Multi-Stage Protection using Pixel Selection Technique for Enhancing Steganography

Khalil Ibrahim Mohammad Abuzanouneh[1], Mohammed Hadwan[1,2,3]

[1]Department of Information Technology, College of Computer, Qassim University Buraydah, Saudi Arabia
[2]Department of Computer Science, College of Applied Sciences, Taiz University, Taiz, Yemen
[3]Intelligent Analytics Group (IAG), College of Computer, Qassim University, Buraydah, Saudi Arabia

***Abstract*:** Steganography and data security are extremely important for all organizations. This research introduces a novel stenographic method called multi-stage protection using the pixel selection technique (MPPST). MPPST is developed based on the features of the pixel and analysis technique to extract the pixel's characteristics and distribution of cover-image. A pixel selection technique is proposed for hiding secret messages using the feature selection method. The secret file is distributed and embedded randomly into the stego-image to make the process of the steganalysis complicated. The attackers not only need to deter which pixel values have been selected to carry the secret file, they also must rearrange the correct sequence of pixels. MPPST generates a complex key that indicates where the encrypted elements of the binary sequence of a secret file are. The analysis stage undergoes four stages, which are the calculation of the peak signal-to-noise ratio, mean squared error, histogram analysis, and relative entropy. These four stages are used to demonstrate the characteristics of the cover image. To evaluate the proposed method, MPPST is compared to the standard technique of Least Significant Bit (LSB) and other algorithms from the literature. The experimental results show that MPPST outperforms other algorithms for all instances and achieves a significant security enhancement.

***Keywords*:** Digital Image Processing Techniques, Cryptography, Data Embedding, Steganography, Pixel Selections, Dynamic Threshold.

1. Introduction

Digital Image Processing Techniques (DIPTs) are considered as a branch of artificial intelligence (AI) that is concerned with the enhancement and analysis of images. Steganography is the art and science of hiding secret messages where only the sender and recipient know about them. Generally, Steganography uses media such as images, sound, video, etc. to hide secret messages. In the area of Steganography and security, DIPTs is among the significant interpretation and visualization methods [1].

DIPTs has tools for analyzing, detecting, transmitting, storing images. Network security is required for data protection, which uses the development-integrated systems during each stage from design to implementation. The successful testing of security systems is based on the analyzing process through different stages (information collection stage, data testing, and security evaluation). DIPTs are applied for analyzing and evaluating image pixels, the image colors are the main method to hide data based on the extracted information. Many methods in image processing are employed for analyzing image pixels [2]. This is to have robust and effective systems for data security. There are several complications to extract data from images related to the shape, edge, position, and size. Imbalanced illumination causes varied image contrast between the background and boundaries, which depend on the conditions of the capturing process of an image to get more hiding secrecy for the data. This research focuses on the stenography and encryption-decryption process from images to build an effective security system based on the work in [3]. The DIPTs can be applied in pixel segmentation based on a set of constraints to find features and calculate the security ratio. DIPTs aims at classifying and identifying various types of pixels' cell security, for example, edges, border detection, feature extraction, pixel selections, region growing, filtering processes, and mathematical morphology. Researchers in [4], analyzed and developed LSB based method against sample pair analysis steganalysis detection. Therefore, this research aims to enhance the security mechanisms of the LSB technique by improving the process of hiding secret files to make it difficult for intruders to recover them. It also aims to increase the size of hidden files without affecting the quality of stego-image.

The RGB color system produced all other colors by combining the main colors, which are Red, Green, and Blue. Each color has 8 bits with integer values ranging from 0 - 255 which can make 256 * 256 * 256 =16777216 RGB possible colors [5]. Each pixel in the RGB monitor system is shown by joining these three colors (RGB). When the red pixel is set to 0, it means it turned off, while 255 means turned fully on any value between them and adjusts LED using partial light emission [5]. RGB has a code that includes a 24 bits' format ranging (0-23 bits) RGB = (R*65536) + (G*256) + B. In computer graphics, the image refers to a collection of numbers that represent the light intensities at different regions.

For image colors, each pixel value has a bit of depth that represents the range of colors. The most widely used is 1 byte (8bits) and 3 bytes (24 bits) for color images. The slight variation in the image is very significant to carry and hide a secret message by using Human Vision Systems. This carrier type can give us the flexibility to conceal a secret message based on the slight changes in the pixel values. The selected pixels are utilized by the steganography technique to exploit the pixel's bit(s) to embed the secret message [5]. Whenever the higher the bit depth, the more data will be hidden. The availability of different types of image formats and techniques for image modification has made the steganography mechanisms development particular to the cover-image type. In [6], the robustness of LSB steganography is improved based on the random pixel selection. The image region can be selected, and then random values are generated to locate pixels after analyzing the required region. Additionally, the random values are appended to the pixels as password encryption. Authors in [7] have proposed a new technique to increase the LSB algorithm security and maintain the minimum effects on the quality of the cover image. Their technique modifies the LSB of the selected pixels by applying a bit-inversion to reduce the hiding impact on the pixel values. Their results showed enhancement of stego-image quality and get a better Peak Signal-to-Noise (PSN) value.

In this paper, a novel algorithm is proposed using MPPST with LSB focusing on misleading and camouflaging to enhance the steganography process. MPPST embeds secret data in the LSB plane of the cover-image using a random complex key with an encryption-decryption method via a disordered system to make a dynamic distribution in the stego-image. The compensation pattern is changing based on the analysis of features extraction and pixels selection of an image.

2. Literature Review

The literature review related to the proposed algorithm is presented in this section. The researchers proposed to analyze the image file based on DISTs. This helps in increasing the data security level, achieving accuracy, and reducing the efforts as well. The MPPST is an automated analysis system that has been developed to discover the image characteristics as a prior step in information hiding. The region's pixels contain a wide range of different colors that need analysis tools to detect the image features and the diversity of the image colors. The classifications of image pixels will be formulated as patterns to be recognized for the input process. The checking of patterns and colors reflect the features of the pixels of the original image that are extracted for merging the secret file into it.

In [8], the authors proposed a stenographic technique to improve the security of LSB. Their technique defines four types of LSB array. The type of LSB array can be selected based on the appropriate message size. Therefore, the first type of LSB0 is used for smaller messages, while the LSB3 is used for a longer message. The secret message data are represented based on the selected array (LSB0, LSB1, LSB2, LSB3) to get maximum matching. The starting for embedding message and the length of secret message words in the cover-images are encrypted using the RSA algorithm. Therefore, a weakness of their pro using RSA encryption can be considered as a weakness because it needs to create long keys to get high security which makes the execution of encryption very slow. Another weakness is appending the indications of starting for embedding message and message size, when using and applying the LSB array, this procedure can serve the intruder to explore the secret message [9]. The authors O. Zanganeh and S. Ibrahim proposed a substitution technique of image steganography as a new approach, the algorithm enhanced the embedding capacity of the image without losing the stego-image imperceptibility [10]. In [11], the authors proposed adaptive real-time reversible data hiding based on coefficients of discrete cosine transformation blocks (DCTBs) to enhance data hiding capacity and image quality. S. Y. Shen and L. H. Huang in [12] proposed a new method to build a cover image that is first mapped into a 1D pixels sequence using Hilbert filling curve to be divided into non-overlapping embedding units with two consecutive pixels. The method uses differences in the pixel value to evaluate the digit base to be embedded into pixel pairs and get good embedding capacity to enhance the embedding rate [12]. The authors in [13] presented the spatial LSB domain technique based on sharp edges of the image colors to hide secret messages in sharper edge regions. After that, in the image smooth regions, the message embedding depends on the image content and the message size. The method hides the message in sharp edges of stego-image to get better visualization quality of stego-image and low embedding rate. In the high embedding rate, edges and smooth regions of the cover image are used for increasing the capacity of the data hiding [13]. The authors in [14] proposed the complementary embedding method to avoid several statistical attacks. In this method, many statistical attacks are implemented to detect the stego-images using the proposed method; the steganographic technics have good performance in capacity and security for secret communication.

3. Proposed algorithm

In this research, the MPPST is applied to analyze and describe the chromatic features for the texture of the image colors. This is considered as highly discriminating features to perform hiding secret data inside media outside suspicions. MPPST uses convergence process iterations to detect image pixels, and color intensity aiming at improving the search capacity of the image environment for information hiding. The pixel values are modeled as random variables of the complex key, and then the probability of distribution of color density is calculated using the Entropy algorithm. An optimum threshold based on the threshold amplitude is a basic part that can be used in the techniques of image segmentation. Pixels threshold T is selected to split up two region modes in the image. The first mode of image points I (x, y) > = T, is represented in object points, on the other hand, where I (x, y) < T the points are called background points [15]. The threshold equation is defined as in Equation 1:

$$g\,(x,y) = \begin{cases} 0, & I(x,y) \; < \; T \\ 1, & I\,(x,y) \; \geq \; T \end{cases} \qquad \textbf{(1)}$$

The threshold is considered as global when Tg sets on the whole image I (x, y) is in Equation 2:

$$T_g\,(y,x) = [\,I(x,y)\,] \quad \forall T_g \in I(x,y) \qquad \textbf{(2)}$$

Whereas the local threshold is considered when Tl (x, y) applies into spatial coordinates of the image, but in the image that has dynamic threshold Td depends on Il(x, y) and specifying of property p (x, y) of the local threshold. The level average of pixels in a neighborhood centered on point coordinates Il (x, y). The weight function of the dynamic threshold is given by Equation (3).

$$T_d\,(y,x) = [\,p\,(x,y)\,,\, I_l(x,y)],\, \forall T_d \in I(x,y)\,, p\,(x,y) \qquad \textbf{(3)}$$

The author in [16] introduced a technique, which uses the content addressable method for RGB images. In this method, the color image will be divided into a set of clusters and indexed using the content addressable method. The clustering algorithm is checking all pixels of cover image pixels and sorting each index in its cluster. In [17], the authors presented a secured method using image formats steganography technique for BMP or JPEG images, based on clustering technique, the cover image divided into 8x8 blocks then used the content addressable method. The clustering technique of the color image is used to protect the embedded steganography from attacks. Bawaneh et al. in [18] proposed a secure technique called flash video file steganography that saves the quality of video frames. Their proposed technique is able to hide different types of secret message inside the cover image.

In this section, the proposed analysis and security system for image segmentation are explained in detail. The phases of MPPST are presented in figure1. In the first stage, MPPST applies image pre-processing to remove noise, improve contrast variation, and luminance in the cover-images for data hiding. In the second stage, MPPST explores and isolates the interesting objects in the image, using segmentation processes. The next stage is devoted to the extraction of the characteristics of the object to be used in the next phase. The procedure of image analysis is applied to the extraction of features and pixel selection. The utilization of the capability of the complex key is aimed to generate random locations and point creation to enhance the security of the LSB algorithm. The complex random key is used for the selection and controlled by subkeys parameters (k1, k2, k3, and k4). The coordinate of k1 and k2 of any point (row, column) cannot exceed the size of the cover image. The variables of image size can be adjusted to the required values, such as the vertical and horizontal pixel counts. The image size determines the pixel selection boundary on the cover image. The recipient must receive this complex key to extract the embedded secrets.

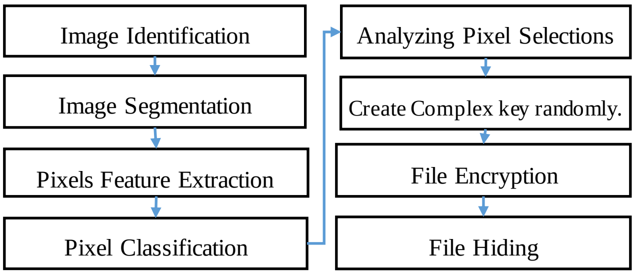


Figure 1: The Block Diagram of Image Analysis and Processing for Hiding Information

The selected features are used as a reference strategy for the classification method; MPPST takes the decision related to pixel selection. In the next stage of the proposed system, we developed a new algorithm to generate a complex secret key based on image analysis and segmentation techniques for hiding information. It contains multiple subkeys that are hidden randomly inside the image with the secret data. One complex key should be entered, in addition to the three characters used for the key decryption. The obtained results of using MPPST show that the human eyes cannot spot the differences between original images and stego-images. The stego-image can be sent through any available media to the receiver to extract and decrypt the hidden data in the stego-image, LSB's last bits of all bits into an image is replaced with secret file bits. When using three bytes (24-bits), the last bits of each color component of RGB are used, and 8 bits in each pixel can be used to hide data. Therefore, 800 × 600-pixel images can hide 1440000 bits or 180000 bytes of the embedded file.

To analyze the image, Hough transform technology is used for feature extraction to get the pixel numbers and color count. MPPST algorithm applied a formula to convert pixel numbers in the cover-image to actual count by using Hough Transform [19]. In the used images, the pixel numbers per cubic millimeter based on the color number are calculated as given by Equations (4). The steps of the MPPST algorithm for image Analysis and Data Security are introduced in Figure 2.

$$Accuracy = \left(\frac{pixelsCount}{actualCount} \right) * 100 \qquad \textbf{(4)}$$

3.1 Compute MSE, PSNR, and entropy

Information security uses steganography algorithms depending on the level of human imperceptibility of the original image changing. Therefore, a stenographic algorithm will generate enough innocent stego-images. The stego-image deformation degree to the original image plays an essential part. The image distortion is measured by PSNR.

3.1.1 Mean Squared Error (MSE)

MSE is computed between the cover image and the stego-image. The lower values of MSE indicated the minimum error noticed in the inverse relation between the MSE and PSNR [20]. To calculate the value of PSNR, the MSE must be calculated according to Equation 5.

$$MSE = \frac{1}{M.N} \sum_{x=0}^{M} \sum_{y=0}^{N} |I_1(x,y) - I_2(x,y)| \qquad \textbf{(5)}$$

Where M and N represent the dimensions of the image. The x and y represent the loop variables. Respectively I_1 is stego-image, I_2 is the cover image. The obtained results according to the experiments of MPPST, LSB, and chaotic steganography and encryption text in a discrete cosine transform domain (DCT) for image pixels, in [21] are presented in Tables 1, 2, and 3 respectively.

3.1.2 Peak Signal-To-Noise Ratio (PSNR)

PSNR is used to compute the difference between the cover-image and stego-image. While the ratio value is used as a quality measurement of the cover-image and a stego-image. PSNR will evaluate the quality of the newly constructed setgo-image. The PSNR is calculated as presented by Equation (6). 8-bits as an integer number (0- 255). Pseudocode of PSNR and MSE introduced in figure 3. In equation 7, R denotes the maximum pixel intensity between both images, where R = 255. The color range has.

$$PSNR = 20 \log_{10} \left(R \ / \ \sqrt{MSE} \right) \qquad \textbf{(6)}$$

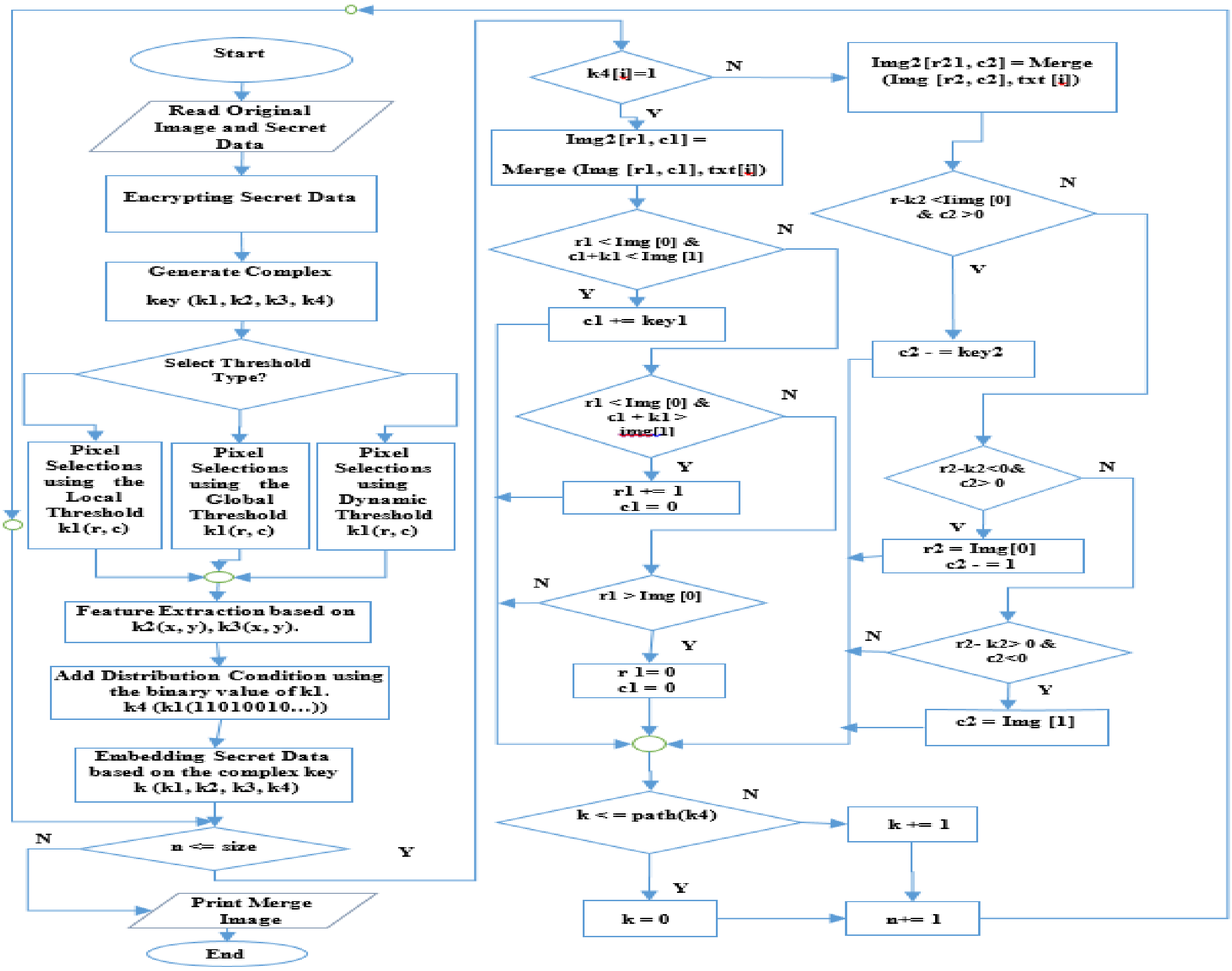

Start
Read Original Image and Secret Data
Encrypting Secret Data
Generate Complex key (k1, k2, k3, k4)
Select Threshold Type?
Pixel Selections using the Local Threshold k1(r, c)
Pixel Selections using the Global Threshold k1(r, c)
Pixel Selections using Dynamic Threshold k1(r, c)
Feature Extraction based on k2(x, y), k3(x, y).
Add Distribution Condition using the binary value of k1. k4 (k1(11010010…))
Embedding Secret Data based on the complex key k (k1, k2, k3, k4)
n <= size
N
Y
Print Merge Image
End
k4[i]=1
N
Y
Img2[r1, c1] = Merge (Img [r1, c1], txt[i])
Img2[r21, c2] = Merge (Img [r2, c2], txt [i])
r1 < Img [0] & c1+k1 < Img [1]
N
Y
c1 += key1
r-k2 < Img [0] & c2 > 0
N
Y
c2 - = key2
r1 < Img [0] & c1 + k1 > img[1]
N
Y
r1 += 1
c1 = 0
r2-k2<0& c2> 0
N
Y
r2 = Img[0]
c2 - = 1
r1 > Img [0]
N
Y
r 1= 0
c1 = 0
r2- k2> 0 & c2<0
N
Y
c2 = Img [1]
k <= path(k4)
N
Y
k += 1
k = 0
n+= 1

Figure 2: The Proposed algorithm MPPST for image analysis and embedding process

3.1.3 The entropy of text file

The entropy method is applied for the symbols sequence of data. It specifies a minimum bound after calculating the average number and bits required for the encoding process of the secret file in the proposed algorithm. In this research, the secret text is a set of symbol sequences; the entropy is calculated according to equation 8. It is included in the Numpy package in the Python programming language. Entropy text encoding is one of the lossless compression methods applied for text after the quantization process. Entropy represents the texts efficiently using less memory for the saving and transmission process. Entropy encoding is applied to increase the ratio of text compression to minimize distortion of stego-image as calculated by Equation (7).

$$Entropy \ = -\sum_{i=0}^{L-1} Pi \log_2 (Pi) \qquad (7)$$

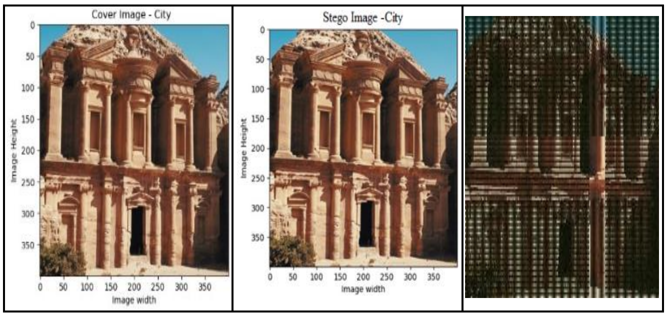
Where L is equal to the message length as of in characters. The pi is indicating the probability of correlated character in level i. Maximum entropy has calculated the sum of all products of character probability with the information content of character.

```
Calculate PSNR and MSE between cover-image and
stego-image using Python code.
import numpy; import math; import cv2
img1 = cv2.imread("pic1.png")
img2 = cv2.imread("pic2.png",1)
def psnr (pic1, pic2):
  mse = numpy.mean( (pic1 – pic2) ** 2 )
  return 20*math.log10( R /math.sqrt (MSE))
print (psnr (img1, img2))
```

Figure 3: The function of computing PSNR and MSE

4. Experimentation and results

The proposed algorithm is implemented and tested using the Python programming language with 75 different sizes of the cover-images. A laptop with Intel Core i5, 2.90 GHz processor, 8 GB RAM, and Windows 10 64-bit operating system was used to perform the experiments. In the testing stage, various images of different sizes and types were used for cover-images with various secret data. In the experiments, cover-images were classified according to the segmentation, extraction features, and pixel selections in an image. This is to reduce the complexity of image recognition. The image segmentation is a necessary process to divide the cover-image into several regions based on regional homogeneous features of the image colors based on different types of thresholds. In the first experiment, Petra's image is used to hide a size file (85 kB) that was inserted-encrypted and decrypted-extracted using a local threshold to be part of the complex key and the distribution of the file bits based on the LSB method. Figure 4 shows the cover-image, stego-image, and byte's distribution image for the Petra image. Table 1 presents the experiments carried out in this research using the proposed algorithm, the PSNR and MSE are used as a measure of the quality of the image reconstruction.



(a). Petra cover-image (b). Petra Stego-image (c) Byte's distribution

Figure 4: (a) Petra cover-images; (b) stego-images using Local threshold; (c) Distribution of hidden bytes

Figure 5 shows the distortion and their histogram for both the cover and stego-images of the Petra image.

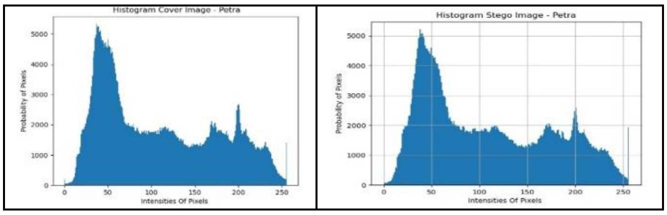


Figure 5: Histogram of Cover-image and Stego-image of Petra

Table 1: The experimental results of the image analysis of MPPST using different Images and sizes.

Image Name	**Image Size (bytes)**	**Secret file (bytes)**	**PSNR (dB)**	**MSE**	**Hidden Ratio (%)**
Lena	96,297	30,699	36.29	15.26	31.8
KSA	47,670	24,971	49.75	0.687	52.3
Hotel	126,322	72,092	40.87	5.31	57.0
Baboon	44,131	19,108	37.34	11.97	43.2
Baboon		30,699	36.29	15.26	69.5
Baboon		25,517	36.70	13.90	57.8
Petra	40,794	193.00	54.41	0.23	0. 473
Petra		24,971	40.43	5.88	61.2
Petra		30,699	40.27	6.11	75.2
City1	78,605	47,371	38.64	8.88	60.2
City1		30,699	40.07	6.39	39.0
City1		72,092	35.76	17.22	91.7
City2	511,996	409,504	39.22	7.765	79.9

The higher PSNR indicates the reconstruction of higher quality, we conclude that the PSNR values of Petra (54.41) and, KSA (49.75) images are the highest. The lower values of MSE indicated the minimum error noticed in the inverse relation between the MSE and PSNR. The MSE is calculated to compare image reconstruction quality. The MSE illustrates the cumulative squared error between the stego-image and the cover image. The lower value of the MSE means that the lower error between two images. The MSE of Petra (0.23) and KSA (0.687) have the lowest values when compared to the MSE results between cover-image and stego-images. Furthermore, the maximum value of the hidden ratio for City Image is (91.7) with good indicators of PSNR and MSE of the Proposed algorithm as presented. Besides, it can be deduced from the results above that the difference among average PSNR of JPEG images is 55 dB. The main reason is that the discrete cosine transform (DCT) coefficients of the cover-images that can slightly be changed. Table 2 shows a comparison of the experimental results using the PSNR of the proposed method and other methods from the literature.

Table 2: PSNR performance based on the comparative results of previous studies and the proposed algorithm.

Methods		PSNR of Images		
		Lena	Pepper	Baboon
Proposed Algorithm		58.32	58.33	58.67
RDHS	[11]	47.27	44.42	31.5
Chang		35.95	41.41	40.49
Comp embedding	[22]	34.67	34.75	37.64
Jseteg		35.36	35.45	38.82
Out Quesses		36.37	35.32	38.22
Adaptive PVD	[23]	50.89	51.29	52.29
Adaptive PVD	[24]	46.17	47.06	48.49

Our proposed algorithm outperforms all of the other techniques for all test cases. The best results based on the literature obtained from [23] Lena, Pepper, and Baboon images are (50.89, 51.29, 52.29) respectively, while our algorithm obtained (58.32, 58.33, 58.67) which are significantly better for all images. It is noticed that our proposed algorithm provides improved values to reach a 13.49 % improvement average compared with previous results for the best three images that are greater than 50. The experiment results are illustrated in table 3 where the images have different capacities and properties. Therefore, the embedding amount highly depends on the optimum threshold for the selected pixels, and the image features.

Table 3: Comparative results of the payload capacity & PSNR.

No	**Image Names**	**Hiding Capacity (bpp)**	**PSNR [21]**	**PSNR [25]**	**PSNR [26]**	**Proposed Method**
1	**Lena**	0.0958	54.53	46.73	49.68	**56.81**
2	**Elaine**	0.0673	56.20	45.37	53.60	**58.72**
3	**Pepper**	0.0807	55.18	41.12	51.88	**57.85**
4	**House**	0.1181	54.59	41.50	53.92	**56.68**
5	**Boat**	0.0658	56.18	45.69	53.82	**58.66**
6	**Airplane**	0.1378	53.47	43.25	47.83	**56.75**
7	**Trees**	0.0684	56.03	40.69	54.41	**58.65**

Based on the results in table 3, the proposed method outperforms all other methods from the literature for PSNR values. For instance, the PSNR value for Elaine's image is 58.72 while other methods values were 53.60, 56.20, and 45.37 obtained by [21], [25] and [26] respectively. In general, the threshold value is adjusted to be low and the capacity of the image should be high. However, it is noticed for those images that they have a high threshold; the image capacity is not enough to hide a large number of bits. As shown in table3, the results of the hiding capacity for the House image has 0.1601 bpp, and the Airplane image contains 0.1887 bpp. The number of bits and capacity of House and Airplane images are the largest when compared with the results of other images. Figure 6 presents an illustration of PSNR values using the cover-images (Elaine, Pepper, House, Boat, Airplane, Trees of size (512 × 512)).

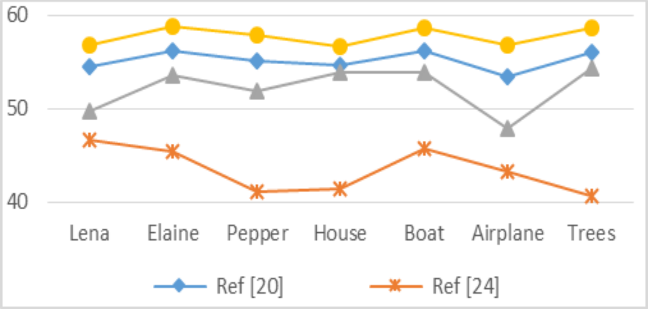


Figure 6: PSNR values using the cover-images (Elaine, Pepper, House, Boat, Airplane, Trees of size (512 × 512)).

Figure 6 shows the PSNR results taken from table3 for test images. The optimum threshold was adjusted in the Proposed Method to be (0.7), which means the payload capacity is increased, and the PSNR value is changed a little bit, which makes it an important part of controlling the capacity of the data hiding and distribution process. The optimum threshold is considered an essential part of the complex key.

5. Conclusion

To sum up, the main contribution of this research is that it introduces a new algorithm based on a complex and random secret key that contains multiple keys to reduce the detectability of secret files. Moreover, applying the MPPST in the steganography system is very significant to hide and retrieve secret data that are hidden inside the image files. Besides, it is helpful in the encrypting and decrypting secret message to increase the trust process of data transfer between the end-users. The secret file is hidden randomly with a complex key inside the stego-image. The obtained results for the relative entropy show that MPPST is sufficiently secure to hide secret messages and to decrease the size of stego-image with an unnoticeable change that cannot be detected easily. The values of embedding probability demonstrated the reliability and robustness of the security system of MPPST against the intruders and attackers.

6. Acknowledgement

The authors wish to thank the financial support of this work from the Deanship of Scientific Research, Qassim University according to the agreement of the funded project No. 5175-COC-2018-1-14-S, Qassim, Kingdom of Saudi Arabia.

References

[1] K. Abuzanounneh, "New Image Processing Techniques Using Elitism Immigrants Multiple Objective of Genetic Algorithms for Disease Detection", Int. J. Comput. Sci. Netw. Secur, vol. no.15, pp 252-260, December 2017.

[2] K. Abuzanouneh, "Hybrid Algorithm for Enhancing and Increasing Image Compression Based on Image Processing Techniques", Int. J. Adv. Comput. Res, vol. 16, pp. 90-98, Mar 2018.

[3] V. M. Ladwani and S. Murthy K, A new approach to securing images, Int. J. Comput. Sci. Netw, no. 1, vol. 4, 224–227, 2015.

[4] P.C. Wu, W.H. Tsai, "Detection of LSB Steganography via Sample Pair Analysis", Pattern. Recognit. Lett, vol. 24, 1613-1626, 2003.

[5] N. F. Johnson, and S. Jajodia, "Exploring steganography: Seeing the unseen. Computer, 31: DOI: 10.1109/MC, pp. 26-34 .1998.

[6] V.M. Viswanatham. and J. Manikonda, "A novel technique for embedding data in the spatial domain". Int. J. Comput. Sci. Eng., 2: pp. 233-236, 2010.

[7] N. Akhtar, S. Khan, and P. Johri," An improved inverted LSB image steganography", Proceedings of the International Conference on Issues and Challenges in Intelligent Computing Techniques, Feb. 7-8, IEEE Xplore Press, DOI: 10.1109/ICICICT, pp. 749-755, 2014.

[8] G. Swain, and S.K. Lenka, "A novel steganography technique by mapping words with LSB array", Int. J. Signal Imag. Syst. Eng, 8: pp. 115-122, 2015.

[9] C. Liuand S.R. Liao, "High-Performance JPEG steganography using complementary embedding strategy, Pattern.Recognit, vol 41, no. 9, pp. 2945–2955, 2008.

[10] O. Zanganeh and S. Ibrahim, "Adaptive image steganography based on optimal embedding and robust against chi-square attack", Information Technology Journal, no.7, vol.10, 1285–1294,2011.

[11] C. N. Yang, C. Kim, and Y.-H. Lo, ''Adaptive real-time reversible data hiding for JPEG images,'' J. Real-Time Image Process., vol. 14, no. 1, pp. 147–157, Jan. 2018.

[12] S. Y. Shen and L. H. Huang, ''A data hiding scheme using pixel value differencing and improving exploiting modification directions,'' Comput. Secur., vol. 48, pp. 131–141, Feb. 2015.

[13] A. J. Umbarkar, P. R. Kamble, and A. V. Thakre, ''Comparative study of edge-based LSB matching steganography for color images,'' ICTACT J. Image Video Process., vol.6, no.3, pp.1185–1191, Feb. 2016.

[14] C. L. Liu and S.-R. Liao, ''High-performance JPEG steganography using complementary embedding strategy,'' Pattern Recognit., vol. 41, no. 9, pp. 2945–2955, Sep. 2008.

[15] T. R. Singh, S. Roy, O.I. Singh, and K.M. Singh, "A new local adaptive thresholding technique in banalization", Int. J. Comput. Sci. Info, vol. 8, pp. 271-277, 2011.

[16] M. Othman, "New Image Watermarking Scheme based on Image Content Addressing Method", Proc. 13th WSEAS, Int. Conf. Appl. Comput. Sci, Kuala Lumpur, April 2014.

[17] M. Othman, A. Ansari, M. Mohammadi, "Digital color image steganography for nonspecific format and secured based on Clustering", Int. J. Comput. Sci. Netw. Secur, VOL.19, No 4, pp. 20-27. April 2019.

[18] M. Bawaneh, Obeidat A., Al-kofahi, M., "An adaptive FLV steganography approach using simulated annealing," International Journal of Communication Networks and Information Security (IJCNIS), Vol.10, No. 2, pp. 56-66. 2018.

[19] E. Achtert, C. Böhm, J. Davidand A. Zimek, "Global Correlation Clustering Based on the Hough Transform ", Statistical Analysis and Data Mining: The ASA Data Science Journal, Vol. 1, No. 3, pp.111–127, 2008.

[20] U Sara1, M Akter2, M Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study", JCC, vol.7, pp.8–18, 2019.

[21] P. Maneroo, Tohari Ahmad," Information hiding scheme for digital images using difference expansion and modulus function", Vol. 31, Issue 3, pp. 335-347. July 2019.

[22] C. L. Liu and S.-R. Liao, ''High-performance JPEG steganography using complementary embedding strategy,'' Pattern Recognit., vol. 41, no. 9, pp. 2945–2955, Sep. 2008.

[23] A. Pradhan, K. R. Sekhar, and G. Swain, ''Adaptive PVD steganography using horizontal, vertical, and diagonal edges in six-pixel blocks,'' Secur. Commun. Netw., vol. 2017, pp. 1–13, 2017.

[24] G. Swain, ''Adaptive pixel value differencing steganography using both vertical and horizontal edges,'' Multimedia Tools Appl., vol. 75, no. 21, pp. 13541–13556, 2016.

[25] T. Ahmad; M. Holil; W. Wibisono; I. Royyana,” An improved Quad and RDE-based medical data hiding method “, International Conference on Computational Intelligence and Cybernetics (CYBERNETICSCOM), pp. 141-145, 2013.

[26] A. M. Alattar. 2004. Reversible watermark using the difference expansion of a generalized integer transform. Trans. Img. Proc. 13, 8, pp.1147–1156, 2004.