Securing Digital Data: A New Edge Detection and XOR Coding Approach for Imperceptible Image Steganography

Hayat Al-Dmour
Faculty of Information Technology
Mutah University
Mu'tah, Karak, Jordan

***Abstract*—The rapid progress of digital devices and technology, coupled with the emergence of the internet has amplified the risks and perils associated with malicious attacks. Consequently, it becomes crucial to protect valuable information transmitted through the internet. Steganography is a tried-and-true technique for hiding information beneath digital content, such as pictures, texts, audio, and video. Various methodologies of image steganography have been developed recently. In image recognition, edge detection secures an image into well-defined areas. This paper introduces a novel image steganography algorithm with edge detection and XOR coding techniques. The proposed approach aims to conceal a confidential message within the spatial domain of the original image. In contrast to uniform regions, the Human Visual System (HVS) is less responsive to variations in the sharp areas; an edge detection algorithm is applied to identify edge pixels. Furthermore, to enhance the efficiency and reduce the embedding impact, XOR operation has been utilized to embed the secret message in the Least Significant Bit (LSB). According to the results of the experiments, the proposed method embeds confidential data without causing noticeable modifications to the stego image. The proposed method system produced imperceptible stego images with minimal embedding distortions compared to existing methods. Based on the results, the proposed approach outperforms the conventional methods regarding image distortion techniques. The PSNR values achieved by the proposed method are higher than the acceptable level.**

***Keywords*—Steganography; information hidings; bits modification; decoding algorithm; edge detection; canny edge detection; human visual system**

I. Introduction

Data security is considered one of the most noteworthy factors of Information and Communication Technology (ICT) due to the rapid growth in electronic technologies and the internet. Therefore, a necessary prevention mechanism is needed to protect the data securely. Commonly, sensitive information can be protected either by using cryptography or Steganography. The technique of cryptography conceals the contents of sensitive data to unreadable text by several transformations; however, Steganography is the art of concealing data in an explicit transport file in such a manner that unapproved third parties find it challenging to discover and retrieve the concealed data [1]. An image is represented as an array of numbers corresponding to the light intensities at different points, known as pixels. These pixels collectively form the raster data of the image [2]. At the same time, digital images are the most common form of cover object used for Steganography, as they are the most prevalent carrier on the internet [3]. Image steganographic methodologies can be classified into the Spatial domain and Transform domain methods. In the Spatial domain, the intensity of the pixels is used to implant information. In contrast, information is embedded in the frequency domain of the previously transformed images in the Transform domain. In image steganography, confidential information is protected from malicious attacks by changing the pixels, and the modifications applied to the image are made unnoticeable. The original image without sensitive information is called the cover image, while the cover image with secret information embedded in it is called the stego image. Steganography requires two files: the message and the cover image, which conceal the message. Digital images are preferred over videos due to their compact and smaller size compared to the large and redundant size of the videos when transmitted over low bandwidth networks [4].

The image's detailed contents, information, and features cannot be observed by the Human Visual System (HVS) or the naked eye. Therefore, the use of techniques becomes necessary to check whether an image is an original or a stego image. It is essential to evaluate the strengths and weaknesses of steganography methods based on various characteristics [5]. Some important requirements conflict with each other to develop a reliable steganography algorithm. The three fundamental characteristics of image steganography are capacity, robustness, and imperceptibility. A cover medium's capacity refers to how many bits it can contain. Imperceptibility and robustness are standard requirements that conflict with embedding capacity. Imperceptibility refers to the quality of the stego carrier. The steganography algorithm satisfies the imperceptibility requirement, even though the stego carrier content may differ from the original one if that difference cannot be noticed by the human visual system (HVS) [6]. The imperceptibility is usually computed by the Peak Signal Noise Ratio (PSNR). Greater PSNR means higher imperceptibility. Robustness relates to the strength of the stego medium to endure different types of manipulation. It is, therefore, hard for attackers to illegally modify or remove the embedded secret data [7].

Different image steganography techniques are used to embed the information in the cover media, i.e., spatial and transform domain techniques. The Least Significant Bit (LSB) replacement is one such approach that seeks to replace the least significant bit of each pixel with the associated concealed data

pixel. As a result, such a pixel's initial magnitude changes by 0 or 1, a tendency repeated throughout the cover picture. Least Significant Bit (LSB) is the most widely used method for image steganography based on spatial domain techniques. In this method, the message is embedded in LSB directly [8].

The research problem at hand revolves around enhancing image steganography techniques to embed data securely while maintaining imperceptibility and robustness. This research aims to answer the following questions:

- How can image steganography methods be improved to enhance imperceptibility without compromising capacity?
- What novel approaches can be developed to ensure secure and undetectable data embedding in images?

The main contribution of this study lies in the proposition of an image steganography method based on edge detection, guaranteeing identical edge images in the original and stego images while securely embedding messages. This approach ensures that the concealed message can be correctly extracted from the stego image. The results of the experiments show that the proposed method embeds secret data without causing noticeable modifications to the stego image.

The rest of this work is structured as follows. Section II presents the most recent steganography techniques. The proposed scheme is explained in Section III. Section IV reports on the experimental results, analysis, and discussion. Section V brings the paper to a conclusion.

II. Steganography and its Techniques

Steganography is a scientific discipline that involves concealing data within another form of data. For example, it can include hiding a plaintext message within an image file. Throughout history, various entities such as individuals, the military, secret intelligence agencies, and governments have leveraged Steganography to covertly communicate and transmit information without arousing suspicion. Steganography has a wide range of uses, including confidential communication, electronic watermarking, reliability of data, copyright protection, and identifying manipulation of data [9]. Extensive research work has been carried out to develop digital image steganography. The LSB technique was one of the first to obscure data transmission by embedding secret data into unimportant bits of pixels. LSB approaches, in general, substitute the identical length bits in each underlying pixel with the embedding data. Nevertheless, not all pixels in the image can have equal levels of alteration without producing apparent distortion [10]. The stego image resembles the original image because altering the LSB of such a pixel does not significantly alter the color. Despite this, not all pixels in an image are capable of enduring equal amounts of modifications without noticeable distortion resulting in a low-quality stego image. To handle this issue, some image steganography methods based on LSB have used HVS features to conceal the secret bits in the cover image [11].

Image Steganography is achieved by conducting an XOR operation on the bits of pixel values. In this regard, Joshi et al., retrieved the two rightmost LSBs and two leftmost MSBs of the pixel value using their method. They combined the first and second bits using the XOR operation. The message bit 0 was concealed in the LSB of the pixel value if the result of the two XOR operations was 11 or 00, and the message bit 1 was hidden if the impact of the two XOR operations was 10 or 01 [12].

XOR is the logical operation performed on the LSB and MSB based on the technique. Based on the result of the XOR operation, the message is embedded in the LSB of a particular pixel. Baek et al., suggested a method for embedding the information in the grayscale image to share it secretly. They used the XOR operation to represent the bits at a specific location in the image [13].

Further, a previous study proposed a complicated method for image steganography to hide information in the LSB area of an image pixel. The authors used the XOR operation three times before embedding the message in the LSB. XOR operation was performed on the three MSB bits. This operation behaved as a key to embedding a message in an image. Better security was provided using this simple operation. A study proposed an alpha-trimmed mean filter to enhance the image quality, whereas they used XOR operation on 6-MSBs to add two bits of secret message in the image at 2-LSBs [12].

Additional options for concealing the information included two levels of encryption and an obfuscation phase. Two XOR operations and a private key were used to encrypt the data. The LSB method was then used to incorporate the information in the cover picture. One straightforward XOR-based process selected the colors using a sequencing technique and modified several LSB methods. Three MSBs were utilized as the key in the steganographic procedure, and they employed a triple XOR literary content that needed to be delivered [14].

The least significant bit of an image is subjected to an XOR procedure. When an 8-bit random key is used in the application, pixel 1 from the red matrix's second bit is XORed with the pixel. Because the result of the XOR of the taken bit 1 and the taken bit 0 is 1, the pixel must be satisfied by delivering an encrypted message that conceals the value of the pixel's first LSB bit. The pixel will pass if the XOR operation returns a result of 0, but the following step's process will continue using the same 8-bit random key. Based on the length of the encrypted message, this procedure will continue [15].

To find the secret message, bits from a pixel were extracted and saved in an encrypted message. In the recovery process, pixel 1 of the second bit for the red matrix and the 8-bit random key is XORed with the pixel, and the result of the XOR of bits 1 and 0 is 1. Consequently, the pixel must convey an encrypted message buried in the value from its first LSB. The pixel will pass if the supplied response during the XOR operation is 0, but the following step in the procedure will still use the same 8-bit random key. Following the completion of this phase, the decrypted image is extracted from the stego-image. These bits are taken into the LSB of the identical pixel shown in the stego-image. This process continues till the length of the message is sent [16].

III. Materials and Methods

Compared to smooth regions, the human visual system is less sensitive to modification in image regions with sharp transitions. To attain undetectable Steganography, the secret

message has to be embedded in the edge regions of the cover image. Regardless of how insignificant the changes are in the cover image, conventional edge detection methods produce sensitive edge images. Since hiding the message might cause some alterations to the cover image, this feature restricts the implementation of image steganography based on edge detection. The edge detection technique is commonly used in digital images to determine if each pixel has a high or low spatial frequency [17]. It is the technique of finding locations in a computer image where the image brightness swiftly changes, for example, pixels diverging from minimal intensities to high intensities or the other way around, displaying certain discontinuities [18]. Therefore, this paper presents a new image steganography method based on edge detection that produces identical edge images in the original and stego images. Sobel and Canny edge detection methods are used to extract edges. The Canny edge detection method was created by John Canny in 1986. It is one of the most efficient and well-known [19]. Therefore, Sobel and Canny edge detection method gives the identical edges of both the original and stego-image. In this way, the concealed message can be correctly extracted from the stego image.

In the proposed method, three bits of the message are embedded in a grayscale image intended for transmission to the receiver. The colored image has three channels: Red, Green, and Blue, each of matrix length and width of image size. In the case presented, the input image is in a grayscale where only one channel represents the image in one matrix. The method is centered on the advantage of XOR operation. The XOR operation is performed on 4-pixel values to get 3 XORed results. Three bits of message XORed with four least significant bits of the image to embed 3 bits of message in stego image. The block diagram of the proposed methodology is given in Fig. 1. The steps of the proposed image steganography algorithm are as follows:

Step 1: Convert the image into grayscale.

Step 2: Resize the image into 512 x 512 to get a uniform image size.

Step 3: Perform Edge detection on the image using Canny or Sobel edge detection methods.

Step 4: Convert the message (secret message to be sent) into decimal using the American Standard Code for Information Interchange (ASCII) code character by character.

Step 5: Convert decimal ASCII codes for each character into a binary format where ASCII-encoded data is of 8-bit length.

Step 6: Iterate over 3 bits of the message (secret message to be sent).

Step 7: Get 4-pixel locations in the cover image from the identified edges, i.e., P_1, P_2, P_3, and P_4, where P_1, P_2, P_3, and P_4 are the pixel 4-LSBs of the cover image.

Step 8: Calculate the XOR operation on P_1 and P_2, P_3 and P_4 and P_1 and P_3 as follows:

$$k_1 = P_1 \oplus P_2 \ , \ k_2 = P_3 \oplus P_4 \ , \quad k_3 = P_1 \oplus P_3$$

Step 9: Embed message bits into the stego image based on the XOR calculated by comparing the three estimated bits, $k1$, $k2$, and $k3$, with the three secret message bits according to Table I.

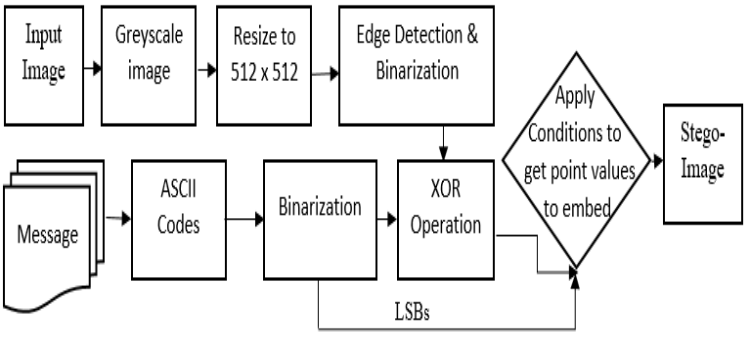

Fig. 1. Block diagram of the proposed methodology.

Step 10: Convert the image matrix from binary to decimal to get the stego image matrix.

In the first step, the RGB image is converted to grayscale. After that, the cover image is resized to a fixed size of dimensions 512×512, where the image contains 512 rows and 512 columns. Then, Canny or Sobel edge detection methods are performed to detect edges in the image.

Edge detection is the technique for identifying points in digital images with discontinuities. It is an abrupt change in the brightness of the image. The point where this change occurs is the edge. The four groups of edges are created that represent the place where the message has to be embedded. The location of four points, P_1, P_2, P_3, and P_4, are extracted from the edges. Fig. 2 shows an example of edge detection. The edges are calculated to store the message and the position for steganographic content (secret message) is calculated as X in Fig. 2. The position of x is calculated using the location of edge detection points, and then XOR these points with each other. Edge detection is performed using two types of edge detection methods: Sobel filter and canny edge detection, for images to use for XOR coding in image steganography.

The Sobel filter is an edge detection method for digital images. It is also called Sobel Feldman sometimes. Edges are the discontinuities in the image that cause a rapid change in intensity value, as discussed earlier. It uses a filter in both vertical and horizontal directions, producing thick and bright edges in every direction. It uses gradient operation. The estimated magnitude of gradient operation is measured using the sum of specific values of slope in two directions: horizontal and vertical.

Canny edge detection is the method for detecting edges in the image. It uses three criteria to detect edge detection performance: localization precision, SNR, and single-edge response precision. It produced the best results in many problems.

In order to send secret messages, steganographic codes are converted into ASCII codes. Subsequently, the binary data was obtained by converting it into binary codes (ASCII), enabling the application of embedding using the suggested XOR approach. Just on spots located using the edge detection technique, XOR was used. The process begins with the points $P1$, $P2$, $P3$, and $P4$, and then XOR is performed to

101	103	106	109	99	98
100	102	103	70	80	70
99	60	103	71	83	72
50	61	80	79	85	71
51	80	82	80	79	73

(a) a

0	0	0	0	0	0
0	0	0	1	1	1
0	1	0	x	x	x
1	x	x	0	0	0
x	0	0	0	0	0

(b) b

Fig. 2. (a) Intensity values of the input image (b) Stego bit positions identification using edge detection.

TABLE I. CONDITIONS TO EMBED THE MESSAGE

Condition Description	Action
All m bits match k bits	-
if m3 does not match k3	$\bar{P}3$ and $\bar{P}4$
if m2 does not match k2	$\bar{P}4$
m2 does not match k2 and m3 does not match k3	$\bar{P}3$
if m1 does not match k1	$\bar{P}2$
m1 does not match k1 and m3 does not match k3	$\bar{P}1$
m1 does not match k1 and m2 does not match k2	$\bar{P}2$ and $\bar{P}4$
None of the m bits match their corresponding k bits	$\bar{P}1$ and $\bar{P}4$

$P1$, $P2$, $P3$, and $P4$ and recorded the results in $k1$, $k2$, and $k3$, where $k1$, $k2$, and $k3$ are indeed the XORed results of the estimated edge point.

Based on the above conditions, shown in Table I, three message bits are embedded into four-point locations that were extracted using the edge detection method. Hence, three message bits are embedded into the image, showing the average embedding is 1.25 bits.

To extract the messages from steganographic images, it starts with calculating the edge based on the same method used during the embedding process. The points were identified using edge detection, as shown in Fig. 3. For example, a total of 1000 edges are presented in the file, enabling the embedding of three bits in four-point locations, thereby accommodating 750 bits within 1000 edges. Nonetheless, 2 bits are used to predict the threshold and 1 bit for data embedding. In every position, the first two bits are used to indicate the threshold, and then the next bit is used to embed the data; using this way, the 1000 locations will get reduced to 3 times = 333 locations. Now in 333 locations, the data is embedded, which reduces the embedding rate. Conversely, it enhances data corruption if any attack on the image happens.

As the locations chosen for embedding the data are not employed to predict the edges, this technique enhances the extraction process. These are the edge points where the XOR operation was performed to embed the secret messages. Once these points are calculated and represented to get m1, an XOR operation is performed on q1 and q2. To get m2, the XOR operation is performed on q3 and q2. To acquire m3, the XOR operation is conducted on q1 and q3. The block diagram for extracting the message is shown in Fig. 3.

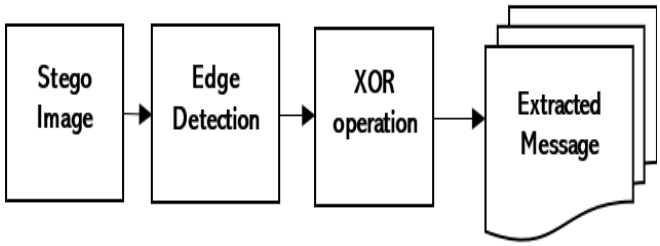

Fig. 3. Block diagram for message extraction.

IV. RESULTS AND DISCUSSION

Distortion evaluation methods are the methods that measure the distortion in the image by comparing the original image and the steganographic image. PSNR, the number of edges in the image, SSIM, MSE, and UQI are used to evaluate image distortion in the image. The proposed method is evaluated using various embedding distortion evaluation methods.

PSNR is the peak signal-to-noise ratio. It is the ratio between the peak signal, which means the maximum power of the test image, and the noise, maximum noise. Noise is the image distortion that affects the image's representation quality. It is calculated as shown in Eq. 1.

$$PSNR = 10\log_{10}\left[\frac{255^2}{MSE}\right] \ (dB) \tag{1}$$

MSE is the mean squared error in the image, calculated by the mean of squared differences between input and steganographic images. The difference is calculated from pixel to pixel in an input image and stego image. The summation of all the pixel differences is then divided by the total number of pixels, that is, width x height of the images (input and stego). Eq. 2 represents how the mean squared error is calculated.

$$MSE = \frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H}(I_{ij} - SI_{ij})^2 \tag{2}$$

Where Iij and $SIij$ represent the pixel value at the location i and j in the input image and stego-image. At the same time, W and H are used to represent the width and height of both of the images.

SSIM is the structural similarity index. It is the quality measure of the image that represents the image degradation after performing the Steganography in the image. It is calculated by the lamination, contrast, and structure between the input image and the stego image. To calculate the SSIM, local mean, standard deviation, and cross-covariance are used between input and stego image. It is the perceptual difference between both images. Eq. 3 is used to calculate it.

$$SSIM(I, SI) = \frac{(2\mu_I\mu_{SI} + C_1)(2\sigma_{ISI} + C_2)}{(\mu_I^2 + \mu_{SI}^2 + C_1)(\sigma_I^2 + \sigma_{SI}^2 + C_2)} \tag{3}$$

Where I, SI,$\sigma I\sigma SI$, and σISI represent the local mean, standard deviation, and cross-covariance for both images.$C1$

(a) a (b) b (c) c (d) d (e) e (f) f (g) g (h) h

Fig. 4. Cover images (a) Baboon (b) Boat (c) Couple (d) House (e) Lena (f) Pepper (g) Sailboat (h) Tank.

and $C2$ are the regularization constants. UQI is the universal image quality. It is the ratio between the multiplicative variance and summation of variances of the input image and stego-image.

In order to perform the experimentation, the benchmark dataset of USC-SIPI is used. The experimentations are conducted on 8 images from the dataset, as shown in Fig. 4. The dimension of 512×512 is utilized, as already discussed in the methodology. The images for this experimentation purpose are (a) Baboon, (b) Boat, (c) Couple, (d) House, (e) Lena, (f) Pepper, (g) Sailboat, and (h) Tank.

Distortion evaluation is performed for two cases: Sobel filter and Canny edge detection, after completing the distortion

TABLE II. IMAGE EVALUATION RESULTS ON MESSAGE LOAD OF 24000 BITS

Filter	Image	PSNR	No. of Edges	Embedding Rate	SSIM	MSE	UQI
Canny	Tank	65.08	51602	0.091553	0.99992	0.0202	1
	baboon	64.24	70387	0.091553	0.99996	0.0245	1
	boat	64.84	58471	0.091553	0.99987	0.0213	1
	couple	64.93	56701	0.091553	0.99987	0.0209	0.99
	house	64.71	59923	0.091553	0.99991	0.0219	1
	lena	65.32	47839	0.091553	0.99983	0.0191	0.99
	pepper	65.20	50535	0.091553	0.99983	0.0196	0.99
	sailboat	64.86	55720	0.091553	0.99991	0.0212	1
Sobel	Tank	64.65	60875	0.091553	0.99990	0.0223	1
	baboon	63.30	94586	0.091553	0.99995	0.0304	1
	boat	63.58	86512	0.091553	0.99980	0.0285	1
	couple	63.50	88790	0.091553	0.99980	0.0291	1
	house	63.61	86757	0.091553	0.99987	0.0284	1
	lena	63.98	77158	0.091553	0.99974	0.0259	0.99
	pepper	63.50	87713	0.091553	0.99974	0.0290	1
	sailboat	63.58	85944	0.091553	0.99987	0.0285	1

TABLE III. IMAGE EVALUATION RESULTS ON MESSAGE LOAD OF 32000 BITS

Filter	Image	PSNR	No. of Edges	Embedding Rate	SSIM	MSE	UQI
Canny	Tank	64.58	51590	0.12207	0.99992	0.0227	1
	baboon	63.85	70395	0.12207	0.99995	0.0268	1
	boat	64.37	58445	0.12207	0.99986	0.0238	1
	couple	64.43	56687	0.12207	0.99986	0.0235	1
	house	64.24	59926	0.12207	0.99989	0.0245	1
	lena	64.75	47833	0.12207	0.99981	0.0218	0.98
	pepper	64.66	50504	0.12207	0.99981	0.0223	0.99
	sailboat	64.44	55703	0.12207	0.99988	0.0234	1
Sobel	Tank	64.18	60866	0.12207	0.99989	0.0248	1
	baboon	62.98	94583	0.12207	0.99994	0.0327	1
	boat	63.20	86497	0.12207	0.99979	0.0312	1
	couple	63.17	88743	0.12207	0.99979	0.0313	1
	house	63.24	86786	0.12207	0.99986	0.0309	1
	lena	63.56	77105	0.12207	0.99972	0.0286	0.98
	pepper	63.15	87698	0.12207	0.99971	0.0315	1
	sailboat	63.25	85963	0.12207	0.99986	0.0307	1

TABLE IV. IMAGE EVALUATION RESULTS ON MESSAGE LOAD OF 40000 BITS

Filter	Image	PSNR	No. of Edges	Embedding Rate	SSIM	MSE	UQI
Canny	Tank	64.14	51605	0.152588	0.99991	0.0251	1
	baboon	63.42	70392	0.152588	0.99993	0.0296	1
	boat	63.94	58445	0.152588	0.99985	0.0263	1
	couple	64.01	56649	0.152588	0.99984	0.0259	0.99
	house	63.83	59984	0.152588	0.99986	0.0270	1
	lena	64.23	47783	0.152588	0.99979	0.0246	0.97
	pepper	64.22	50509	0.152588	0.99979	0.0246	0.99
	sailboat	63.94	55711	0.152588	0.99985	0.0262	1
Sobel	Tank	63.71	60875	0.152588	0.99988	0.0277	1
	baboon	62.67	94591	0.152588	0.99993	0.0351	1
	boat	62.91	86535	0.152588	0.99978	0.0333	1
	couple	62.87	88774	0.152588	0.99977	0.0336	1
	house	62.88	86763	0.152588	0.99984	0.0335	1
	lena	63.21	77063	0.152588	0.99970	0.0311	0.97
	pepper	62.85	87631	0.152588	0.99970	0.0338	1
	sailboat	62.92	85927	0.152588	0.99983	0.0332	1

TABLE V. IMAGE EVALUATION RESULTS ON MESSAGE LOAD OF 48000 BITS

Filter	Image	PSNR	No. of Edges	Embedding Rate	SSIM	MSE	UQI
Canny	Tank	63.74	51593	0.183105	0.99990	0.0275	1
	baboon	63.09	70343	0.183105	0.99992	0.0319	1
	boat	63.56	58483	0.183105	0.99983	0.0287	1
	couple	63.64	56645	0.183105	0.99982	0.0281	0.98
	house	63.45	59928	0.183105	0.99984	0.0294	1
	lena	63.82	47830	0.183105	0.99977	0.0269	0.97
	pepper	63.77	50491	0.183105	0.99977	0.0273	0.98
	sailboat	63.54	55660	0.183105	0.99983	0.0288	1
Sobel	Tank	63.33	60878	0.183105	0.99987	0.0302	1
	baboon	62.35	94581	0.183105	0.99991	0.0378	1
	boat	62.59	86505	0.183105	0.99977	0.0359	1
	couple	62.56	88777	0.183105	0.99976	0.0361	1
	house	62.54	86734	0.183105	0.99982	0.0362	1
	lena	62.88	77108	0.183105	0.99968	0.0335	0.96
	pepper	62.54	87630	0.183105	0.99968	0.0362	1
	sailboat	62.57	85883	0.183105	0.99980	0.0360	1

(a) a (b) b

(c) c

Fig. 5. (a) Cover edge image, (b) stego edge image, and (c) difference between cover and stego edge images.

evaluation methods on message loads of 24000, 32000, 4000 and 48000 bits, respectively. The obtained results for tl message load of 24000 bits are shown in Table II. Tables III, IV, and V represent the results obtained on message load of 32000, 40000, and 48000 bits, respectively, for both cases. As shown in Tables II, III, IV, and V, both edge detection methods performed well. The result shows that when the message load is increased, the image quality is degraded. The image steganography can be applied on smaller message loads for better securing the data. As much as the message load increases, then the image quality degrades. Hence, it would be prone to be detected as a stenographic image.

The statistics provided show the proportion of even and odd pixels for every pairing of pictures. The cumulative numbers are quite comparable both before and after. Consider that the discrepancy between the sum of the variations and the number of manuscripts in the secret message is less. Fig. 5(a) shows the edge pixels of the cover image, which is identified by applying Sobel edge detection. Edge pixels of the stego image obtained after embedding a message are shown in Fig. 5(b), and Fig. 5(c) shows the difference between the two edge images. This indicates that the edge pixels in the cover and stego images are similar.

The image quality improves as more data is in the stego image and with the increase in the number of edges. The peak noise signal (PSNR) method for implementing the steganography procedure is the primary determinant of optimizing efficiency. The stego image will be more similar to the actual image if PSNR quality increases. The hiding capacity is increased with the pixel in stenography.

These findings show that the pictures look entirely unmodified to the unaided eye and are statistically the same. It is challenging for a person looking at both images or a computer looking at just one image to notice the possibility of increasing statistical similarity by introducing noise. Before and after changes, the ratio of even and odd dots is approximately the same. PSNR and MSE are used to assess the quality of stego images compared to cover images. Based on experimental results, the proposed method achieves a high-quality image by using the XOR operation to reduce the difference between the cover and stego images. PSNR values vary between 65 dB and 62 dB with embedding rates of 9% - 18%, where 35 dB is the minimum acceptable value.

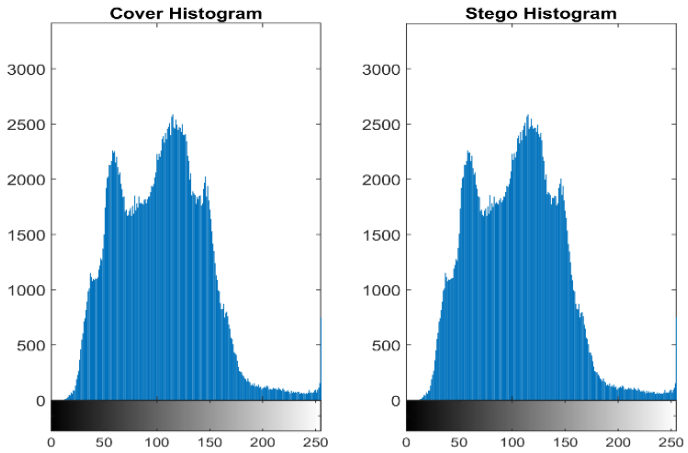

Fig. 6. (a) The cover image histogram and (b) The histogram of the corresponding stego image using the proposed algorithm.

The visual differences between the cover and stego images cannot be discovered by the human eye, and even the histograms of the stego images (illustrated in Fig. 6(a) and 6(b) are pretty similar. The algorithm's effectiveness was demonstrated by employing LSB for steganographic techniques and evaluating the histogram, PSNR, safety, and resemblance between human and machine readings.

V. Conclusion

The human eye is capable of detecting a significant variation in an image pixel. This means that the edges of an image might allow greater distortion than the other sections of the cover image since the edge portions have a sharper shift in pixel values than those that surround them. As a result, in edge-based Steganography, the majority of message bits are embedded in the edge pixels, and as one moves from the edge regions into the homogeneous areas of the cover picture, the number of bits embedded reduces, making the distortion less visible to the human eye. In this paper, an XOR-based embedding method was proposed to embed the data into the image. Sobel and Canny edge detection methods were used to extract edges. Sobel edge detection method is the traditional edge detection method used a few years ago [14], [20]. On the identified edges, an XOR operation is performed, which is a disjunction property [19]. The embedding capacity of an image is enhanced by edge detection methodology. In the presented method, a good rate of PSNR has been achieved.

Using an edge detection approach along with multiple-bit modification methods leads to high security. The contribution of this paper is embedding the message efficiently by incorporating XOR coding and identifying identical edges in the cover and stego images using the traditional edge detection methods. The technique was tested on various image distortion methods. The proposed approach performed efficiently, as shown in the results. The intensity of the edges in the input and steganographic images are estimated to be identical. The MSE is almost zero on a message load of 24000, 32000, 40000, and 48000 bits. Universal image quality and SSIM are closer to one representing the image quality, and the similarity between cover and stego images is almost the same. The approach may be expanded to several picture formats, including grayscale images, and requires no further information other than the stego image.

In terms of benefits, the suggested solution is undetectable since it only employs three LSBs to hide the secret data in the pixels of the detected edges. Furthermore, the recommended approach scatters bits of the secret data over specific regions of the identified edges rather than over all pixels of the carrier picture. A second benefit is that the buried data may be recovered. Different outputs for the same input image and secret data can be generated by hiding the secret data using a specific pattern denoted by the Canny algorithm, as well as parameterizing the algorithm to allow communicating parties to alter the effects and results of the algorithm.

References

[1] N. Ibraheem and M. Hasan, "Combining several substitution cipher algorithms using circular queue data structure," *Baghdad Science Journal*, vol. 17, no. 4, pp. 1320–1320, 2020.

[2] 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, 2019.

[3] O. Rachael, S. Misra, R. Ahuja, A. Adewumi, F. Ayeni, and R. Mmaskeliunas, "Image steganography and steganalysis based on least significant bit (lsb)," in *Proceedings of ICETIT 2019: Emerging Trends in Information Technology*. Springer, 2020, pp. 1100–1111.

[4] S. K. Ghosal, A. Chatterjee, and R. Sarkar, "Image steganography based on kirsch edge detection," *Multimedia Systems*, vol. 27, no. 1, pp. 73–87, 2021.

[5] G. C. Kessler and C. Hosmer, "An overview of steganography," *Advances in Computers*, vol. 83, pp. 51–107, 2011.

[6] H. Al-Dmour, N. Ali, and A. Al-Ani, "An efficient hybrid steganography method based on edge adaptive and tree based parity check," in *MultiMedia Modeling: 21st International Conference, MMM 2015, Sydney, NSW, Australia, January 5-7, 2015, Proceedings, Part I 21*. Springer, 2015, pp. 1–12.

[7] D. R. I. M. Setiadi, "Psnr vs ssim: imperceptibility quality assessment for image steganography," *Multimedia Tools and Applications*, vol. 80, no. 6, pp. 8423–8444, 2021.

[8] M. M. Emam, A. A. Aly, and F. A. Omara, "An improved image steganography method based on lsb technique with random pixel selection," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 3, 2016.

[9] H. Al-Dmour and A. Al-Ani, "A medical image steganography method based on integer wavelet transform and overlapping edge detection," in *Neural Information Processing: 22nd International Conference, ICONIP 2015, November 9-12, 2015, Proceedings, Part IV 22*. Springer, 2015, pp. 436–444.

[10] H. Khamis, "Studies on image steganography," Master's thesis, Itä-Suomen yliopisto, 2021.

[11] S. Gupta and N. K. Garg, "Optimized data hiding for the image steganography using hvs characteristics," in *Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences: PCCDS 2020*. Springer, 2021, pp. 275–285.

[12] Y. P. Astuti, E. H. Rachmawanto, C. A. Sari *et al.*, "Simple and secure image steganography using lsb and triple xor operation on msb," in *2018 International Conference on Information and Communications Technology (ICOIACT)*. IEEE, 2018, pp. 191–195.

[13] P. P. Balgurgi and S. K. Jagtap, "Intelligent processing: An approach of audio steganography," in *2012 International Conference on Communication, Information & Computing Technology (ICCICT)*. IEEE, 2012, pp. 1–6.

[14] P.-Y. Chen, W.-E. Wu *et al.*, "A modified side match scheme for image steganography," *International Journal of Applied Science and Engineering*, vol. 7, no. 1, pp. 53–60, 2009.

[15] S. L. Chikouche and N. Chikouche, "An improved approach for lsb-based image steganography using aes algorithm," in *2017 5th International Conference on Electrical Engineering-Boumerdes (ICEE-B)*. IEEE, 2017, pp. 1–6.

[16] E. J. Kusuma, C. A. Sari, E. H. Rachmawanto, and D. R. I. Moses Setiadi, "A combination of inverted lsb, rsa, and arnold transformation to get secure and imperceptible image steganography." *Journal Of ICT Research & Applications*, vol. 12, no. 2, 2018.

[17] Y.-H. Yu, C.-C. Chang, and Y.-C. Hu, "Hiding secret data in images via predictive coding," *Pattern recognition*, vol. 38, no. 5, pp. 691–705, 2005.

[18] C.-H. Yang, C.-Y. Weng, S.-J. Wang, and H.-M. Sun, "Adaptive data hiding in edge areas of images with spatial lsb domain systems," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 488–497, 2008.

[19] K. Gaurav and U. Ghanekar, "Image steganography based on canny edge detection, dilation operator and hybrid coding," *Journal of Information Security and Applications*, vol. 41, pp. 41–51, 2018.

[20] H.-W. Tseng and H.-S. Leng, "High-payload block-based data hiding scheme using hybrid edge detector with minimal distortion," *IET Image Processing*, vol. 8, no. 11, pp. 647–654, 2014.