IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 16, 2020, accepted January 5, 2021, date of publication January 11, 2021, date of current version January 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3050737

Coverless Image Steganography Based on Optical Mark Recognition and Machine Learning

AL HUSSIEN S. SAAD[1], M. S. MOHAMED[2,3], AND ESLAM H. HAFEZ[4]

[1]Computer Science Department, 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-Azhar University, Cairo 11884, Egypt
[4]Mathematics Department, Faculty of Science, Helwan University, Helwan 11795, Egypt

Corresponding author: Al Hussien S. Saad (al.hussien_seddik@mu.edu.eg)

This work was supported by the Taif University Researchers Supporting Project, Taif University, Taif, Saudi Arabia under Grant TURSP-2020/160.

ABSTRACT The most significant factor to consider during private information transmission through the internet (i.e., insecure channel) is security. So, to keep this data from unauthorized access during transmission, steganography is used. Steganography is the scheme of securing sensitive information by concealing it within carriers such as digital images, videos, audio, text, etc. Current image steganography methods work as follows; it assigns cover image then embeds the secret message within it by pixels' modifications, creating the resultant stego-image. These modifications allow steganalysis algorithms to detect the embedded secret message. So, a coverless data hiding concept is proposed to solve this problem. Coverless does not mean that the secret message will be transmitted without using a cover file, or the cover file can be discarded. Instead, the secret message will be embedded by generating a cover file or a secret message mapping. In this paper, a novel, highly robust coverless image steganography method based on optical mark recognition (OMR) and rule-based machine learning (RBML) is proposed.

INDEX TERMS Coverless information hiding, optical mark recognition (OMR), rule-based machine learning (RBML), image steganography.

I. INTRODUCTION

Communicating and storing sensitive and confidential information has become part of day-to-day life. The digitalization of information and innovations in internet technologies has supported the exponential use of information transmission. Thus, secure transmission and storage of private information have received many researchers' attention [1]. Actually, according to a study by the ''Ponemon Institute'' and ''IBM,'' in 2015, data breach average cost was USD 3.79 million, whereas another study by ''Juniper Research'' forecasted that by 2019, cybercrimes would cost about USD 2.1 trillion [2]. As such, many techniques for hiding private and sensitive information in digital carriers have been developed. Hiding this information in images, text, videos, and audio is termed steganography [1].

The method proposed in this paper depends on the following: coverless image steganography, optical mark recognition (OMR), and rule-based machine learning (RBML). So, the rest of the introduction discusses these related topics briefly.

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

A. IMAGE STEGANOGRAPHY

Steganography word consists of two words of Greek origin, ''steganos'' and ''graphien,'' which, when combined, mean ''covered writing'' [1]. Steganography is a method to secure messages during transmission by concealing them within a carrier such as an image, video, text, or audio, which results in stego media [3]. Carriers may also include hiding information in various formats such as codes, DNA, HTML, XML, or executable files (EXE) [1].

By using steganography, the secret message format does not change, and the actual data are maintained. The objectives are to provide end-to-end secure data communication [1], concealment of the communication existence, and personal data protection [4].

The cover medium chosen for embedding must have two features: it should be familiar, and the modifications should be invisible to a third party. To the best of our knowledge, digital images are the most famous carrier in steganography because they contain significant amounts of redundant data and can conceal sensitive data without any visible effects [4]. So, in image steganography, confidential information is exclusively hidden in images [5].

1) GENERAL IMAGE STEGANOGRAPHY PROCEDURE

The primary goal of digital image steganography is the unnoticeable concealment of private or secret data inside a cover image. The secret message type may be image, bits, text, or video files. The embedded data hidden in a carrier image are termed a 'secret message' or 'payload,' and the result is a 'stego-image.' Then stego-image is shared through an insecure channel [1], [4].

For better security, security systems may use an encryption algorithm and optional key throughout the embedding process. This key may contain data such as embedding coefficients, the password used in the encryption process, etc. It must be shared between the sender and recipient [1], [4]. So, image steganography terms are [6]:

- Secret message (payload): sensitive information that is embedded in the carrier image [6],
- Cover image (carrier): an original image that is used as the medium to hold the payload [6],
- Stego-image: resultant image after hiding the payload within the carrier image [6], and
- Stego-key: optional additional information that is used for embedding and extracting the payload [6].

The image steganography system comprises two phases (Figure 1): embedding and extracting the secret message. In the embedding phase, the secret message or payload is hidden in locations selected within the carrier image, depending on the steganography method. Then stego-image is submitted to the recipient [5]. In general, the embedding phase can be represented by this equation [1]:

$$SI = \text{Embedding}\,(CI, \text{Encryption}\,(SM, K1), K2) \quad (1)$$

where SI refers to the stego-image obtained, Embedding(,) refers to embedding function, CI is the cover image, Encryption(,) is an optional encryption function, SM is the secret message, and K1 and K2 are optional keys used for encryption and embedding, respectively. The SI is then sent through a communication channel to the receiver side.

The extraction phase can be represented using the following equation [1]:

$$SM = \text{Decryption}\,(\text{Extraction}\,(SI, K2), K1) \quad (2)$$

where SM is the secret message; Extraction(,) is the extraction function, which is the same as the embedding function but in reverse); SI is the stego-image received at the receiver side; Decryption(,) is the decryption function; and K1 and K2 are optional secret keys used for extraction and decryption, respectively [1].

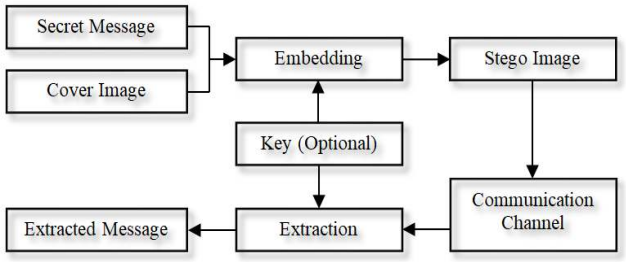


FIGURE 1. The general procedure in image steganography [5].

2) IMAGE STEGANOGRAPHY REQUIREMENTS

Any image steganography system has four essential properties: robustness, capacity, imperceptibility, and security to test its effectiveness [1]. A trade-off exists among these properties; if the payload amount increases, the artifacts' effect increases, and the resistance toward modification decreases. So, all the properties must be maintained at an optimum level [1], as will be discussed next in more detail.

- Imperceptibility (Undetectability): The critical requirement for the image steganography method is undetectability. The strength of any image steganography method lies in embedding the payload in the carrier image so that it is undetectable with the use of statistics or by the naked eye [1], [7]. Imperceptibility/undetectability means how much the stego-image is indistinguishable from the carrier image [4]. Simply, if the stego-image and the cover image are identical, the communication is more secure [1], [7].
- Security: An image steganography method is regarded as secure if the payload is not detectable by statistical means or removable by an attacker after being detected, i.e., how stego-images can resist different steganalysis attacks [4]. Another critical requirement of any image steganography system is the secure transmission of the payload. So, security in avoiding unauthorized information access by a third party while transmitting through an insecure communication channel is crucial [1], [7].
- Payload Capacity: An effective image steganography system aims to transmit the maximum payload using the minimum amount of carrier image. The embedding capacity is the maximum amount of payload embedded compared to the carrier image size. Maintaining a higher embedding rate without destroying image security and undetectability is challenging when hiding data [1], [7].
- Robustness: is the ability to embed and extract the hidden payload from a stego-image even if it is damaged/modified by a third party using image compression, scaling, resizing, etc. Image steganography systems are of low robustness when stego-images are modified via image compression, conversion of the file format, and digital to analog format conversion [1], [7]. Thus, the ideal image steganography method must simultaneously achieve the above objectives: high robustness, good imperceptibility, high capacity, and security [4].

3) IMAGE STEGANOGRAPHY CLASSIFICATIONS

Two types of steganography methods are based on digital images [8]: the spatial domain methods, which hide the secret

message in the carrier image directly through pixel manipulation [9], including methods such as ''pixel-value difference'' (PVD) and ''least significant bit'' (LSB); and the frequency domain methods, which embed payload by modifying coefficients of the transformed image, including ''discrete cosine transform'' (DCT), ''discrete Fourier transform'' (DFT), and ''discrete wavelet transform'' (DWT) [10].

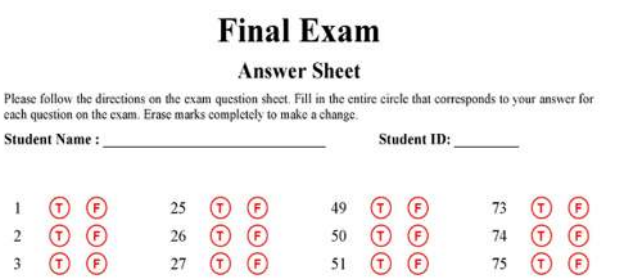
4) COVERLESS IMAGE STEGANOGRAPHY

In almost all traditional image steganography methods, the payload is embedded into cover image pixels, creating modification effects. In these methods, the stego-images may be detected by any image steganalysis tool, and due to this, security cannot be guaranteed. To address this issue, a coverless data hiding concept has been proposed [11]. Coverless data hiding was proposed to resist existing steganalysis tools, first introduced in 2015 [10]. The main idea of this technique is to find images that contain a payload. In coverless data hiding, a mapping relationship exists between the payload and the cover image. Compared with traditional steganography, coverless steganography does not change the cover pixels, such as LSB, PVD, etc. Therefore, the security of coverless steganography methods is higher than traditional steganography methods [11].

Coverless information hiding does not mean that a cover is not required. However, compared with traditional steganography, coverless steganography directly uses the contents of the cover itself to represent the payload [10]. The existing coverless steganography methods can be classified into text-based and image-based methods, depending on the type of cover transmitted. For the text-based type, the current methods mainly search for the texts containing the payload according to specific rules, such as the stego-texts, then determine the location of the secret data using labels [10]. The current image-based methods are similar to image retrieval techniques; these methods use images retrieved from the image database to represent the payload [10].

B. OPTICAL MARK RECOGNITION (OMR)

OMR is an electronic method used to gather human-marked data by identifying specific markings on a bubble sheet document (Figure 2). Usually, this process is achieved with the aid of a special scanner that checks light reflection through the sheet; marked bubbles/circles reflect less light than the blank bubbles/circles, resulting in less reflectivity. Currently, the OMR process is widely used to administer different types of exams [12]. This technology involves collecting data from fill-in-the-bubble sheets like tests, surveys, multiple-choice questions (MCQ) sheets, and true/false (T/F) sheets (Figure 2). OMR enables the respondent/student to choose the answer of a question by filling in a bubble or circle associated with the correct answer. Collecting data using OMR is more precise than handwriting answers' recognition [13]. Although OMR is not a new technology, it has evolved; conventional OMR systems require preprinted


Final Exam

Answer Sheet

Please follow the directions on the exam question sheet. Fill in the entire circle that corresponds to your answer for each question on the exam. Erase marks completely to make a change.

Student Name : _____ Student ID: _____

1	T F	25	T F	49	T F	73	T F
2	T F	26	T F	50	T F	74	T F
3	T F	27	T F	51	T F	75	T F

FIGURE 2. True/false (T/F) OMR bubble sheet.

forms and special scanners. Although OMR is a promising technology, it is expensive and limited to very high volume applications [13].

C. RULE-BASED MACHINE LEARNING

''Machine learning'' (ML) is a form of ''artificial intelligence'' (AI) that enables the system to learn from data instead of through programming [14]. ML is a rapidly growing field that refers to the automated detection of patterns in a dataset, becoming a standard tool in any task that requires information extraction from large data sets. ML-based technology surrounds us, including facial detection in digital cameras and personal assistance apps on smartphones that recognize voice commands [15]. ML is classified into ''supervised learning,'' ''unsupervised learning,'' and ''reinforcement learning'' [14].

The selection of a proper algorithm is both an art and a science. As an example, if two data scientists are solving the same problem, two different algorithms may be chosen. However, understanding different ML algorithms types help identify the best type to be used [14]. These algorithms include Bayesian, clustering, decision tree, dimensionality reduction, instance-based, neural networks, deep learning, linear regression, regularization to avoid overfitting, and rule-based machine learning (RBML).

RBML algorithms describe data using relational rules. This system can be contrasted from ML systems that create a model that can be generally applied to all the incoming data. To summarize, these systems are straightforward: If A is input data, perform B. However, as systems become active, RBML can be too complicated. For example, a system can include more than 100 pre-defined rules, so it is essential to be careful when creating an approach to avoid over-complication [14].

The main contribution of this study is to propose a novel and highly robust coverless image steganography method using the bubble sheet as a cover based on an OMR system and RBML in order to enhance coverless image steganography security, robustness, and capacity.

The remainder of the paper is arranged in this way; Section 2 introduces some related studies on digital image steganography methods: coverless and traditional. Section 3 explains the proposed method in detail. Section 4 presents the proposed method's results then compares them with existing methods. Finally, Section 5 covers the conclusion of the paper.

II. RELATED STUDIES

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

Lee [3] presented adaptive ''least-significant-bit'' (LSB) data hiding for color images in PNG format on smartphones. For each channel—red (R), green (G), and blue (B)—combinations of all-4bit, one-4bit + two-2bit, and two-3bit + one-2bit LSB replacements have been proposed. In the all-4bits method, the method embeds four bits in each pixel of R, G, and B. In one-4bit + two-2bit, the method embeds four bits in one pixel, the other two pixels with two bits only, and so on. The inputs to the method are a color image and the zipped (compressed) data to be hidden. Then, based on the data's size, the number of required pixels in a row is calculated. For each row, the proposed method sort delta_i, which is the difference value of every two consecutive pixels (i.e., $\text{delta}_i = |x_{i+1} - x_i|$) in descending order, then select the chosen pixels and replace the LSBs of chosen pixels with the secret zipped data bits. Also, the size is embedded in a specific head area in the cover image.

St nescu *et al.* [2] proposed a model with the pre-requisite that the sender must have a set of original cover objects *CR*, which are processed in based upon a processing function *fp* to create cover objects *C* that are used in the steganography process. So, the cover objects *C* are obtained after the processing of *CR*. To confuse the attacker, the sender selects and incorporates the payload only in some of the cover objects, which become steganography objects *S*. However, the entire set of cover objects, including those without confidential information, are sent to the receiver. The authors stated that the cover objects are selected with the help of a switch. If the switch is in the ''0'' state, the cover object is sent to the receiver, and it does not contain any payload but will confuse the attacker. If the switch is in the ''1'' state, then the steganography object is transmitted to the receiver. The operation of the switch is controlled according to a function known by both sender and recipient. The changes completed using the processing function are achieved in such a way that the original cover object does not differ too much from the processed object. The transformation that can be applied through the processing function is the noise; as the transformation is applied similarly to all of the original cover objects, the entropy increases for all of them. Finally, the payload is embedded in some of the processed cover objects. Following this step, the complete set of cover objects, steganography objects, and processed cover objects is sent to the recipient.

Li *et al.* [16], an iris image steganography method, was proposed. This method embeds the personal privacy data in an iris image using ''syndrome trellis coding'' (STC)-based data hiding. To minimize the effect of data hiding on iris image recognition quality, the authors proposed a distortion function that assigns a high embedding cost to the critical iris feature region. The method works as follows: In the enrolment phase, personal data are encrypted using an existing algorithm. Then, the cipher data are embedded into a registered iris cover image using a proposed embedding algorithm to obtain the iris stego-image with private data. Finally, the iris stego-image is saved in the database for authentication. Only the iris image database is needed, instead of additional storage space.

Tao *et al.* [17] proposed a robust image steganography framework. First, the channel compressed version of the original image is obtained. Secondly, the payload is hidden into the original compressed image using any JPEG-based steganography method that creates a stego-image after transmission. Then, the compressed image is obtained: to generate the intermediate image, which is the corresponding image before channel transmission, a coefficient adjustment scheme modifies the original image based on the stego-image, and this adjustment is performed so that the channel compressed version of the intermediate image and the stego-image are identical. Finally, after channel transmission, the payload can be fully extracted from stego-image.

Devi *et al.* [18] proposed a method closely related to the method [16], in which some information is embedded in brain MR images, and they checked if the stego-image could still support the same classification accuracy as the original image.

Duan *et al.* [19], proposed a coverless data hiding framework based on a ''generative model'' was proposed. First, the selected secret image is fed as input to the ''generative model database'' to generate another independent and meaning-normal image that is different from the selected secret image. This image is then transmitted to the recipient and fed as the input to the ''generative model database'' to generate another visually similar image as the selected secret image. Thus, only the ''meaning-normal'' image that is not related to the selected secret image is transmitted. Finally, the sender and recipient must share the same dataset and the same parameters.

Cao *et al.* [10], proposed a coverless data hiding framework based on the ''molecular structure images of material''. Firstly, the image is divided into several blocks (sub-images), and then the average pixel values of each block are calculated and used to represent a fragment of the secret information. The secret information is transformed into a binary string then the binary string is segmented into fragments according to the mapping relationship to obtain a binary sequence, which is represented by the calculated average pixel value of sub-images. The proposed method searches the appropriate sub-images to express the secret information. Finally, the authors used a sub-image to represent a fragment of the four-bit binary. Then, labels are used to identify sub-images locations that represent the binary sequence. They also used a ''multi-level inverted index'' structure to improve search efficiency.

Another coverless image steganography method was introduced, Wu *et al.* [20], based on the ''grayscale gradient co-occurrence matrix.'' The secret information is converted into a secret binary stream M with a length L. Then, this stream is divided into eight-bit segments. Secondly, each eight-bit segment is coded using the Turbo encoder. As a result, the length of each segment becomes 16 bits. Thirdly, the sub-pool image that corresponds to the 16 bits length segment is searched for

according to the mapping relationship. Fourthly, the search step is repeated until all the segments are mapped. Fifthly, the image equal to the length of the binary stream is selected, then added to the end of those images. Finally, all images are transferred to the recipient.

Another proposed framework is as follows, Zhou *et al.* [21]: a database of images is created. Then, for each image, a hashing algorithm is used for hash sequence generation. After that, to build an ''inverted index structure,'' all of the database images are indexed according to their hash sequences. So, to transmit the payload, the payload is first transformed into a stream of bits and divided into equal-length segments. For each segment, by searching in the inverted index structure, the image with a hash sequence matching the segment is found. Finally, the series of images that can be referred to as stego-images are obtained and then transmitted to the recipient.

A coverless steganography method based on image hashing has been reported, Zheng *et al.* [22]. Based on (SIFT) feature, the proposed algorithm extracts 18-bits from each image as the robust hash value. So, rather than embedding the secret data into these images, the authors established a relationship between the images and payload using a hash map. Before the secret communication, a local database must be created containing all the images with the same hash values as all 18-bit binary sequences. Finally, the series of these images with the hash sequences the same as the segments are obtained and transmitted to the recipient.

III. PROPOSED METHOD

According to different hiding methods, the commonly used image steganography schemes, either in the frequency or spatial domain, leave modifications to the stego-image that are detectable by any detection algorithm [19]. Detection algorithms, or steganalysis [4], can be identified as payload detection art [2]. Coverless information hiding has been proposed to resist the existing steganalysis tools [10]. Steganalysis can be classified into passive and active methods. For passive methods, the absence or existence of secret data can be identified, whereas active steganalysis can recover, modify, or tamper with the hidden message so that the recipient is unable to extract it [2], [4]. Another unsolved problem in image steganography is the artifacts that could destroy the stego image, such as network transmission errors [2] or image manipulations, such as image filtering, scaling, conversion of file format, digital to analog format conversion, noise, and data compression [23].

The current coverless image steganography methods are similar to image retrieval. These methods use images retrieved from the image library (i.e., a pre-defined local database created at the sender and receiver; both ends) to represent the payload [10]. To the best of our knowledge, these techniques are not very robust, as they can resist some steganalysis tools or algorithms, but not all [10]. Also, they cannot overcome issues due to network transmission errors or image manipulations. As with coverless image steganography, the selection of the image from the dataset to represent the secret message is based on its pixel values, characteristics, or histogram; so, if any pixel changes or the image data are tampered with, changed, or modified due to active steganalysis, image manipulation, or network transmission error, a part or all of the secret message is lost.

Another challenge with the coverless image steganography methods is that these techniques have low embedding capacity [10], mainly because an image can only generate a fixed-length binary stream, as shown in section 2, and the selected image features are not evenly distributed [10].

Any steganography system must be secure and robust against manipulations by an active attacker and against artifacts that could result in secret message loss, such as network transmission errors [2] or image manipulations [23]. Thus, the following properties of coverless image steganography techniques need to be improved: security and robustness against these attacks, overcoming challenges stated above, and greater capacity than the few bits/cover provided in the previously proposed coverless methods. So, to address these problems, a novel coverless image steganography method that is based on OMR and RBML is proposed here.

A. PROPOSED METHOD COMPONENTS

The components of the proposed method are as follows:

- Bubble sheet: Corresponds to the cover image in traditional steganography methods. It is a generated answer/bubble sheet that is used as a carrier.
- Secret message (payload): Sensitive information mapped to the bubble sheet (bubble sheet answers).
- Answered/mapped bubble sheet: Corresponds to stego-image—the bubble sheet obtained after answering/mapping the secret message.
- Mapping algorithm: The embedding algorithm used to answer the bubble sheet based on secret message bits.
- Detection algorithm: The extraction algorithm used to collect and detect the answers from the answered bubble sheet to obtain and form the secret message.

B. SYSTEM (MAPPING AND DETECTION PHASES)

Before the mapping phase, a bubble-sheet template (True/False answer sheet, four columns $\times$ N rows) must be generated only once at the sender side using any free OMR software. In this study, it was created using Microsoft Office Word 2016 and a free OMR font [24]. Then, the generated bubble sheet template is saved as an image in any format, which is the image used instead of the local database currently used in coverless image steganography methods. No image library and no images are required, and no image has to be created again either at the sender or receiver side, as required with previous coverless steganography methods.

1) MAPPING PHASE

Figure 3 shows that the generated bubble sheet and the secret message are fed as inputs to the RBML mapping algorithm (Algorithm 1). Then, the secret message is divided

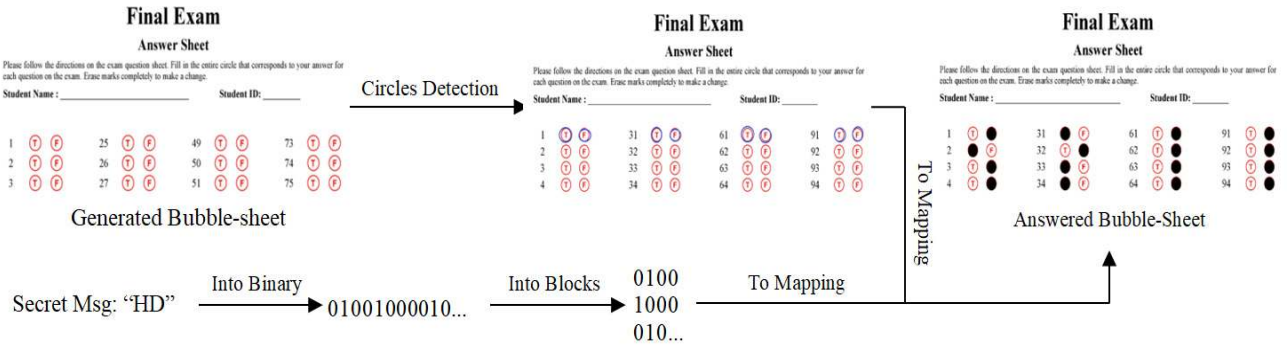


FIGURE 3. RBML mapping example.

into characters and converted into binary stream 0's and 1's. After that, this bitstream is divided into blocks of 4 bits each; each block's length is equal to the number of columns of each row in the generated sheet, which is four columns. Secondly, the RBML algorithm is now created and learned to complete the following tasks: detect and count the number of rows in the bubble sheet, detect all circles found in the generated bubble sheet row by row, calculate the bubble sheet full capacity as each byte (i.e., 8 bits) will be embedded in 2 consecutive rows, and mapping the secret data by marking and answering the bubbles/circles in the bubble sheet, which will be discussed next.

The mapping phase works as follows: the trained ML algorithm takes a block from the secret message (4 bits) and a row from the bubble sheet (4 columns = eight circles), then answers the sheet by marking the bubbles in the current row based on the 4 bits. The bit number corresponds to the question/column number in each row (i.e., bit number 1 is the answer of the first true/false question/column and so on). So, if the first bit is 1, the answer is true and the true circle is detected and marked in the first question/column in the bubble sheet by the RBML algorithm; otherwise, if the first bit is 0, so the answer is false and the false circle will be marked in the first question/column in the bubble sheet by the RBML algorithm. So, the bubble sheet as in Figure 3, contains 4 columns, and each column contains two answers for one question, which are true/false (i.e., 8 bubbles), and depend on the secret message bit, which is 0/1. The ML algorithm should mark the correct bubble/answer as explained above (i.e., mapping). As an example, if the secret message block is 0100, as shown in Figure 3, then the algorithm will circle the false bubble for the first question, circle the true bubble for the second question, then false, and so on, row by row. At the end of the mapping process, the algorithm will solve the whole answer sheet based on the secret message and the answered/mapped bubble-sheet will be ready to be shared (i.e., sent to receiver).

2) DETECTION PHASE

The detection phase works as the mapping phase but in reverse order, as shown in the OMR detection algorithm (Algorithm 2). Firstly, the answered/mapped bubble-sheet is scanned row by row, and the circles are detected for each row. Then, answers are collected, and secret bits are obtained and concatenated to form the bitstream. After, the bitstream is represented by 8 bits blocks, and these blocks are converted into characters using ASCII. Finally, these obtained characters are joined together, forming the secret message.

C. ALGORITHMS

All the operations of the RBML mapping algorithm (Algorithm 1) are completed at the sender side. The steps of the OMR detection algorithm (Algorithm 2) to detect and collect the answers (i.e., secret bits) are completed on the receiver's side.

Algorithm 1 RBML Mapping Algorithm

Input: Bubble Sheet (BS), Secret Message (SM).
Output: Mapped Bubble Sheet (MBS).
1. Message 'SM' is split onto characters; SM= {sm_1, sm_2, sm_3, …, sm_n}.
2. Convert message characters into a stream of bits 'SoB' based upon ASCII standards; SoB= {0's and 1's}.
3. Divide 'SoB' into blocks of 4 bits; Blk= {blk_1, blk_2, blk_3 …blk_n}.
4. Detect and count the number of rows in bubble sheet 'R'.
5. Detect all bubbles in each row 'RB', RB= {rb_1, rb_2, rb_3 …rb_n}.
6. **For i**= 1 to length (Blk)
7. BS= MarkBubbles(BS, blk_i, rb_i) //Mapping Step
8. **End For**
9. MBS. = BS
10. Return Mapped/Marked Bubble Sheet 'MBS'.

Func. BubbleSheet=MarkBubbles (BubbleSheet, Block, RowBubbles)

1. For i=1 to 4
2. If (Block(i)==1)
3. Circle RowBubbles(i, 1) found in BubbleSheet //Answer is True
4. Else
5. Circle RowBubbles(i, 2) found in BubbleSheet //Answer is False
6. End if
7. End For
8. Return BubbleSheet
End Function

Algorithm 2 OMR Detection Algorithm

Input: Mapped Bubble Sheet (MBS).
Output: Secret Message (SM).
1. Detect/count number of rows 'R' in Mapped Bubble Sheet 'MBS.'
2. Detect all bubbles in each row; RB= $\{rb_1, rb_2, rb_3 \ldots rb_n\}$.
3. **For i**= 1 to length (RB)
4. blk_i =OMRScanBubbles(rb_i) //Detection Step
5. **End For**
6. Combine all detected blocks Blk= $\{blk_1, blk_2, blk_3 \ldots blk_n\}$.
7. Create stream of secret Msg bits by concatenating 'Blk' elements.
8. Divide and convert stream 'SoB' back into secret message characters 'sm' using ASCII.
11. Join all 'sm' characters to create Secret Message 'SM', SM= $\{sm_1, sm_2, sm_3, \ldots, sm_n\}$.
12. Return Secret Message 'SM'

Function Block = OMRScanBubbles(RowBubbles)

1. BitCounter=0
2. For i=1 to 4
3. If (RowBubbles(i, 1) is Marked)
4. Block(BitCounter)=1 // True Answer Detected
5. BitCounter= BitCounter + 1
6. Else if (RowBubbles(i, 2) is Marked)
7. Block(BitCounter)= 0 // False Answer Detected
8. BitCounter= BitCounter + 1
9. End if
10. End For
11. Return Block

End Function

IV. EVALUATION AND COMPARISONS

Upgrading the properties of coverless image steganography, such as enhancing capacity, security, imperceptibility, and robustness, is this paper's main contribution. So, the proposed method was experimentally evaluated based on these main four essential properties to assess its effectiveness and to verify and evaluate its efficiency. The results were used to compare the effectiveness of the proposed method with existing coverless image steganography methods.

A. CAPACITY

As shown in Tables 1 and 2, the proposed method's hiding capacity is the highest amongst the available methods. Notably, the proposed method capacity is not limited to 120 bits/cover, as more bubbles can be added until the sheet capacity according to the actual needs. In this experiment, the sheet was four columns × 30 rows.

TABLE 1. Proposed method embedding capacity.

Method	Capacity (bits/carrier)
Z. Zhou et al. [21]	8
C. Yuan et al. [25]	8
S. Zheng et al. [22]	18
Y. Cao et al. [10]	36
L. Zou et al. [11]	80
Proposed Method	120

TABLE 2. The number of images needed when the same data are hidden [22].

Method	Secret Message Length			
	1 byte	10 bytes	100 bytes	1 kilobyte (kB)
Z. Zhou et al. [21]	1	10	100	1024
S. Zheng et al. [22]	2	6	46	457
Proposed Method	1	1	7	68

Table 1 shows that the proposed method's hiding capacity is the highest among the previously proposed coverless image steganography methods, 120 bits/cover; this means that the proposed method has significant embedding capacity, increasing the hiding capacity of the coverless image steganography technique. As stated above, the bubble sheet may contain even more bubbles to represent more binary bits.

Table 2 shows that the number of images required to hold the same secret message is the lowest among other methods, which means a smaller number of covers needed. For example, to map/hide one kB of secret data, Zhou *et al.* [21] required 1024 images, Zheng *et al.* [22] required 457 images, and I required only 68 images. As each bubble sheet can map up to 15 bytes (120 bits), mapping 1024 bytes only requires 68 images.

B. ROBUSTNESS

Robustness is the steganography method's ability to resist attacks. In the secret data/payload transmission process, algorithm failure is caused by various attacks such as noise attacks, scaling attacks, and JPEG compression. The robustness of the coverless proposed method was tested, evaluated, and verified through experiments and comparisons [20].

First, a bit error rate (BER) must be defined. BER is a robustness measure of the steganography algorithm in the communication process, calculated as follows [20]:

$$\mathrm{BER} = e/n,\ e = \sum p_i \oplus q_i;\ \text{Where} i = 1 : n \qquad (3)$$

E represents the number of errors found, n is the total number of bits, p is the original secret bits vector before the attack, and q is the secret bits vector after being attacked. If BER = 0, no errors were found, and the secret bits were successfully extracted with 100% accuracy, which means that the method is 100% robust to this attack. Otherwise, if BER > 0, an error rate exists in the extracted secret bits after the attack (i.e., some secret bits have been changed/damaged), which means the method is not 100% robust to this attack.

1) JPEG COMPRESSION ATTACK

JPEG is the compression standard for still images and is the most popular method. JPEG is a lossy compression that allows data to be lost and applied to images before/during transmission [20]. The stego-image is modified/damaged

through transmission if it is compressed. The proposed method robustness (BER) was measured against the JPEG attack. The JPEG quality factor ranges from 1 to 100, whereas the highest compression ratio = 1 and the lowest compression ratio = 100.

TABLE 3. 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
70	0.038	0.080	0.009	0.08	0.002	0
50	0.151	0.146	0.146	-	0.007	0

Table 3 compares the CBD, CBZS, CSD, CIHRIH, Wu *et al.* methods [20], and the proposed method. The proposed method performed the best with 0 BER for the same compression values, which means the proposed algorithm is 100% robust to JPEG compression attacks at these values and the secret message extraction accuracy is 100%.

2) NOISE ATTACK

"Salt and pepper" noise is randomly distributed through an image, but its depth is fixed. In general, two kinds of noise, salt, and pepper, show up simultaneously in the image. The proposed method was analyzed for noise with densities ranging from 0.01 to 0.04 in 0.01 increments.

TABLE 4. Comparison of BER after noise attack.

Noise Density	CIHWE [21]	CIHRIH [22]	Wu et al. [20]	Proposed (Single Layer)
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

Table 4 compares the BER among the following methods: CIHWE [21], CIHRIH [22], Wu *et al.* [20], and the proposed method after attack by "salt and pepper" noise. In the proposed method experiment, salt and pepper were applied on a single layer only from the image, as other methods use greyscale images, which are single layer images. So, the results showed that the proposed method had a zero BER at the same noise density, which means the proposed algorithm is 100% robust to "salt and pepper" noise attacks at these densities if applied to a single layer from the image. I attempted to salt and pepper the full image (i.e., the three R, G, and B layers), but the proposed method failed to detect the data, as the ML algorithm could not detect the circles due to the noise.

3) SCALING ATTACK

Stego-image scaling can destroy the extracted secret message. The scaling attack results are listed in Table 5.

Table 5 compares the RCIS, Wu *et al.* method [20], and proposed method. The results showed that at a 0.3 scaling ratio, the proposed method has three error bits out of the 120 bits. The BER was 0.02500. This error was due to the tiny size of the circles that the ML algorithm could not detect the circles. The BER of the proposed method at the 0.5, 0.75, and 1.5 scales was 0, which means the method is 100% robust, and the message was fully detected successfully.

TABLE 5. Comparison of BER after scaling attack.

Ratio of scaling	RCIS [26]	Wu et al. [20]	Proposed method
0.3	0.146	0.015	0.025000
0.5	0.057	0.009	0
0.75	0.039	0.002	0
1.5	0.016	0.025	0

4) OTHER ATTACKS

The results of applying some attacks that are hard to resist by almost all image steganography methods, including color space conversion (i.e., red, green, blue (RGB) to greyscale conversion), thresholding (i.e., RGB to binary image conversion), conversion of file format, and digital into analog format conversion (i.e., print and scan stego-image) are listed in Table 6.

TABLE 6. BER of the proposed method after different attacks.

Attack					Proposed Method			
RGB to greyscale conversion					0			
RGB to Binary Conversion					0			
File Format Conversion	.bmp	.jpg	.tiff	.png	0	0	0	0
Digital–Analog-Digital Conversion					0			

As shown in Table 6, the proposed method succeeded in resisting the above attacks, and the BER was 0, which means 100% robust to these types of attacks. In the print-scan attack, the used scanner should be of acceptable quality, and the printed sheet should be appropriately placed on the scanner lens (i.e., not rotated, not shifted, etc.).

C. SECURITY

Security is compromised when communication is monitored. If the attacker wants to obtain the payload, they have to identify the existence of the communication and read the secret information [20]. In the proposed method, there is no hidden message and no stego-image; it is only a student's answered bubble-sheet for a final exam, which is not suspicious. Thus, the proposed method provides a high-security level, and is hard for an attacker to detect the payload.

1) STEGANALYSIS ATTACK

Data hiding methods should resist various steganalysis attacks. Unfortunately, almost all the existing image steganography methods are detectable by steganalysis tools that use the pixel bits changes resulting from the hiding operations applied in steganography methods [21]. However, these tools cannot efficiently detect coverless steganography methods because without employing a cover image for embedding the secret data, the proposed method directly finds the circle

and marks it to map the secret bits. Notably, OMR sheets are not suspicious as they are used in exams and surveys, and they have not been previously used as cover images.

V. CONCLUSION

In this paper, a highly robust, highly secure, and highly embedding capacity coverless image steganography method based on OMR and RBML was proposed. In the proposed method, the secret information (payload) is represented by a binary string. Then, the bubble sheet is used to represent binary fragments according to a mapping function. Finally, the generated mapped bubble sheet is the answered version that is sent to the receiver. The proposed method depends on the rule-based machine learning (RBML) and optical mark recognition (OMR) algorithms. The RBML algorithm was first developed and then learned to simulate a student's behavior in his final bubble-sheet T/F exam, detecting the circles and marking them based on the correct answers. The secret bit is taken as the correct answer of a question; the algorithm detects its corresponding question number, then marks its circle based on the given correct answer to solve the question and map the secret bit.

The second algorithm was the OMR, which is used in the detection phase. OMR systems require special scanners, which are expensive. An OMR algorithm, which is free, was developed instead of using special expensive scanners. The OMR algorithm, with the help of the RBML algorithm that was developed in the mapping phase, is used to collect the correct answers of the student from the mapped bubble sheet, as these answers will be the secret message that was mapped in the mapping phase.

Finally, the proposed coverless steganography method has some advantages compared to current coverless methods:

- No database is required. No database is needed for cover images, neither at the sender side nor the receiver side, compared with previously reported coverless methods.
- No secret information sharing required. No information has to be shared between the sender and receiver, such as databases, labels, bit locations, etc.
- No time wasted in searching. Almost all previous methods have to search within a database for the required image. Searching for images in databases can be slow, depending on the size of the database.
- Very high capacity. As shown in Tables 1 and 2, the proposed method has the highest embedding capacity among the previously proposed methods. Also, embedding capacity can be increase as required.
- Very high robustness. The experiments showed that the method has very high robustness, as shown in Tables 3 to 6, and can resist different attacks as scaling, color space conversion, JPEG compression, thresholding, ''salt and pepper'' noise, file format conversion, converting from digital into analog format, and steganalysis tools.
- Very high security as bubble sheets are not suspicious and have not been previously used as a cover file.
- The proposed method applies machine learning algorithms in steganography, which were previously used in steganalysis. In the OMR phase, no special or expensive tools are required.

ACKNOWLEDGMENT

The authors are thankful of the Taif University. Taif University researchers supporting project number (TURSP-2020/160), Taif University, Taif, Saudi Arabia.

REFERENCES

[1] I. J. Kadhim, P. Premaratne, P. J. Vial, and B. Halloran, ''Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research,'' *Neurocomputing*, vol. 335, pp. 299–326, Mar. 2019.
[2] D. Stănescu, M. Stratulat, R. Negrea, and I. Ghergulescu, ''Cover processing-based steganographic model with improved security,'' *Acta Polytech. Hungarica*, vol. 16, no. 1, pp. 227–246, 2019.
[3] H. Lee, ''Data hiding in spatial color images on smartphones by adaptive R-G-B LSB replacement,'' *IEICE Trans. Inf. Syst.*, vol. E101.D, no. 8, pp. 2163–2167, 2018.
[4] M. Hussain, A. W. A. Wahab, Y. I. B. Idris, A. T. S. Ho, and K.-H. Jung, ''Image steganography in spatial domain: A survey,'' *Signal Process., Image Commun.*, vol. 65, pp. 46–66, Jul. 2018.
[5] E. A. Abbood, R. M. Neamah, and S. Abdulkadhm, ''Text in image hiding using developed LSB and random method,'' *Int. J. Electr. Comput. Eng.*, vol. 8, no. 4, pp. 2091–2097, 2018.
[6] X. Huan, H. Zhou, and J. Zhong, ''LSB based image steganography by using the fast marching method,'' *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 3, pp. 1–5, 2019.
[7] N. Dey, A. S. Ashour, and S. Acharjee, *Applied Video Processing in Surveillance and Monitoring Systems*, vol. I. Hershey, PA, USA: IGI Global, 2016.
[8] G. Swain, ''Digital image steganography using variable length group of bits substitution,'' *Procedia Comput. Sci.*, vol. 85, pp. 31–38, Jan. 2016.
[9] R. Shanthakumari and S. Malliga, ''Dual-layer security of image steganography based on IDEA and LSBG algorithm in the cloud environment,'' *Sādhanā*, vol. 44, no. 5, pp. 1–12, May 2019.
[10] Y. Cao, Z. Zhou, X. Sun, and C. Gao, ''Coverless information hiding based on the molecular structure images of material,'' *Comput., Mater. Continua*, vol. 54, no. 2, pp. 197–207, 2018.
[11] 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 Appl.*, vol. 78, no. 7, pp. 7965–7980, Apr. 2019.
[12] *OMR*. Accessed: Aug. 19, 2019. [Online]. Available: https://www.quora.com/What-is-OMR
[13] *OMR*. Accessed: Aug. 19, 2019. [Online]. Available: https://remarksoftware.com/omr-technology/what-is-omr-optical-mark-recognition/
[14] J. Hurwitz and D. Kirsch, *Machine Learning for Dummies*. Hoboken, NJ, USA: Wiley, Jun. 2018.
[15] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*, vol. 9781107057. New York, NY, USA: Cambridge Univ. Press, 2014.
[16] S. Li, X. Chen, Z. Wang, Z. Qian, and X. Zhang, ''Data hiding in iris image for privacy protection,'' *IETE Tech. Rev.*, vol. 35, no. sup1, pp. 34–41, Dec. 2018.
[17] J. Tao, S. Li, X. Zhang, and Z. Wang, ''Towards robust image steganography,'' *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 2, pp. 594–600, Feb. 2019.
[18] S. Devi, M. N. Sahoo, K. Muhammad, W. Ding, and S. Bakshi, ''Hiding medical information in brain MR images without affecting accuracy of classifying pathological brain,'' *Future Gener. Comput. Syst.*, vol. 99, pp. 235–246, Oct. 2019.
[19] X. Duan, H. Song, C. Qin, and M. K. Khan, ''Coverless steganography for digital images based on a generative model,'' *Comput., Mater. Continua*, vol. 55, no. 3, pp. 483–493, Jul. 2018.
[20] J. Wu, Y. Liu, Z. Dai, Z. Kang, S. Rahbar, and Y. Jia, ''A coverless information hiding algorithm based on grayscale gradient co-occurrence matrix,'' *IETE Tech. Rev.*, vol. 4602, no. sup1, pp. 22–33, 2019.
[21] Z. Zhou, H. Sun, R. Harit, X. Chen, and X. Sun, ''Coverless image steganography without embedding,'' in *Proc. ICCCS*, vol. 1. Cham, Switzerland: Springer, 2015, pp. 123–132.

[22] S. Zheng, L. Wang, B. Ling, and D. Hu, "Coverless information hiding based on robust image hashing," in *Proc. ICIC*, vol. 1. Cham, Switzerland: Springer, 2017, pp. 536–547.
[23] A. Yahya, *Steganography Techniques for Digital Images*. Cham, Switzerland: Springer, 2019.
[24] *OMR Free Font*. Accessed: Aug. 19, 2019. [Online]. Available: https://remarksoftware.com/support/office/form-design/fonts/
[25] C. Yuan, Z. Xia, and X. Sun, "Coverless image steganography based on SIFT and BOF," *J. Internet Technol.*, vol. 18, no. 2, pp. 435–442, Mar. 2017.
[26] X. Zhang, F. Peng, and M. Long, "Robust coverless image steganography based on DCT and LDA topic classification," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3223–3238, Dec. 2018.

AL HUSSIEN S. SAAD was born in Minya, Egypt, in 1985. He received the B.Sc. degree in computer science from the Computer Science Department, Faculty of Science, Minia University, Minya, in 2006, and the M.Sc. and Ph.D. degrees in image steganography from Minia University, in 2012 and 2015, respectively. In 2006, he joined the High Institute for Engineering and Technology (H.I.E.T), Minya, as a Demonstrator, where he became an Assistant Lecturer, in 2013. He joined the Computer Science Department, Faculty of Science, Minia University, as an Assistant Lecturer in 2013. He is currently a Supervisor of M.Sc. and Ph.D. students. He is the author of more than 20 scientific articles in steganography in refereed journals and international conferences and a book chapter. His current research interests include traditional steganography, coverless steganography, data hiding and security, and the Internet of Things (IoT).

M. S. MOHAMED received the M.Sc. and Ph.D. degrees in mathematics from the Faculty of Science, Al-Azhar University, Cairo, Egypt, in 2002 and 2007, respectively. He is currently a Full Professor of mathematics with the Department of Mathematics, Faculty of Science, Al-Azhar University. He has supervised and examined some of M.Sc. and Ph.D. degree students. He is the author of more than 70 scientific articles and two textbooks in refereed journals and international conferences. His research interests include theory of differential equations and its application, numerical analysis, modeling, and numerical and semi-analytical and computational methods for solving differential equations.

ESLAM H. HAFEZ was born in Cairo, Egypt, in 1986. He received the B.Sc. degree in mathematical statistics from the Mathematics Department, Faculty of Science, Helwan University, Helwan, Egypt, in 2007, and the M.Sc. and Ph.D. degrees in order statistics from Helwan University, in 2013 and 2018, respectively. In 2007, he joined the Faculty of Science, Helwan University, where he became an Assistant Lecturer, in 2013.