ISSN: 0067-2904

Color Image Steganography Using Gradient Selective Bezier Curves

Suhaila N. Mohammed
Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq

Received: 14/3/2022 Accepted: 2/10/2022 Published: 30/7/2023

Abstract

Internet technology has revolutionized the landscape of communication technologies in the modern era. However, because the internet is open to the public, communication security cannot be guaranteed. As a result, data concealment approaches have been developed to ensure confidential information sharing. Various methods have emerged to achieve the goal of secure data communication via multimedia documents. This study proposes a method, which is both adaptable and imperceptible, for concealing a secret text in a color image. From an adaptivity perspective, image corners are detected using the Harris corner detection algorithm and utilized as anchor points for picking the optimal hiding regions of interest using Bezier curve interpolation. On the other hand, because human vision is less sensitive to aberrations in edge regions, imperceptibility is guaranteed by utilizing curves that cross through these regions. Experiments indicate that utilizing gradient selective Bezier curves for secret text concealment can keep the imperceptibility even when the payload capacity is increased.

Keywords: Image steganography, Harris corner detection, Bezier curve, Least Significant Bits (LSBs), Sobel gradient filter.

الإخفاء داخل الصور الملونة باستعمال منحنيات بيزير الانتقائية المتدرجة

سهيلة نجم محمد

قسم علوم الحاسوب, كلية العلوم, جامعة بغداد, بغداد, العراق

الخلاصة

أحدثت شبكة الإنترنت ثورة في تقنيات الاتصالات في العصر الحديث. ومع ذلك، نظرًا لأن الإنترنت متاح للجميع، فلا يمكن ضمان أمن البيانات. نتيجة لذلك، تم تطوير العديد من طرق إخفاء البيانات لضمان سرية تبادل المعلومات. وايضاً تم اقتراح طرق مختلفة لتحقيق هدف اتصال البيانات الآمن عبر استعمال بيانات الوسائط المتعددة. في هذه الدراسة تم اقتراح طريقة تكون قابلة للتكيف وغير محسوسة من العين البشرية لغرض اخفاء نص سري داخل الصورة الملونة. من منظور التكيف، تم استعمال زوايا الصورة التي تم تحديد مواقعها باستعمال خوارزمية Harris كنقاط ربط لاختيار مناطق الاخفاء المثلى باستعمال منحنى بيزير. من ناحية أخرى، نظرًا لأن الرؤيا البشرية أقل تحسس للتغايرات اللونية التي تحدث في مناطق الحواف، تم استغلال ذلك من خلال اختيار المنحنيات التي تمر عبر هذه المناطق لغرض الاخفاء. اشارت التجارب إلى أن استعمال منحنيات بيزير الانتقائية المتدرجة لإخفاء النص السري قادر على المحافظة على عدم إدراك العين البشرية لتغايرات الصورة حتى عند زيادة سعة الحمولة.

*Email: suhailan.mo@sc.uobaghdad.edu.iq

1. Introduction

Steganography is a data hiding technique that involves dissimulating secret information into digital files such that it can be effectively extracted by the appropriate receiver [1,2]. Checksum embedding, copyright control, secure transmission of company secret data, improving the strength of image-based search engines, TV broadcasting, and many others are examples of steganography applications [3].

Steganography systems use a variety of multimedia components as cover media, including images, video, audio files, and so on. Images are mostly used since consumers commonly employ images in internet-based communications such as e-mail and e-learning platforms [3, 4].

There are various types of image steganography techniques, each with their own unique mechanism for embedding the data. These approaches are divided into two categories: spatial domain and transform domain approaches. The spatial domain approaches, such as the Least Significant Bits (LSB) method, utilize the original pixels of the cover image to conceal the secret data [5]. The method involves embedding secret data directly into image pixels by changing the least significant bit value. It has a high payload and is computationally straightforward but vulnerable to statistical attacks [6]. In the transform domain methods, the cover image is first transformed into another representation (i.e., from the spatial domain to the frequency domain) prior to the embedding process [7]. For example, before incorporating the secret information into the image, a Discrete Cosine Transformation (DCT) can be applied to the pixels of the cover image [3]. Transform-based methods are more resistant to eavesdropper attacks [5].

Regardless of the application domain, all image steganography techniques must concentrate on three key points [8]:
1) Adaptivity. This point focuses on adaptive steganography, where the hiding places are different depending on the image content. Adaptiveness not only improves embedding efficiency, but it can also prevent stego-analysis with proper and effective countermeasures.
2) Imperceptibility. It means that the change in the image pixels' values should not be detected by the human vision system. The embedding of secret data must occur in specific regions that give the least distortion (Regions of Interest (ROIs)) rather than the entire image to achieve a safer embedding.

3) Payload capacity. Payload capacity is the maximum size of the secret data that can be embedded in the cover image. The higher the payload capacity, the better the method efficiency is. However, the amount of data pinned within the cover image will have an impact on the change level of the cover image pixels. Thus, any embedding procedure must maintain a higher payload capacity without affecting imperceptibility.

Image steganography methods that embed secret data into particular ROIs within an image are relatively new approaches. Edge regions are one of the ideal image ROIs for data concealment because the human visual system is less sensitive to distortions in edge regions, and it also provides randomized pixel placements [8,9].

In order to achieve adaptivity and increase payload capacity while maintaining even increased imperceptibility, this research proposes an image-based steganography method with the following contributions:

1) To ensure adaptivity, corner points of the image are utilized as anchor points for locating the ROIs. ROIs are formed as curves connect these corners.
2) To ensure imperceptibility, a fitness value is associated with each curve based on the gradient values of its points. Text hiding is then performed on the highest fitness curves. The hiding in gradient selective curves can lead to a less perceptible change in image colors.
3) In order to increase payload capacity while maintaining imperceptibility, the LSB method is utilized on the pixels, forming curves with the highest gradient fitness.

The rest of the paper is organized as follows: Section 2 presents a review of the related works, followed by a description of the concepts used in the proposed method in Section 3. The proposed methodology is given in Section 4. Section 5 presents the experimental results. Finally, the conclusion is provided in Section 6.

2. Related Works

Various works in color image steganography have been proposed. However, due to their proximity to the presented work, only those works that use edge detection, corners, and interpolation techniques will be highlighted in this section.

-*Edges-based methods*: Bassil [10] used the Canny edge detection algorithm to find locations of edges in the cover image and then hide bits of the secret data by replacing the three LSBs of every edge pixel. Islam et al. [11] proposed a technique for steganography in grayscale images. Data is hidden at the edges, which are dynamically selected based on the length of the data. Smitha and Baburaj [12] proposed Edge Adaptive based on the Least-Significant-Bit Matched Revisited (EALSBMR) approach with the help of Sobel edge detection. The EALSBMR approach gives a peak signal to noise ratio (PSNR) value equal to 43.6168 using the color version of the Lena test image. Sheelavathy et al. [13] used Canny edge detection and Hamming code algorithms for secret data embedding with the help of the XOR technique. Kumar [5] presented an adaptive steganography method based on a novel fuzzy edge identification for estimating the precise edge areas of the cover image. Setiadi [14] proposed a dilated hybrid edge detection using the three Most Significant Bits (MSBs) pixels of the cover image while the hiding is performed in the LSB of the pixels. The PSNR value for the greyscale version of the Lena test image was 69.5470.

-*Corners-based methods*: Mangayarkarasi and Sujatha [15] proposed the Harris method to measure image boundaries and corner points of the cover image. They achieved a PSNR value of 17.1627 using the parrot test image. Talib [16] also used corner points to hide secret data instead of edges. Additionally, they proposed the Clearing Least Significant Bit (CLSB) method to retrieve data from the stego-image without sending pixels to the pixel map to increase the security of the proposed corner-based hiding method.

-*Interpolation-based methods*: Karim et al. [17] suggested a system for steganography in multi-image based on the Bezier curve. A Bezier curve equation was used to select secret image pixel locations in the cover images. They achieved PSNR and Mean Square Error (MSE) values of 39.12 and 7.96, respectively, using the color version of the Lena standard test image. Benhfid et al. [1] proposed a reversible steganographic method based on pixel interpolation. They embedded the secret data in the error between the cover and interpolated pixels using an optimal pixel adjustment procedure and message adaptive error. The PSNR value was 40.2854 using the color version of the Lena test image. Hassan and Gutub [18] investigated the parabolic interpolation (PI) method to scale-up the original image, then embed the secret data using a

quadratic Bezier interpolation technique. They achieved a PSNR value of 32.10 using the pepper test image.

Table 1 summarizes related works, as well as the methodology used and the findings. Although the methods in Table 1 conceal the secret data with edges, corners, and curves, each of these techniques is considered separately, so the optimal hiding ROIs are not precisely identified. Combining these effective methods can boost adaptivity and preserve imperceptibility in a more efficient manner. This is the point at which the suggested work is motivated.

Table 1: The summary of related works

Authors	Used Methods	Test Material	Achieved Results
Bassil [10]	Canny edge detection algorithm and LSBs	24-bit BMP image selected by the author	N/A
Smitha and Baburaj [12]	EALSBMR approach	Lena test image	PSNR = 43.6168
Sheelavathy et al. [13]	Canny edge detection and hamming code algorithms	Lena test image	N/A
Kumar [5]	Fuzzy edge identification for estimating the precise edge areas of the cover image	Images taken from BOWS2 database	The values of PSNR are between 54.11 dB and 48.61 dB
Setiadi [14]	Dilated hybrid edge detection using the three MSBs pixels of the cover image	Lena test image	PSNR=69.5470
Mangayarkarasi and Sujatha [15]	Harris method to measure image boundaries and locations of corner points of the cover image	Parrot test image	PSNR= 17.1627
Talib [16]	Corner points and CLSB	N/A	N/A
Karim et al. [17]	Bezier curve equation to select locations of the secret image pixels in the cover images	Lena test image	PSNR=39.12 MSE= 7.96
Benhfid et al. [1]	Reversible steganography method based on pixels interpolation	Lena test image.	PSNR =40.2854
Hassan and Gutub [18]	PI and quadratic Bezier interpolation technique	Peppers test image	PSNR= 32.10

3. Theoretical Background

The basic concepts of the techniques used in the proposed method are presented here.

3.1 Edge Detection

An edge can be characterized as a set of joined pixels that form a separation between two disjoint areas. Edge detection is fundamentally a method of dividing an image into segments by locating points with unexpected changes in image intensities. Sobel kernels are one of the most effective methods for the detection of edges in both horizontal (180°) and vertical (90°) axes [19]. In Sobel, the x and y derivatives (I_x and I_y) of the image (I) are computed using gradient masks (G_x and G_y) as in the following equations [20]:

$$I_x = G_x * I \quad (1)$$

$$I_y = G_y * I \quad (2)$$

Where , * represents convolution operation and G_x and G_y are defined as follows:

$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$, and $G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$

Finally, the gradient magnitude can be determined using equation (3) [20].

$$|G| = \sqrt{I_x^2 + I_y^2} \tag{3}$$

3.2 Corner Detection

The corner is a point that represents an intersection of two image edges where the intensity changes in two different directions. Harris and Stephens [21] used the differential of pixels' intensities with respect to the direction to find the corner score. A moving window is opened in different directions and the pixel that shows high intensity changes is recorded as a corner. Harris corners can be determined using the following steps:

1) First, the results of derivative production (I_x^2, I_y^2) of every pixel in the image (I) are computed using equations (4) and (5).

$$I_x^2 = I_x * I_x \tag{4}$$

$$I_y^2 = I_y * I_y \tag{5}$$

Where, I_x and I_y are the x and y derivatives of the image (I) that are computed using equations (1) and (2).

2) Then, the response of the detector (R) at each pixel I(x,y) is determined using equation (6).

$$R = \det(M) - k\big(trace(M)\big)^2 \tag{6}$$

Where,

$$M = \sum_{x,y} I(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \tag{7}$$

and det(M) is the determinant of the matrix that can be generated using equation (8).

$$\det(\mathrm{M}) = I_x^2 . I_y^2 - I_x I_y . I_x I_y \tag{8}$$

trace(M) is the sum of diagonal elements of matrix M. It can be computed using equation (9).

$$\mathrm{trace(M)} = \mathrm{I_x^2} + \mathrm{I_y^2} \tag{9}$$

Where k is a tunable parameter within the range [0.04 - 0.06].

3) Finally, given a predetermined threshold value (T), the final decision regarding the pixel status (corner or non-corner) as is taken follows:

$$\mathrm{Decision(I(x,y))} = \begin{cases} \mathrm{Corner} & \text{if } \mathrm{R} > \mathrm{T} \\ \mathrm{Non-corner} & \text{otherwise} \end{cases} \tag{10}$$

3.3 Interpolation Curve

An interpolation curve is the curve that can be generated by interpolation or approximation of a set of control points where the constructed curve passes through these points. Bezier is one of the popular polynomials and is a powerful tool for curve interpolation. This is due to its stability and simple computation. In addition, the curve constructed with Bezier always lies within the convex hull, so it never deviates far from the control points [17].

Let P_i= (x_i, y_i), i = 0, 1, 2, ..., n, represent the Bezier Curve (BC) control points, the BC of degree 'n' can be defined as [22]:

$$BC(t) = \sum_{i=0}^{n} B_i^n(t) P_i \qquad 0 \le t \le 1 \qquad (11)$$

Where, $B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$, $\binom{n}{i} = \frac{n!}{i!\ (n-i)!}$, i = 0, 1, 2, ..., n are Bernstein polynomials of degree n.

In general, BC can be either a Quadratic Bezier Curve (QBC) or a Cubic Bezier Curve (CBC) according to the number of control points. QBC has three control points (P_i = (x_i, y_i), i = 0, 1, 2). There is only one control point in the middle of QBC and two end control points. For this reason, the control points make the QBC restricted by a line. QBC(t) can be defined as [22]:

$$QBC(t) = (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2 \qquad 0 \le t \le 1 \qquad (12)$$

Where, P_0 = (x_0, y_0), P_1 = (x_1, y_1), P_2 = (x_2, y_2) are the three control points of the curve.

The CBC has four control points (P_i = (x_i, y_i), i = 0, 1, 2, 3). There are two control points in the middle of the CBC and two end control points. For this reason, the control points make the CBC have an S-shape or C-shape in addition to the straight line. CBC can be defined as [22]:

$$CBC(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t)P_2 + t^3 P_3 \qquad 0 \le t \le 1 \qquad (13)$$

Where P_0 = (x_0, y_0), P_1 = (x_1, y_1), P_2 = (x_2, y_2), P_3 = (x_3, y_3) are the four control points.

4. Methodogy

4.1 Test Images

The standard Lena and pepper color images as shown in Figure 1 are used as test materials for the purpose of evaluation of the proposed method. Both images are with 24 bits resolution and dimensions equal to 256×256 for the Lena image and 512×512 for the pepper image [1].

(a)

(b)

Figure 1: Test material, (a) Lena image, (b) Peppers image

4.2 The Methodology

As illustrated in Figure 2, the proposed method has two primary paths: the embedding path and the extraction path. In the embedding path, the secret text is concealed within the cover image to generate the stego-image. While in the extraction path, the secret text is concealed outside of the stego-image. However, several stages (drawn in red in Figure 2) are shared by both paths. As a result, the common stages will be explained first, followed by the embedding and extraction procedures.

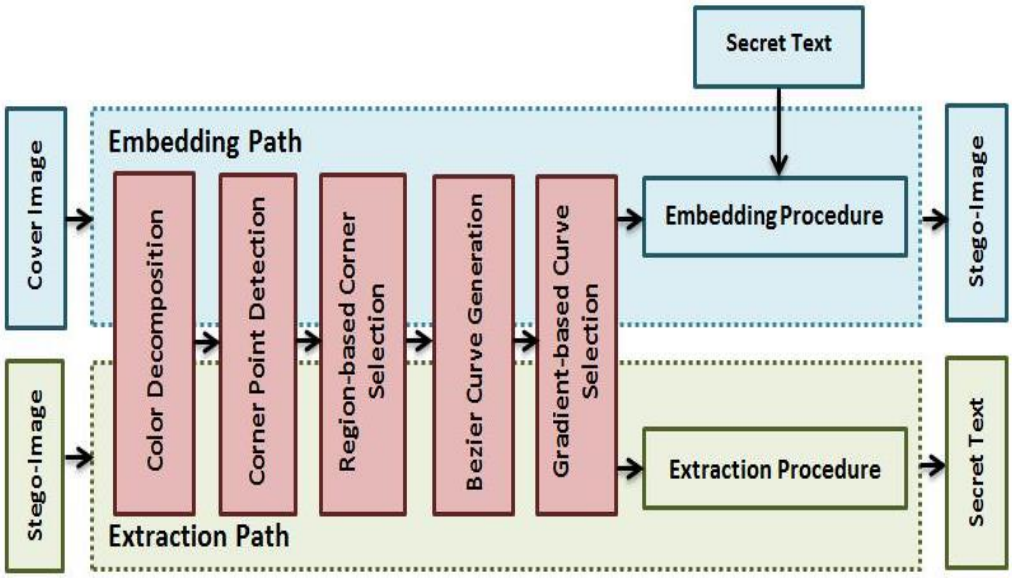


Figure 2: General design of the proposed method

4.2.1 Color Band Decomposition

The color image, which will be either the cover image or the stego-image depending on the path followed, is decomposed into its fundamental Red, Green, and Blue (RGB) color bands at this stage. For text embedding and extraction, the red and blue channels are utilized for hiding tasks. The green channel will remain unchanged since it will be supplied into the corner detection and gradient map generation stages, which are sensitive to color intensity changes. As a result, when the green channel is used to detect edge and corner points, identical curves are generated in both the embedding and extraction paths.

4.2.2 Corner Points Detection

To ensure adaptivity in determining the embedding regions, control points of the BC are located according to image details. To achieve this goal, the Harris corner detection algorithm is applied to the green channel for finding corner points, which will be the basis for locating the best ROIs for text hiding. The corner points differ from one image to another, and this makes the proposed steganography method an adaptive one. Figure 3(a) gives examples of applying the Harris corner detection algorithm on two different standard test images where the corner points are marked in red.

4.2.3 Region-based corner selection

The corner detection stage will provide many corners, but only a small number of these points are required for curve construction. Thus, region-based corner points are selected in this stage. The selected points must be scattered along the image to ensure imperceptibility after the text is hidden. The following two steps are applied to select such corner points:

1-Arrange the corner points in ascending order with respect to the Y-axis.

2-After that, the image is divided into N regions, and only the first corner point is selected from each region. N is a variable value that can be found by experimentation. This ensures that the generated curves will be scattered along the image and passed through pixels in the close positions. The results of applying this stage are shown in Figure 3(b), where the selected corners are marked with blue circles.

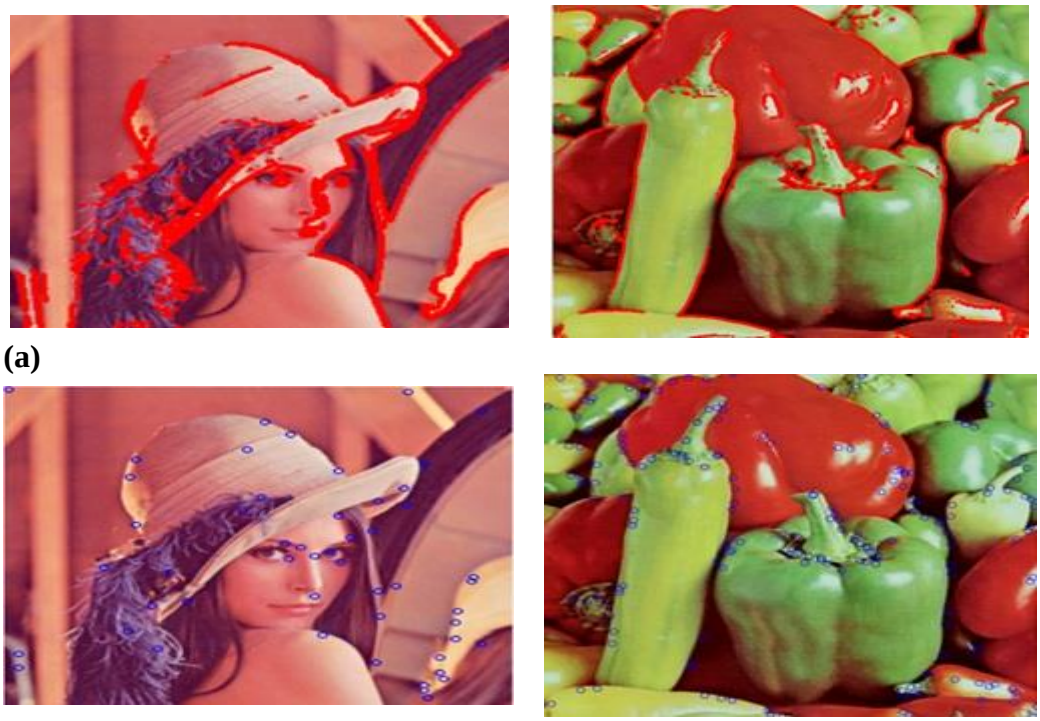


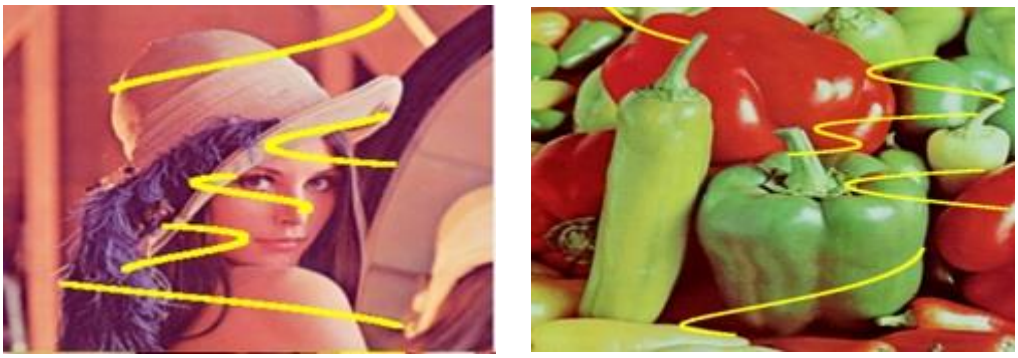
Figure 3: Results of corner detection and selection, (a) Corner detection results, (b) Corner selection results

4.2.4 Bezier Curve Generation

At this stage, the Bezier curves are used to generate the ROI where the secret text is to be hidden. Both QBC and CBC curves are attempted in this work. In QBC, every three successive points are fed to the QBC to generate the curve by approximating the locations of the pixels between these points. The total number of the generated QBC curves from N corner points is equal to $(N/3)$. While in CBC, the four successive control points are considered as control points of the CBC to generate the curve. The total number of the generated CBCs is $(N/4)$. An example of the generated curves for both QBC and CBC is shown in Figure 4.

However, the generated curves may contain pixels with coordinates outside of the image boundaries. In addition, many intersection points may be present among the curves. If such curves are used for text hiding, many of the text characters will be overwritten in the same pixel locations. To avoid such a case, in each generated curve, the duplicated points with the already generated curves are eliminated.

(a)



(b)

Figure 4: Curves generated using selected corner points, (a) using QBC, (b) using CBC

4.2.5 Gradient-based Curve Selection

The text embedding and extraction procedures are performed starting from the curves with the highest gradient degree to the smallest ones. Thus, a fitness value will be associated with each curve based on the gradient degree of its points. To generate the gradient map of the image, Sobel edge detection is applied to the green channel of the image. After generating the gradient map, the fitness of each curve (C_i) of length (L) is computed using equation (14). Finally, the curves are then sorted in descending order with respect to their fitness values.

$$Fitness(C_i) = \frac{\sum_{j=0}^{L-1} Gradients\,(C_i(j).x, C_i\,(j).y)}{L} \quad (14)$$

Where, $C_i(j).x$ and $C_i(j).y$ are the x and y coordinates of the point (j) in the C_i , respectively and $Gradient$ is the gradient map of the image.

4.2.6 Secret Text Embedding

After locating the ROIs for embedding purposes, the text hiding process is started from the first point in the curve with the highest fitness. One character is embedded in each pixel. The LSB method is utilized to embed bits of the American Standard Code for Information Interchange (ASCII) code of the character in the red and blue channels of the pixel. Steps followed to hide the character are:

1) The first four LSBs of the red and blue bands of the pixel (j) in the curve $C_i$are first cleared using equations (15) and (16).

$$R_1(j) = Red(C_i(\mathrm{j}).\mathrm{x}, C_i(\mathrm{j}).\mathrm{y})\ \&\ 240 \quad (15)$$

$$B_1(j) = Blue(C_i(\mathrm{j}).\mathrm{x}, C_i(\mathrm{j}).\mathrm{y})\ \&\ 24 \quad (16)$$

2) After that, the first four MSBs of the character (M(j))are extracted using equation (17), while the first four LSBs are extracted from the character using equation (18).

$$M_1 = (M(j)\ \&\ 240) >> 4 \quad (17)$$

$$M_2 = M\ (j)\ \&\ 15 \quad (18)$$

3) The M_1 part of the character is then embedded in R_1 using equation (19), while the M_2 part of the character is embedded in B_1 using equation (20).

$$Red(C_i(\mathrm{j}).\mathrm{x}, C_i(\mathrm{j}).\mathrm{y}) = R_1 \mid M_1 \quad (19)$$

$$Blue(C_i(\mathrm{j}).\mathrm{x}, C_i(\mathrm{j}).\mathrm{y}) = B_1 \mid M_2 \quad (20)$$

4) Finally, the total number of message characters (NoChs) is embedded in the red and blue channels of the pixel located at the first row of the last column (Image height (H)-1) as in the following equations:

$$\mathrm{Red}(0, \mathrm{H}-1) = \mathrm{NoChs} \,\&\, 255 \quad (21)$$

$$\mathrm{Blue}(0, \mathrm{H}-1) = (\mathrm{NoChs} >> 8) \,\&\, 255 \quad (22)$$

4.2.7 Secret Text Extraction

In the extraction procedure, the hidden text is concealed from the stego-image using the same curves generated during the embedding process. The extraction procedure involves the following steps:

1) The total number of characters is extracted using the following equations:

$$P_1 = Red(0, H-1) \quad (23)$$

$$P_2 = Blue(0, H-1) \quad (24)$$

$$NoChar = P_1 | (P_2 \ll 8) \quad (25)$$

2) The first part of the message (M_1) is retrained from the first LSBs of the red band using equation (26).

$$M_1 = Red\ (C_i(\mathrm{j}).\mathrm{x}, C_i(\mathrm{j}).\mathrm{y}) \,\&\, 15 \quad (26)$$

2) The second part of the message (M_2) is also retrieved from the first LSBs of the blue band using equation (27).

$$M_2 = Blue(C_i(\mathrm{j}).\mathrm{x}, C_i(\mathrm{j}).\mathrm{y}) \,\&\, 15 \quad (27)$$

4) Using equation (28), the two parts are then combined to form the ASCII code of the character (M(j)).

$$\mathrm{M(j)} = (M_1 \ll 4) + M_2 \quad (28)$$

5. Results and Discussion

5.1 Evaluation Metrics

Imperceptibility quality can be measured using PSNR. It measures the distortion in the stego-image. In fact, the PSNR computes the deviation between the stego-image and the cover image [23]. Typical values for the PSNR in lossy image and video compression are between 30 and 60 dB, where the higher values are better. It is expressed in terms of the logarithmic decibel (dB) as [24]:

$$PSNR = 10\ log_{10}\left(\frac{MAX^2}{MSE}\right) \quad (29)$$

Where, MAX is the maximum intensity value in the image, MSE is the mean square error that represents the differences between the original cover image (I) and the stego-image($\bar{I}$). The MSE can be computed using equation (30) [25].

$$MSE = \frac{1}{WH}\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}(I(x,y) - \bar{I}(x,y))^2 \quad (30)$$

Where, W and H are the width and height of the image (I), respectively.

5.2 Experimental Results

Three alternative payload sizes have been used to test the imperceptibility quality of the suggested method with varying payload capacities: 1K, 2K, and 3K. The PSNR and MSE values are measured using QBC with three interpolation step sizes (0.005, 0.01, and 0.03) and several regions (N) equal to 100, 150, and 200. The observed findings for this experiment with the given parameter configuration settings are shown in Tables 2 to 4.

Table 2: Results achieved when using QBC with a payload capacity=1K

No. of Regions (N)	Interpolation Step (t)	Lena image		Peppers image	
		PSNR	**MSE**	**PSNR**	**MSE**
100	0.005	47.8564	1.0652	51.0667	0.5086
	0.01	47.7642	1.0880	51.1056	0.5040
	0.03	47.7788	1.0844	51.1276	0.5015
150	0.005	47.8450	1.0680	51.1529	0.4986
	0.01	47.8244	1.0730	51.0966	0.5051
	0.03	**47.9574**	1.0549	51.0337	0.5125
200	0.005	47.8000	1.0789	51.1216	0.5022
	0.01	47.6895	1.1069	51.1250	0.5018
	0.03	47.8077	1.0772	**51.4691**	0.4067

Table 3: Results achieved when using QBC with a payload capacity=2K

No. of Regions (N)	Interpolation Step (t)	Lena image		Peppers image	
		PSNR	**MSE**	**PSNR**	**MSE**
	0.005	47.0858	1.2720	50.1669	0.6257
100	0.01	47.0274	1.2893	50.1893	0.6225
	0.03	47.0265	1.2888	50.1788	0.6404
	0.005	47.0812	1.2733	**50.3564**	0.6029
150	0.01	47.0648	1.2781	50.1565	0.6272
	0.03	**47.3075**	1.2156	50.1413	0.6294
	0.005	47.0462	1.2836	50.2533	0.6133
200	0.01	46.9937	1.2933	50.1746	0.6246
	0.03	47.0154	1.2928	50.1778	0.6241

Table 4: Results achieved when using QBC with a payload capacity=3K

No. of Regions (N)	Interpolation Step (t)	Lena image		Peppers image	
		PSNR	**MSE**	**PSNR**	**MSE**
100	0.005	45.5436	1.8143	48.5185	0.9145
	0.01	45.5233	1.8553	48.5544	0.9070
	0.03	45.5014	1.8756	48.5721	0.9010
150	0.005	45.5763	1.8007	48.6013	0.8973
	0.01	45.5705	1.8031	48.5306	0.9120
	0.03	45.5700	1.8051	**49.4627**	0.7358
200	0.005	**45.5876**	1.7960	48.6429	0.8887
	0.01	45.5320	1.8191	48.6040	0.8967
	0.03	45.5201	1.8312	48.9756	0.8232

Curve interpolation is conducted in the second experiment using CBC. The same configuration settings that were used in the first experiment are adopted here. Tables 5 to 7 show the PSNR and MSE values achieved for this experiment.

Table 5: Results achieved when using CBC with a payload capacity=1K

No. of Regions (N)	Interpolation Step (t)	Lena image		Peppers image	
		PSNR	**MSE**	**PSNR**	**MSE**
100	0.005	47.8602	1.0642	51.0524	0.5103
	0.01	47.9037	1.0536	50.9906	0.5176
	0.03	47.7657	1.0876	51.1577	0.4980
150	0.005	47.8588	1.0646	51.1208	0.5023
	0.01	47.8745	1.0607	**51.2321**	0.4896
	0.03	47.8710	1.0616	51.1134	0.5031
200	0.005	**47.9513**	1.0654	50.9726	0.5197
	0.01	47.7643	1.0880	51.1034	0.5043
	0.03	47.7741	1.0855	51.1902	0.4943

Table 6: Results achieved when using CBC with a payload capacity=2K

No. of Regions (N)	Interpolation Step (t)	Lena image		Peppers image	
		PSNR	**MSE**	**PSNR**	**MSE**
100	0.005	47.0012	1.2970	50.2319	0.6164
	0.01	47.0121	1.2938	50.1168	0.6329
	0.03	47.0045	1.3059	50.1033	0.6191
150	0.005	46.9368	1.3164	50.2557	0.6130
	0.01	46.9759	1.3046	50.2864	0.6087
	0.03	46.9359	1.3146	50.2030	0.6205
200	0.005	**47.2407**	1.1853	50.0949	0.6361
	0.01	47.0041	1.2961	50.1771	0.6242
	0.03	47.0203	1.2913	**50.4931**	0.5678

Table 7: Results achieved when using CBC with a payload capacity=3K

No. of Regions (N)	Interpolation Step (t)	Lena image		Peppers image	
		PSNR	**MSE**	**PSNR**	**MSE**
100	0.005	45.5796	1.7993	48.6494	0.8874
	0.01	45.3796	1.8214	48.5652	0.9048
	0.03	45.1796	1.8314	48.6548	0.8954
150	0.005	45.5265	1.8214	48.6771	0.8817
	0.01	45.5741	1.8016	48.6009	0.8974
	0.03	45.4741	1.9016	48.9989	0.8999
200	0.005	**45.8673**	1.6844	**49.3695**	0.7556
	0.01	45.5748	1.8013	48.6449	0.8883
	0.03	45.4748	1.9013	48.5449	0.9883

Figure 5 depicts the visual results of the embedding procedure, with the original cover images on the left and the stego-images acquired after applying the embedding procedure to the test images on the right. As it is clearly shown in Figure 5, the imperceptibility is preserved where the visual effects of the hiding process cannot be detected by the human vision system.

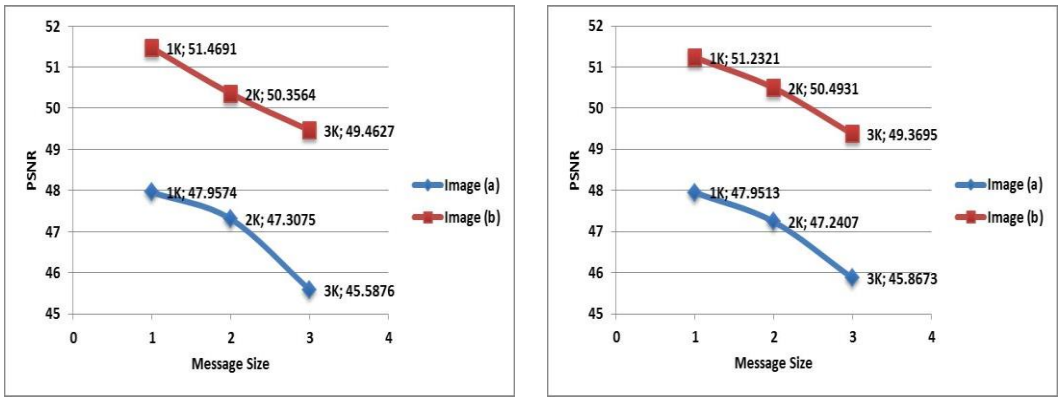
(a)

(b)

Figure 5: Visual results of the hiding procedure, (a) Lena image, (b) Peppers image

5.3 Results Discussion

As previously mentioned, each steganography approach must consider three primary factors: adaptivity, imperceptibility, and payload capacity. As demonstrated in Figure 4, adaptivity is achieved by choosing BC control points depending on image statistics (i.e., corner point locations). As a result, the ROIs will vary from one image to another, as shown in Figures 6 and 7. The second aspect, imperceptibility, is achieved by concentrating the embedding procedure on curves that predominantly pass across gradient pixels. Tables 2 to 7 indicate that the suggested concealment method can save image quality with a PSNR value of more than 45. The third aspect, payload capacity, as shown in Figure 6, has a small influence on the imperceptibility quality of the test images. As it is clearly shown in Figure 6, about 1 dB of degradation occurs in the PSNR value each time the payload size is increased.

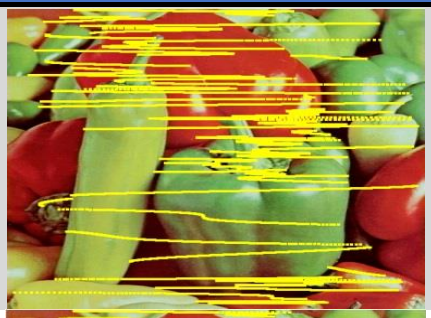
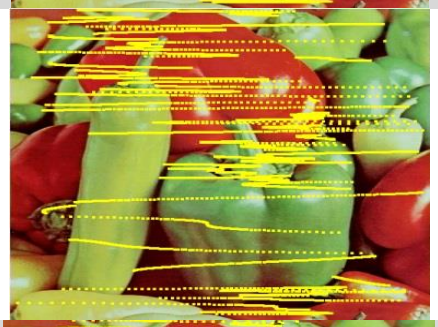
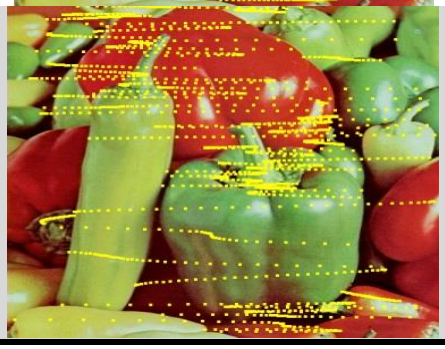


(a) **(b)**

Figure 6: Impact of payload size on the imperceptibility quality, (a) QBC, (b) CBC

The best PSNR values are obtained when the interpolation step is small (nearly t =0.005 or t = 0.01), as demonstrated in Tables 2 to 7. This can be explained by the fact that as the step size is raised, the curve generates fewer points, as shown in Table 8. As a result, additional curves will be required for text embedding, and curves with modest gradients will play a role in the concealment process.

Table 8: The influence of interpolation step size on the generated QBC

Interpolation Step	Lena image	Peppers image
0.005		
0.01		
0.03		

On the other hand, more short-length curves with close point coordinates will be formed as the number of regions (N) increases, and the opposite is also true, as shown in Figure 7. Thus, as it can be seen in Tables 2 to 7, the PSNR values are raised when N increases since curves which mostly pass through gradient regions will be generated.

(a) **(b)** **(c)**

Figure 7: The effect of the number of regions (N) on the generated QBC, (a) N=100, (b) N=150, (c) N=200

Table 9 compares the results of the proposed gradient selective Bezier curves steganography method to those of other studies that used the same test images to ensure fair comparison conditions. The table shows that the PSNR value that is achieved by the proposed method outperforms values achieved by other methods. However, the PSNR value reached by Setiadi [14] is better than that of the proposed method. This is due to the fact that the author used the PSNR equation, which is normalized by taking the square root of the MSE, and this has a significant impact on the outcome because the division will be based on a lower MSE value (kindly see [14]).

Table 9: Comparison between the proposed method and related works

Authors	Used Methods	Test Material	PSNR
Smitha and Baburaj [12]	EALSBMR approach	Lena test image	43.6168
Setiadi [14]	Dilated hybrid edge detection using the three MSBs pixels of the cover image	Lena test image	69.5470
Karim et al. [17]	Bezier curve equation to select locations of secret image pixels in the cover images	Lena test image	39.12
Benhfid et al. [1]	Reversible steganography method based on pixels interpolation	Lena test image.	40.2854
Hassan and Gutub [18]	PI and quadratic Bezier interpolation technique	Peppers test image	32.10
The proposed method	***Harris corner detection algorithm and gradients selective Bezier curves***	***Lena test image***	***47.9574***
		Peppers test image	***51.4691***

6. Conclusions and Future Works

Steganography techniques have recently been used to provide safe transmission of secret data over the internet. From this perspective, a color image steganography method was proposed in this paper. An adaptive localization of ROIs was utilized by using image corner points as anchor points for curve generation. The imperceptibility was achieved by applying the LSB approach to conceal the secret data into curves with the highest gradient fitness. The obtained PSNR values demonstrate the efficiency of the suggested method in terms of imperceptibility and payload capacity. However, the best ROIs were chosen using statistical approaches that

involved ranking the fitness values of the curves. Thus, metaheuristic approaches can be deployed in the future to determine the best ROIs for secret data hiding.

References

[1] A. Benhfid, E. Ameur, and Y. Taouil, "Reversible Steganographic Method Based on Interpolation by Bivariate Linear Box-Spline on the Three Directional Mesh," *Journal of King Saud University –Computer and Information Sciences*, vol. 32, pp. 850–859, 2020.

[2] K. Kordov, and S. Zhelezov, "Steganography in Color Images with Random Order of Pixel Selection and Encrypted Text Message Embedding," *PeerJ Comput. Sci*., vol. 7, 2021.

[3] A. M. Alhomoud, "Image Steganography in Spatial Domain: Current Status, Techniques, and Trends," *Intelligent Automation & Soft Computing*, vol.27, no.1, pp. 69-88, 2021.

[4] M. A. Majeed, R. Sulaiman, Z. Shukur, and M. K.Hasan, "A Review on Text Steganography Techniques," *Mathematics*, vol. 9, no. 21, 2021.

[5] S. Kumar, A. Singh, and M. Kumar, "Information Hiding with Adaptive Steganography Based on Novel Fuzzy Edge Identification," *Defence Technology*, vol. 15, pp. 162-169, 2019.

[6] Z. S. Younus, and M. K.Hussain, "Image Steganography Using Exploiting Modification Direction for Compressed Encrypted Data," Journal of King Saud University – Computer and Information Sciences, 2019, https://doi.org/10.1016/j.jksuci.2019.04.008

[7] A. J. Yousif, "Image Steganography Based on Wavelet Transform and Color Space Approach," *Diyala Journal of Engineering Sciences*, vol. 13, no. 3, pp. 23-34, 2020.

[8] R. Roya, A. Sarkara, and S. Changder, "Chaos based Edge Adaptive Image Steganography," *International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013, Procedia Technology*, vol. 10, pp. 138-146, 2013.

[9] M. Fateh, M. Rezvani, and Y. Irani, "A New Method of Coding for Steganography Based on LSB Matching Revisited," *Security and Communication Networks*, vol. 2021, pp. 1-15, 2021.

[10] Y. Bassil, "Image Steganography based on a Parameterized Canny Edge Detection Algorithm," *International Journal of Computer Applications*, vol.60, no.4, pp. 35-40, 2012.

[11] S. Islam, M. R. Modi, and P. Gupta, "Edge-Based Image Steganography," *EURASIP Journal on Information Security*, vol. 2014, no. 8, pp. 1-14, 2014.

[12] G. L. Smitha, and E. Baburaj, "Sobel Edge Detection Technique Implementation for Image Steganography Analysis," *Biomedical Research*, Special Issue, pp. 487-493, 2018.

[13] S. Sheelavathy, R. Hamsavani, J. Disha, C. Bhavana, and R. Bhoomika, "Image Steganography Technique based on Canny Edge Detection and Hamming Code for Medical Data," *International Journal of Engineering and Advanced Technology* (IJEAT), vol. 8, no. 5S, pp. 23-25, 2019.

[14] D. R. Setiadi, "Improved Payload Capacity in LSB Image Steganography Uses Dilated Hybrid Edge Detection," *Journal of King Saud University – Computer and Information Sciences*, vol. 34, no. 2, pp. 104 -114, 2022.

[15] S. Mangayarkarasi, and P. Sujatha, "Steganographic Method for Digital Images Using Harris Method," *International Journal of Pure and Applied Mathematics*, vol.114, no. 12, pp. 267-276, 2017.

[16] [16] A. Talib, "Corners-based Image Information Hiding Method," *Iraqi Journal for Computers and Informatics* (IJCI), vol. 43, no. 1, pp. 1-5, 2017.

[17] A. A Karim, A. J. Hussein, and H. M. Alwan, "Image Steganography System Using Bezier Curve," *Al-Mansour Journal*, vol. 31, pp. 111- 133, 2019.

[18] F. S. Hassan, and A. Gutub, "Novel Embedding Secrecy within Images Utilizing an Improved Interpolation-Based Reversible Data Hiding Scheme," *Journal of King Saud University – Computer and Information Sciences*, 2020, https://doi.org/10.1016/j.jksuci.2020.07.008.

[19] H. A. J. Albayati, and S. A. Ali, "A Comparative Study of Image Steganography Based on Edge Detection," *Iraqi Academics Syndicate International Conference for Pure and Applied Sciences, Journal of Physics: Conference Series*, vol. 1818, pp. 1-16, 2021.

[20] P. Ram, and S. Padmavathi, "Analysis of Harris Corner Detection for Color Images," *International conference on Signal Processing, Communication, Power and Embedded System* (SCOPES), 2016.

[21] C. Harris, M. and Stephens, "A Combined Corner and Edge Detector," *AVC*, pp. 147-152, 1988, doi:10.5244/C.2.23

[22] M. Abbas, and E. Jamal, and J. Ali, "Bezier Curve Interpolation Constrained by a Line," *Applied Mathematical Sciences*, vol. 5, no. 37, pp. 1817- 1832, 2011.

[23] S. A. Naji, H. N. Mohaisen, Q S. Alsaffar, and H. A. Jalab, "Automatic region selection method to enhance image-based steganography," *Periodicals of Engineering and Natural Sciences*, vol. 8, no. 1, pp. 67-78, 2020.

[24] S. N. Mohammed, S. Ahmed, Gh. Mohammed, and D. Abduljabbar, "Block-based Image Steganography for Text Hiding Using YUV Color Model and Secret Key Cryptography Methods," *Australian Journal of Basic and Applied Sciences*, vol. 11, no. 7, pp. 37-41, 2017.

[25] I. Kich, B. Ameur, and Y. Taouil, "Image Steganography by Modified Simple Linear Iterative Clustering," *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 4, pp. 1640-1647, 2020.