Journal of Positive School Psychology http://journalppw.com
2022, Vol. 6, No. 5, 2566-2576

Text Hiding In Image Based On Robust Spatial Hashing Algorithm

Nadia Abud Al_karim[1] , Suhad A. Ali[2] , Majid Jabbar Jawad[3]

[1]Msc Student at Computer Science Dept., College of Science for Women, University of Babylon, Iraq.
2,3 Computer Science Dept., College of Science for Women, University of Babylon, Iraq.

Abstract

Due to its efficiency in resisting steganographic techniques, the concept of coverless information concealment has been considerably improved since it was first presented. In this paper; a new coverless steganography for text hiding is proposed in order to better hide secret data and improve attack resistance. This method depends on spatial domain. The embedding process consists of several steps. Firstly, the secret data is divided into no overlapping segments. Secondly, find appropriate images to be stego images and divided it to blocks. Thirdly, to build a hash sequence for an image, a powerful hashing algorithm is used. Fourthly, choose the image which its hash equivalent to the secret data segment. Fifthly, choose the image which its hash equivalent to the secret data segment. The suggested method's robustness is evaluated by a series of tests. The results of the studies show that the proposed technique is resistant to JPEG compression, noise, low pass filtering, scaling, rotation, median and mean filtering, brightness, and sharpening, among other image processing attacks.

Keywords— Coverless Steganography, Steganography, information security, information hiding, hashing generation

1. Introduction

In a traditional steganography technique, the secret information is hidden inside the carriers, which are used to hide the other information. The various media files that are commonly used in this process include audios, videos, and digital images [1]. The method that is commonly used in this process involves directly embedding the secret information into the carriers. Due to the partial distortion of the data carrier, a third party can detect the presence of hidden information. Among the various media files that are commonly used in this process are digital images. In a traditional image steganography technique, the values of the pixel are modified to achieve the hiding of the secret information. There are two main methods that are commonly used in this process: transform domain and spatial domain.

In the spatial domain, such as the method proposed in paper [2]which replace the LSB (least significant bits) of the image with secret data, the adaptive LSB hiding method[3], the spatial adaptive steganography algorithm S-UNIWARD[4], HUGO[5], WOW and so on; which is a type of steganography that uses the least significant bits of the image to hide the secret information. The transform domain method, on the other hand, involves modifying the data of the image to achieve the data hiding. Some of the statistical features that are used in this process include the DFT[6], DCT[7], and DWT[8]. Due to the nature of the techniques used in this process, it is inevitable for them to leave traces of their modification on the carriers. This makes them unsuitable for use in steganalysis tools Researchers have proposed a concept that aims to prevent the detection of different types of techniques used in this process[9].

2. Related Works

Briefly, this section reviews some of previously proposed methods related to coverless image

steganography. Instead of using a combination of secret information and a specific image, the researchers proposed by Zhou et al. [6] a coverless steganography method that doesn't involve other methods. This method can be prompted with a confidential message, which then produces a cipher vector for the secret message. Aside from the operation method, this new approach also doesn't generate help resources, which saves a Signal channel. After further studies, a variety of coverless image steganography techniques will be suggested. In 2018 , Xiang Zhang et.al The researchers proposed a coverless image steganography method that uses the LDA topic model and the discrete cosine transform. This method generates feature sequences based on the DC coefficients. After choosing an image, the feature sequences are transmitted to the receiving end[6]. Liming Zou et.al. proposed method generates a set of hash sequences by calculating the average pixel value of the sub image. This method then maps the dictionary and the array through a mapping relationship. It can be efficiently used to retrieve the hidden information from the image[10]. Xiang Zhang et. al. These researchers then used a method known as fractal image generation to hide the secret information. This method was able to achieve this by controlling the rendering of the image[11]. In 2020 Yi Cao et. al. proposed Coverless information hiding based on the generation of anime characters. In this method the researchers then generated anime characters using the hidden information by converting it into an attribute label set. This method can be used to generate various anime characters by using GANs[12]. QiLi et. al. proposed an encrypted coverless information hiding method based on generative models. The encryption and decryption phases of this process involve sending secret images between two image domains. In the encryption stage, the secret image is first embedded into a public image. After that, the image is input to a generative model, which then produces an encrypted image. An extraction module and an adversarial loss algorithm are then used to improve the quality of the images generated during the encryption stage. A second generative model is created in the decryption stage to recreate the synthetic images from the encrypted images. Lastly, the secret image is separated from the synthetic image that has been recreated[13].

3. The Proposed Method

Figure 1 shows the proposed coverless image steganography framework stages. The proposed method consists of two procedures:

- Embedding procedure (sender side)
- Extraction procedure (receiver side)

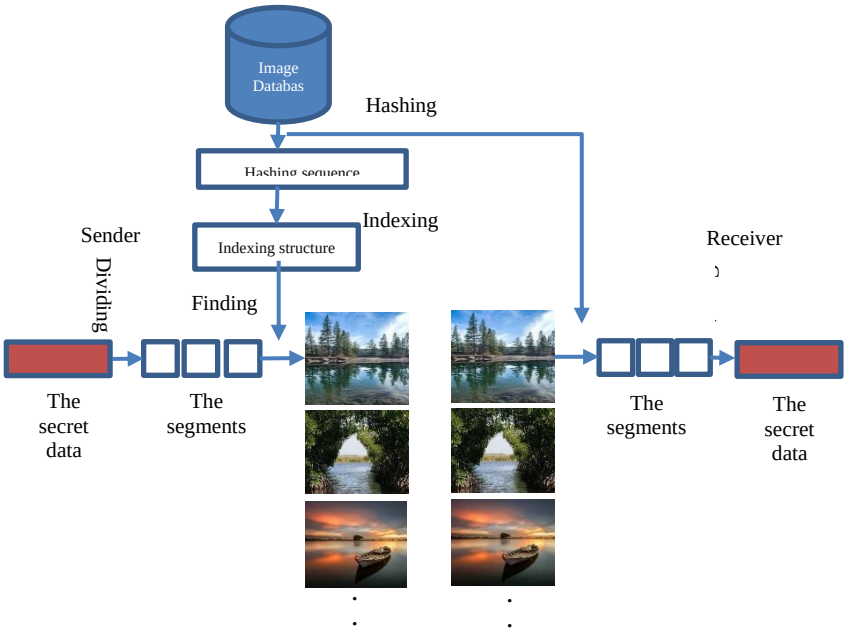

Figure 1: General block diagram of proposed coverless image steganography

3.1 Embedding Procedure
The embedding procedure consists of several activities as follows:

3.1.1 Select the Image
To hide secret data in coverless image steganography based on spatial, one way is to find appropriate images that already contain the information. These images are considered as stego images, which are used for the communication of secret data.

3.1.2 Hash sequence production by robust hashing algorithm

This subsection describes the strong hashing algorithm for creating image hash sequences. Different kind s of attacks could be used to modify the stego images during communication. Rescaling, brightness alteration, contrast improvement, JPEG compression, and noise addition are just a few examples The proposed hashing algorithm is designed to be resistant to most attacks, which would prevent it from modifying the secret sequences during transmission. Several phases can be completed during the production of a hash sequence using algorithm 1.

Algorithm 1: Hash sequence generation
Input: O_image //original image Output: ImgHash //sequence of bits
Step 1: Read the O_ image from the dataset. Step 2: Dividing the O_ image into 3 ×3 non-overlapping segment. Step 3: Computing the average intensity of each segment to obtain 3×3 block average array (OAvg). Step 4: Converting (OAvg) into vector (SAvg) by scanning the array row by row. Step 5: Converting SAvg into binary by comparing every two adjacent elements to obtain binary hash image sequence (ImgHash) with length 8 bits.

The above steps are shown in the figure 2

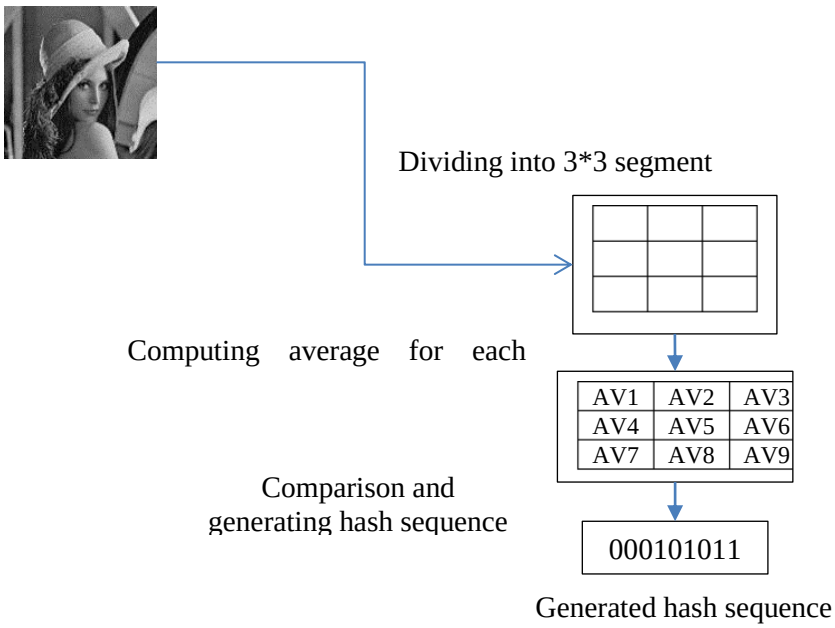

Figure 2: Hash sequence generation

The conversion to binary activity is computed according to the following equation.

$$hj = \begin{cases} 1; \text{ if } hj > hj + 1 \\ 0; \text{ otherwise} \end{cases}$$

, where $1 \leq j < 8$ (1)

2.1.3 Building the inverted index structural by image indexing

For image sequences with a binary hash(ImgHash), it will take a long time to find all images with the same query if we do an extensive search. To speed up the search, the proposed algorithm indexes all of the images in the database according to their hash sequence. Then, for all of the hash sequences, we make a query table T1, which is an inverted index structure. T1 is a lookup table that contains entries to the greatest extent possible Hash sequences of 8 bits. Each value leads to a set of the entire image IXDs that share the same hash sequence. Assume that image A's hash sequence is [1,1,1,0,0,1,1,1] and that its IXD is IXD(A), and that IXD(A) falls into the list pointed by the entry 1,1,1,0,0,1,1,1,1 as shown in Figure 3.

2.1.4 Using the index structure to find suitable images

This subsection describes finding the appropriate image depending on the secret message. The finding procedure is done as the following steps:

a) Transferring the secret message into a bit string that can be sent
b) Dividing the secret message into a number of equal-length segments (8-bit).
c) Obtaining images with hash sequences those are the same with the segment to use each segment as a query.
d) These found Images that are stego images can be considered.

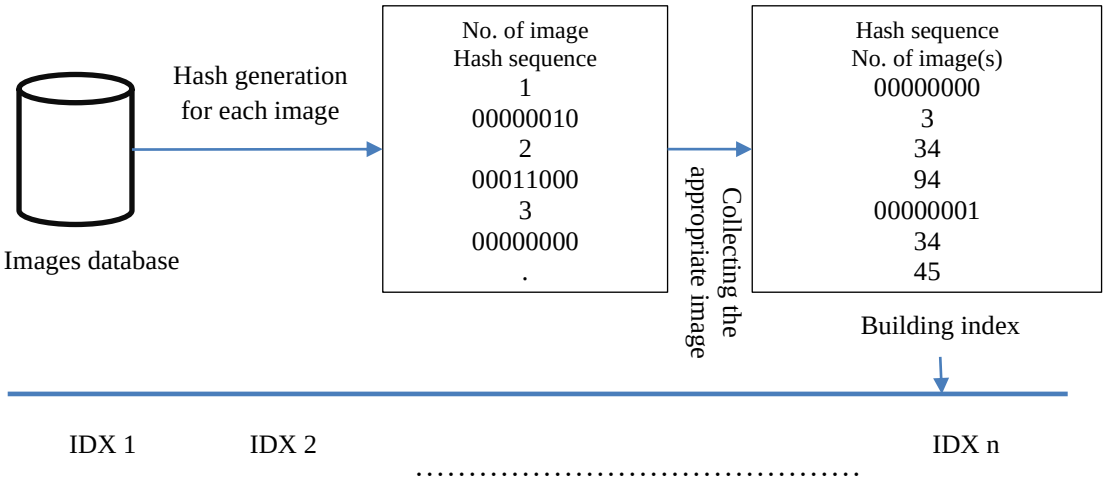

Figure 3: Building index

2.1.5 Sending the stego images

The stego images are sent one by one to the receiver at the sender's end**.**

2.2 Extraction procedure

This section describes extraction the secret message in the receiver side. Herein, the secret data can be recovered without disorder if all stego images are acquired in if all of the stego images are received in order the correct order. Once the receiver has obtained all of the images, the sender uses the same hashing process to generate the hash sequence for each received image (as described in the subsection 3.1.2).To extract the secret data, the receiver generates a new all of their hash sequences of those images according to the sequence of the received images.

3.1 Tests of embedding and extracting procedures

In order to understand the proposed method, suppose that we have a secret message called (computer science) and sample of images dataset as shown in Figure 4.

Image1		Image9	
Image2		Image10	
Image3		Image11	
Image4		Image12	

Image5		Image13	
Image6		Image14	
Image7		Image15	
Image8		Image16	

Figure 4. Sample of images dataset

The embedding procedure is done (in the sender side) as follows:

a) Take the ascii code for each letter in message as shown in Figure 5

99	111	109	112	117	116	101	114	32	115	99	

Figure 5. ASCII code for letters of (computer science) word

b) Generating Hash Sequence for each image. and transforming the into number.

c) Each segment is used as a query; discover the images that have hash sequences that are like as the segment. For example, the number of letter c equal to hash of image1 and the number of letter o equal to hash image2, etc. So, image1 and image2 will be stego image.

d) Sending image1 and image2 to the receiver.

The extracting procedure is done (in the receiver side) as follows:

1. Incoming images' hashes set are computed by the same algorithm used by the sender to generate the corresponding set of value (As described in the subsection 2.1.2). To extract the secret data, the receiver generates a new all of their hash sequences of those images according to the ordering of the received images. So, the result will be as shown in the Figure 6.

99	111	109	112	117	116	101	114	32

Figure 6. Extracted secret message

2. Transforming the ascii in to letters as shown in Figure7.

c	o	m	p	u	t	e	r	-	s	c	i	e	n	c	c

Figure 7. Extracted secret message (as letters)

3.2 Robustness measures

To measure the robustness of the proposed algorithms two measures are used these are normalized correlation (NC) and bit error rate (BER). Normalized correlation is used to measure the resemblance among the extracted secret information and the original ones and Bit error rate is used for measuring the error rate between the extracted secret information and the original. NC is calculated as follows:

$$NC = \frac{\sum_{i=1}^{n} w(i)w'(i)}{\sum_{i=1}^{n} w(i)^2} \quad (2)$$

While BER is calculated as follows:

$$BER = \frac{\sum_{i=1}^{n} w(i) \otimes w'(i)}{n} \quad (3)$$

Where w is the original image and w' is the extracted secret information.

All types of content damage, such as image noise, JPEG compression, rescaling, brightness change, and contrast shift, are unavoidable during the transmission process. and so on. Each stego image selected from the database to represent the secret data segment is subject to these attacks. These variables must be able to withstand the information collected from the image. To put it another way, the hash algorithm is resistant to these types of attacks.

3.2.1 Robust the proposed system against JPEG compression

Each stego image selected from the database to represent the secret data segment is subject to JPEG compression with various quality factors. According Table 1, the proposed system gives a good robustness against JPEG compression attack, where the value of NC and BER is of acceptable.

Table 1. NC and BER values under JPEG compression attacks

Quality factors (Q)	NC	BER
10	0	1
30	0	1
40	0	1
50	0	1

3.2.2 Robustness against noise attacks

In noise attack test, a stego image is attacked with several noise attacks namely, salt &pepper, speckle and Gaussian noises with different noise density. Table 2 shows different types of noise attack.

Table 2. NC and BER values under different density levels of noise

Attack Type	Density of noise	NC	BER
Salt & Pepper	0.001	0	1
	0.01	0	1
	0.02	0	1
	0.03	0	1
Speckle Noise	0.01	0	1
	0.02	0	1
	0.001	0	1
	0.01	0	1

Gaussian Noise	0.02	0	1
	0.001	0	1
	0.1	0.1607	0.8636
	0.4		
Poisson Noise		0	1

3.2.3 Robustness against Filtering Attack

Also, the stego images were filtered with a low pass filtering (Gaussian filter), mean filter and a median filter with window size different sizes of filter kernal. Table 3 illustrates NC and BER values under filtering attack.

Table 3. NC and BER values under filtering attack

Attack Type		BER	NC
Median filter	1×1	0	1
	2×2	0.0268	1
	3×3	0.0268	1
Mean filter	1×1	0	1
	2×2	0.0268	1
	3×3	0	1
Gaussian filter	1×1	0	1
	2×2	0.0268	1
	3×3	0	1

3.2.4 Robustness against geometric attacks

Rotation and resizing Attacks are tests. The proposed method achieved good results in resizing and rotation attack with different rotation degrees as shown in Table 4.

Table 4. NC and BER values under geometric

Attack Type	Angle	BER	NC
Rotation	0.180	0	1
	0.270	0	1
	0.360	0	1
Resize (1024*1024)		0	1

Table 5. NC and BER values Brightness and

Attack Type	Factor	BER	NC
Brightness	+10	0	1
	+20	0	1
Sharpen		0	1

3.2.5 Robustness against brightness and sharpening attacks
The stego images were tested against increasing the brightness image with factor its values (10, 20). Also, the stego images attack with developing the sharpening of image as shown in table 5.

5. Conclusion

The paper presents a framework that allows users to hide the secret information from an image without using a cover image. In this concept, some features are collected form cover which is similar to the secret message based on proposed robust hashing algorithm. The proposed framework can then find the appropriate original image that contains the secret data. Because the proposed framework is able to prevent the traces of modification from leaving in the stego images. Its robust hashing algorithm also ensures that it can be used against various image attacks. Although the proposed framework can hide 8-bit data, it only works with original images. In the future, we will focus on improving the hiding capacity of the proposed framework.

6. References

[1] C. Shen, H. Zhang, D. Feng, Z. Cao, and J. Huang, "Survey of information security," Science in China Series F: Information Sciences, vol. 50, pp. 273-298, 2007/06/01 2007.

[2] R. van Schyndel, A. Tirkel, and C. Osborne, "Towards A Robust Digital Watermark," 01/01 1995.

[3] C. Yang, C. Weng, S. Wang, and H. Sun, "Adaptive Data Hiding in Edge Areas of Images With Spatial LSB Domain Systems," IEEE Transactions on Information Forensics and Security, vol. 3, pp. 488-497, 2008.

[4] V. Holub, J. Fridrich, and T. Denemark, "Universal Distortion Function for Steganography in an Arbitrary Domain," EURASIP Journal on Information Security, vol. 1, 12/01 2014.

[5] T. F. Tomáš Pevný1, and Patrick Bas3, "Using High-Dimensional Image Models to Perform Highly Undetectable Steganography," 2010.

[6] J. Ó. Ruanaidh, W. J. Dowling, and F. M. Boland, "Phase watermarking of digital images," Proceedings of 3rd IEEE International Conference on Image Processing, vol. 3, pp. 239-242 vol.3, 1996.

[7] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 6 12, pp. 1673-87, 1997.

[8] W.-H. Lin, S.-J. Horng, K. tzong-wann, P. Fan, C.-L. Lee, and Y. Pan, "An Efficient Watermarking Method Based on Significant Difference of Wavelet Coefficient Quantization," Multimedia, IEEE Transactions on, vol. 10, pp. 746-757, 09/01 2008.

[9] C. Osborne, A. Tirkel, and T. Hall, "Image and Watermark Registration for Monochrome and Coloured Images," 04/16 2009.

[10] L. Zou, J. Sun, M. Gao, W. Wan, and B. B. Gupta, "A novel coverless information hiding method based on the average pixel value of the sub-images," Multimedia

Tools and Applications, vol. 78, pp. 1-16, 04/01 2019.

[11] X. Zhang, F. Peng, Z. Lin, and M. Long, "A Coverless Image Information Hiding Algorithm Based on Fractal Theory," International Journal of Bifurcation and Chaos, vol. 30, p. 2050062, 2020.

[12] Y. Cao, Z. Zhou, Q. M. J. Wu, C. Yuan, and X. Sun, "Coverless information hiding based on the generation of anime characters," EURASIP Journal on Image and Video Processing, vol. 2020, p. 36, 2020/09/03 2020.

[13] Q. Li, X. Wang, X. Wang, B. Ma, C. Wang, and Y. Shi, "An encrypted coverless information hiding method based on generative models," Information Sciences, vol. 553, pp. 19-30, 2021.