed method was experimentally evaluated to assess its effectiveness and verify and evaluate its efficiency compared with current coverless image steganography methods. All the experiments have been done using MATLAB, images of size 512×512 pixels (dark images are recommended), and puzzle pieces of 50×50 pixels. Almost all previous coverless methods either use 512×512 images or larger [3,9,15], or did not depend on image size at all.

4.1 Capacity

As shown in Tabs. 1 and 2, the proposed method's hiding capacity is the highest among current coverless methods, 760 bits/cover. The cover size is 512×512 pixels, and the puzzle piece size is 25×25 pixels, the smallest puzzle piece size that can be used and detected successfully.

Larger cover images can be used to gain more capacity according to the actual needs. The following equations are used to get the full capacity (bits/cover) of the method:

$$Number\ OF\ Puzzle\ Rows = \frac{Number\ of\ Image\ Rows}{Size\ of\ Puzzle\ Piece\ Rows} \tag{1}$$

$$Number\ OF\ Puzzle\ Columns = \frac{Number\ of\ Image\ Columns}{Size\ of\ Puzzle\ Piece\ Columns} \tag{2}$$

$$Puzzle\ Rows\ Capacity = No.\ of\ Puzzle\ Rows * (No.\ of\ Puzzle\ Columns - 1) \tag{3}$$

$$Puzzle\ Columns\ Capacity = (No.\ of\ Puzzle\ Rows - 1) * No.\ of\ Puzzle\ Columns \tag{4}$$

$$Full\ Image\ Capacity = Puzzle\ Rows\ Capacity + Puzzle\ Columns\ Capacity \tag{5}$$

As an example, if the cover image size is 512×512 pixels, and the puzzle piece size is 25×25 pixels. Then, the hiding capacity for this cover image can be calculated as follows:

$$No.\ of\ Puzzle\ Rows = \frac{512}{25} \approx 20\ Puzzle\ Rows$$

$$No.\ of\ Puzzle\ Cols = \frac{512}{25} \approx 20\ Puzzle\ Cols$$

$$Puzzle\ Rows\ Capacity = 20 \times 19 = 380\ Left/Right\ Sides$$

$$Puzzle\ Columns\ Capacity = 19 \times 20 = 380\ Top/Bottom\ Sides$$

$$Full\ Image\ Capacity = 380 + 380 = 760\ Sides\ (i.e.,\ bits/cover)$$

As shown in Tab. 1, the proposed method achieved the highest hiding capacity among almost all previously proposed coverless image steganography methods. Which is 760 bits/cover; more capacity can be obtained by using larger covers, which means that the proposed method enhanced embedding capacity insufficiency of coverless data hiding technique.

Table 1: Proposed method embedding capacity

Method	Capacity (bits/carrier)
Zhou et al. [16]	8
Yuan et al. [23]	8
Cao et al. [24]	14
Zhang et al. [3]	1∼15
Zhou et al. [25]	16
Zheng et al. [15]	18
Cao et al. [19]	36
Cao et al. [26]	68
Zou et al. [14]	80
Zhou et al. [9]	384
Proposed method	760

Tab. 2 compares the number of required images to represent the same secret message using different coverless image steganography methods. As shown in the table, the method that required

the lowest number of images among all methods is the proposed one, which means a smaller number of covers needed.

Table 2: Number of images needed when the same data is hidden [15]

Method	Secret message length			
	1 byte	10 bytes	100 bytes	1 kilobyte (kB)
Zhou et al. [16]	1	10	100	1024
Yuan et al. [23]	1	10	100	1024
Zhang et al. [3]	2∼9	7∼81	55∼801	548∼8,193
Zheng et al. [15]	2	6	46	457
Proposed method	1	1	1.05	10.7

4.2 Robustness

Robustness is an important performance index in a coverless image steganography algorithm as it measures the ability of the method to resist different attacks, and it determines whether the payload can be correctly extracted or not. Algorithm failure is caused by various attacks such as noise attacks, scaling attacks, JPEG compression, etc. In this section, the robustness of the proposed coverless method will be tested, evaluated, and verified through experiments and comparisons [20]. First, bit error rate (BER) must be defined, which is a robustness measure that can be calculated as follows [20,27]:

$$BER = \frac{e}{n}, e = \sum_{i=1}^{n} p_i \oplus q_i \tag{6}$$

where e is the number of errors found, n represents the total number of bits, p is the original bitstream, and q is the extracted bitstream after the attack. If BER is zero, this means no errors were found (i.e., the secret bitstream has been extracted successfully), and the proposed method is 100% robust to this attack under current circumstances. Contrarily, if BER > 0, the secret bitstream is not extracted successfully (i.e., bits have been modified), and this means that the method is not 100% robust to this attack.

4.2.1 Scaling Attack

Scaling is one of the brutal attacks, as scaling an image can destroy part or whole of the represented message (i.e., hidden message).

Tab. 3 shows that, at a scaling ratio of 0.3, the proposed method failed because after scaling to 0.3, the puzzle piece size became 15 × 15 pixels, and all the tabs and blanks have been intersected, and the algorithm failed to detect them. While at the scale of 0.5, the total number of errors found is 2 bits only, and the BER was 0.011. Finally, starting from scale 1.5 to 10 (and higher), there were no errors found, and the BER values were 0, which means the message was fully detected successfully with 100 % accuracy, and the method is 100% robust in these ranges.

4.2.2 JPEG Compression Attack

JPEG is the most popular, lossy compression standard for images that allows the images to lose data; it is applied before and during digital image transmission [20]. Stego-image loses

secret message, some/all of it, through transmission if it has been compressed. BER has been calculated for the proposed method after the JPEG attack. The quality factor range of the JPEG compression is from 1 to 100; the lowest compression ratio is 100, while the highest compression ratio is 1.

Table 3: Comparison of BER after scaling attack

Ratio of scaling	RCIS [3]	Wu et al. [20]	Proposed method
0.3	0.146	0.015	Failed
0.5	0.057	0.009	0.011
0.75	0.039	0.002	0
1.5	0.016	0.025	0
2.0	–	–	0
3.0	–	–	0
4.0	–	–	0
5.0	–	–	0
6.0	–	–	0
7.0	–	–	0
8.0	–	–	0
9.0	–	–	0
10.0	–	–	0

Tab. 4 compares previously proposed methods; CBD, CBZS, CSD, CIHRIH, Wu et al. [20] methods, and the proposed method. The proposed method results were 0 BER for all image qualities from (90 to 20), which means the proposed algorithm is 100% robust to JPEG attacks at these values. The secret message is fully detected successfully. Except for the lowest image quality, 10, the number of errors found was 3 bits. As an important note, the original image file (.PNG) size before compression was 242 KB. The compressed file sizes were "87, 63, 50, 42, 37, 31, 26, 20 and 10 KB" corresponding to image qualities "90, 80, 70, 60, 50, 40, 30, 20 and 10" respectively.

Table 4: Comparison of BER after JPEG compression attack

Quality	CBD [20]	CBZS [20]	CSD [20]	CIHRIH [20]	Wu et al. [20]	Proposed method
90	0.022	0.048	0.002	0	0	0
80	–	–	–	–	–	0
70	0.038	0.080	0.009	0.08	0.002	0
60	–	–	–	–	–	0
50	0.151	0.146	0.146	–	0.007	0
40	–	–	–	–	–	0
30	–	–	–	–	–	0
20	–	–	–	–	–	0
10	–	–	–	–	–	0.0167

4.2.3 Noise Attack

"Salt and Pepper" noise occurs due to failures of software or breakdown of hardware, etc. It corrupts image quality by substituting the pixels randomly to its highest value, which is 255 (i.e., white pixel), or to its lowest value, which is 0 (i.e., black pixel) [27]. Naked eyes easily detect these black/white dots. The proposed method's robustness will be evaluated by applying "salt and pepper" noise at different densities, as shown in Tab. 5 [27].

Table 5: Comparison of BER after noise attack

Noise density	CIHWE [16]	CIHRIH [15]	Wu et al. [20]	Proposed method
0.01	0.02	0.01	0	0
0.02	0.06	0.04	0	0
0.03	0.11	0.05	0	0
0.04	0.16	0.09	0.0005	0
0.05	–	–	–	0
0.06	–	–	–	0
0.07	–	–	–	0
0.08	–	–	–	0
0.09	–	–	–	0
0.1	–	–	–	0.0278

Tab. 5 compares the BER values among CIHWE [16], CIHRIH [15], Wu et al. [20] methods, and the proposed method after being attacked by salt and pepper noise. The results showed that the BER of the proposed method is zero at almost all noise densities except the density of 0.1, which means the proposed method is 100% robust to "salt and pepper" noise attacks at these densities.

4.2.4 Other Attacks

This section includes other attacks including RGB to greyscale conversion, RGB to binary image conversion, image file format conversion, Facebook attack: Facebook automatically alters and compresses uploaded images, so, If the image is a stego-image, Facebook destroys it [28], WhatsApp attack: WhatsApp also compresses any image or media file automatically before transmitting it to the receiver which in turn degrades the quality of original file [29], median filter attack: which is a filter that is used for noise removal from a digital image [30]. Tab. 6 presents the BER results of applying these attacks.

As presented in Tab. 6, the proposed method succeeded in resisting almost all of the above attacks as the BER values were 0 except the median filter's final one. The puzzle pieces have been mixed with image pixels due to filtering the image by a 6×6 median filter.

4.3 Security

4.3.1 Resistance to Attackers

As stated before, if the communication is monitored, then the security is compromised. If the attacker needs to get the secret message, he has to discover that the communication is taking place then read the message [20]. There is no hidden message in image pixels; it is only a jigsaw puzzle image that is not suspicious at all. Thus, the proposed method affords a high-security level.

Table 6: BER of the proposed method after different attacks

Other attack		Proposed method
Color space conversion (RGB)	Greyscale	0
	Binary	0
File format conversion (PNG)	JPG	0
	256 color bitmap (8-bits)	0
	GIF (8-bits)	0
	TIF (32-bits)	0
	BMP (24-bits)	0
Facebook attack	Send → receive → sendback	0
WhatsApp attack	Send → receive → sendback	0
Median filter	2×2	0
	3×3	0
	4×4	0
	5×5	0
	6×6	Failed

4.3.2 Resistance to Steganalysis Attack

An ideal information hiding method should resist various tools of steganalysis. Almost all current methods can be detected by steganalysis tools, which use the resulted pixel value changes from the secret message hiding process [16]. However, these tools cannot detect the proposed coverless method. Instead of modifying the selected image pixels' values for embedding the payload, it directly generates tabs and blanks, creating a traditional jigsaw puzzle image representing the secret message bits. Notably, jigsaw puzzle images are not suspicious at all as they are a commonly known game.

5 Conclusion

In this paper, a highly robust, highly secured, and high embedding capacity coverless image steganography method based on jigsaw puzzle image generation was proposed. The method works in this way; the payload is transformed into a bitstream. On the other hand, the cover image will be divided into equal rows then into equal columns creating similar image blocks; these blocks are the puzzle pieces without tabs and blanks. Then, each block will have tabs and blanks at each side, representing the secret message bits according to the mapping function. Where secret message bit 1 will be represented by a tab and secret message bit 0 will be represented by blank at the current block side. This process will be repeated until the whole message bits are represented, generating a full jigsaw puzzle image. Finally, the generated puzzle image will be transferred to the receiver. The experimental results and analysis in Section 4 showed that the proposed method has a very high embedding capacity, as shown in "Section 4.1," Tabs. 1 and 2, and a larger image can be used according to actual needs to achieve higher hiding capacity for the proposed method. Also, the proposed method has very high robustness to common image attacks, as shown in "Section 4.2," Tabs. 3–6. Moreover, the security objective has been achieved, as discussed in Section 4.3. Finally, the proposed method successfully solved almost all of the previously stated problems facing current coverless methods; hiding capacity and robustness have been enhanced. No database is required, and only one image can represent the secret message; as cover image capacity is 760 bits/cover, so one transmission to the receiver can represent the whole

secret message, which in turn not arousing suspicion. Also, no other algorithms are required, such as hash function and searching algorithms. So, the proposed method is not sensitive to image processing operations. So, the main goal of this paper has been achieved.

Acknowledgement: The authors are thankful of the Taif University. Taif University researchers supporting Project No. (TURSP-2020/160), Taif University, Taif, Saudi Arabia.

Funding Statement: This paper was funded by "Taif University Researchers Supporting Project No. (TURSP-2020/160), Taif University, Taif, Saudi Arabia."

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

References

[1] H. H. Liu and C. M. Lee, "High-capacity reversible image steganography based on pixel value ordering," *EURASIP Journal on Image and Video Processing*, vol. 2019, no. 1, pp. 1–15, 2019.

[2] D. Stănescu, M. Stratulat, R. Negrea and I. Ghergulescu, "Cover processing-based steganographic model with improved security," *Acta Polytechnica Hungarica*, vol. 16, no. 1, pp. 227–246, 2019.

[3] 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.

[4] X. Huan, H. Zhou and J. Zhong, "LSB based image steganography by using the fast marching method," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 3, pp. 1–5, 2019.

[5] 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. sup1, pp. 34–41, 2018.

[6] P. Malathi, M. Manoaj, R. Manoj, V. Raghavan and R. E. Vinodhini, "Highly improved DNA based steganography," *Procedia Computer Science*, vol. 115, pp. 651–659, 2017.

[7] H. Lee, "Data hiding in spatial color images on smartphones by adaptive R-G-B LSB replacement," *IEICE Transactions on Information and Systems*, vol. 101, no. 8, pp. 2163–2167, 2018.

[8] E. A. Abbood, R. M. Neamah and S. Abdulkadhm, "Text in image hiding using developed LSB and random method," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 4, pp. 2091–2097, 2018.

[9] 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.

[10] I. J. Kadhim, P. Premaratne, P. J. Vial and B. Halloran, "Comprehensive survey of image steganography: Techniques, evaluations and tirends in future research," *Neurocomputing*, vol. 335, pp. 299–326, 2019.

[11] Z. Wang and X. Zhang, "Secure cover selection for steganography," *IEEE Access*, vol. 7, pp. 57857–57867, 2019.

[12] X. Duan, H. Song, C. Qin and M. K. Khan, "Coverless steganography for digital images based on a generative model," *Computers, Materials & Continua*, vol. 55, no. 3, pp. 483–493, 2018.

[13] M. Hussain, A. W. A. Wahab, Y. I. Bin Idris, A. T. S. Ho and K. H. Jung, "Image steganography in spatial domain: A survey," *Signal Processing and Image Communication*, vol. 65, pp. 46–66, 2018.

[14] 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, pp. 7965–7980, 2019.

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

[16] Z. Zhou, H. Sun, R. Harit, X. Chen and X. Sun, "Coverless image steganography without embedding," in *Int. Conf. on Computational Science*, vol. 1, pp. 123–132, 2015.
[17] S. Mukherjee, S. Roy and G. Sanyal, "Image steganography using mid position value technique," *Procedia Computer Science*, vol. 132, pp. 461–468, 2018.
[18] M. Taleby Ahvanooey, Q. Li, J. Hou, H. Dana Mazraeh and J. Zhang, "AITSteg: An innovative text steganography technique for hidden transmission of text message via social media," *IEEE Access*, vol. 6, pp. 65981–65995, 2018.
[19] 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.
[20] J. Wu, Y. Liu, Z. Dai, Z. Kang, S. Rahbar *et al.,* "A coverless information hiding algorithm based on grayscale gradient co-occurrence matrix," *IETE Technical Review*, vol. 4602, pp. 22–33, 2019.
[21] "Jigsaw Puzzle Definition," [Online]. Available: http://www.learnersdictionary.com/definition/jigsawpuzzle.
[22] "Jigsaw Puzzle," [Online]. Available: https://en.wikipedia.org/wiki/Jigsaw_puzzle.
[23] C. Yuan, Z. Xia and X. M. Sun, "Coverless image steganography based on SIFT and BOF," *Journal of Internet Technolgy*, vol. 18, pp. 435–442, 2017.
[24] Y. Cao, Z. Zhou, Q. M. J. Wu, C. Yuan and X. Sun, "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.
[25] 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, pp. 527–536, 2016.
[26] Y. Cao, Z. Zhou, C. N. 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.
[27] A. K. Sahu and G. Swain, "A novel n-rightmost bit replacement image steganography technique," *3D Research*, vol. 10, no. 1, pp. 1–18, 2019.
[28] J. Hiney, T. Dakve, K. Szczypiorski and K. Gaj, "Using facebook for image steganography," *Journal of Computer Science and Information Security*, vol. 14, pp. 428–445, 2015.
[29] "WhatsApp," [Online]. Available: https://www.quora.com/How-can-I-send-original-high-quality-photos-on-WhatsApp-and-avoid-sending-automatically-compressed-photos.
[30] "Median Filter," [Online]. Available: https://en.wikipedia.org/wiki/Median_filter.