A Novel Approach to Secure Data Hiding in Streak Images

G. Yamuna [1,1] D. Siva Kumar [2]

[1] Department of Electrical Engineering
[2] Department of Instrumentation Engineering
Annamalai University, Annamalai Nagar,
India-608002

Abstract

*A new and efficient steganographic method for concealing a piece of critical information into a gray-valued streak image is proposed. The secret message bits are embedded into Least-Significant-Bit (LSB) of erroneous pixel. The erroneous pixels are first detected using a **S**ignal **D**ependent –**R**ank **O**rdered **M**ean (SD-ROM) technique. A 32-bit binary matrix and a Threshold value were used to protect the hidden information. The number of bits which can be embedded is decided by the erroneous pixels. This method provides an easy way to produce a more imperceptible result than those yielded by simple Least-Significant-Bit replacement methods. The embedded secret message can be extracted from the resulting stego-image without referencing the original cover image. Adjusting threshold values in SD-ROM method to embed more information and reduce the MSE and increase the PSNR values is evident from the Experimental results.*

Keywords: *Steganography; Data Hiding; Rank-Ordered-Mean; LSB Substitution; Streak; SD-ROM.*

1. INTRODUCTION

Steganography is the practice of encoding secret information in a manner such that the very existence of the information is concealed. Steganography is not the same as cryptography. In Cryptography the structure of a message is changed to render it meaningless and unintelligible unless the decryption key is available.

Cryptography makes no attempt to disguise or hide the encoded message, where as Steganography does not alter the structure of the secret message, but hides it inside a cover. It is possible to combine the technique by encrypting message using Cryptography and hiding the encrypted message using Steganography. A review of Steganography along with application was presented by Eugene T. Lin et. al **[1]**.

In a given grayscale image, the LSB of each pixel can be changed to embed the hidden secret was illustrated by R. G. van Schyndel et. al **[2]**. Data hiding schemes based on the conventional pseudo-random number generator was illustrated by Elke Franz et. al **[3].** A protocol that allows public key steganography has been reported by Anderson et. al **[4, 5].** Several techniques and applications for data hiding were discussed by W. Bender et. al. **[6, 7].** Katzenbeisser et. al **[8]** suggested the usage of steganographic methods for the placement of secret message in the noise component of a signal.

As the first part of the present investigation, the SD-ROM technique has been used to detect the erroneous pixel from streaks. The Novelty in this work lies in the fact, that the determination of erroneous pixel from streaks were done by using an arbitrary threshold value given by the user, followed by hiding the data in the streak image.

[1] Corresponding Author. Tel: +91 4144 238641 E-mail Address : yamuna.sky@gmail.com

2. THE PROPOSED DATA-HIDING SCHEME FOR STREAK IMAGES

The proposed scheme uses a binary matrix as the secret key. The EX-OR operator is adopted so that the keys cannot be compromised easily. The inputs to the method are as follows:

i. *c* is a streak image (i.e., bitmap) which is to be modified to conceal data.

ii. *K* is a 32-bit secret key shared by the sender and the receiver.

iii. *r* is the number of erroneous pixels in streak image. *n* is chosen as the number of bytes to be embedded with (n < r).

iv. *m* is the secret information consisting of *n* bytes to be embedded in *c*.

The detailed step by step procedure is as follows:

STEP 1: The total number of erroneous pixels 'r' in the streak image are detected using the SD-ROM method. The number of bytes to be embedded is chosen by Eq. (1).

STEP 2: From the streak image detect erroneous pixel using SD-ROM. If pixel is considered erroneous do next step.

$$n = \frac{r}{8} - 9 \qquad \textbf{(1)}$$

STEP 3: The recovered pixel values from the SD-ROM technique are x(i-1,j-1), x(i-1,j), x(i-1,j+1) and x(i,j-1). This 32-bit pixel value is then EX-ORed with 32-bit secret key matrix.

STEP 4: The above result is component-wise EX-ORed to get 1 bit value and finally EX-ORed with m_i *(i th* message bit).

STEP 5: The final resultant bit is embedded into Least-Significant-Bit of erroneous pixel.

STEP 6: If the embedded pixel value is not erroneous re embed the secret bit using STEP 2-5. Otherwise do STEP 2 until *n* number of bytes to be embedded.

3. DETECTION OF ERRONEOUS PIXELS

A streak can be any sequence of pixels in the image which has been replaced with random values. In the present method, entire rows of the images are arbitrary sequence of values ranging from 0 to 255. The SD-ROM technique developed by Abreu et. al **[9]** has been used here to detect the erroneous pixel.

Let x(n) denote the luminance values of image at pixel location $n = [n_1,n_2]$. Consider a 3x3 window centered at x(n) and define w(n) as an eight-element observation-vector containing the neighboring pixels of x(n) inside the window, (excluding x(n), itself) which is given in Eq. (2).

$$\begin{aligned} w(n) &= [\, w_1(n),\, w_2(n), \ldots\ldots, w_8(n) \,] \\ &= [\, x(n_1-1,n_2-1),\, x(n_1-1,n_2),\, x(n_1-1,n_2+1), \\ &\quad x(n_1,n_2-1),\, x(n_1,n_2+1), \\ &\quad x(n_1+1,n_2-1),\, x(n_1+1,n_2),\, x(n_1+1,n_2+1) \,] \end{aligned} \qquad \textbf{(2)}$$

The observation samples are also ordered by rank to define the vector in Eq. (3).

$$r(n) = [\, r_1(n), r_2(n), \ldots\ldots, r_8(n) \,] \quad \textbf{(3)}$$

Where $r_1(n), r_2(n), \ldots\ldots, r_8(n)$ are the elements of w(n), they are arranged in the ascending order, like $r_1(n) \le r_2(n) \le \ldots\ldots \le r_8(n)$

Finally, the rank-ordered differences are defined by Eq. (4):

$$d_i(n) = \begin{cases} r_i(n) - x(n), & x(n) \le m(n) \\ x(n) - r_{9-i}(n), & x(n) > m(n) \end{cases} \quad \textbf{(4)}$$

for $i = 1,\ldots,4$, where $m(n) = (r_4(n) + r_5(n)) / 2$ (Rank-Ordered-Mean or ROM).

The algorithm detects x(n) as a noisy sample if any of the following inequalities are true:

$$d_i(n) > T_i, \quad i = 1,\ldots,4 \quad \textbf{(5)}$$

where T_1, T_2, T_3, T_4 are threshold values with $T_1 < T_2 < T_3 < T_4$.

If x(n) is detected as a corrupted sample, the secret message bit is to be embedded, otherwise it is kept unchanged.

When streaks are present in the image, the SD-ROM is capable of effectively detecting pixels with a high probability of belonging to a streak. Suggested values of the thresholds are given in Table 1

5. EXPERIEMNTAL RESULTS

The proposed scheme in the present experiment is to visualize the data-hiding effect. For the same, two steak cover images “GIRL” Fig. 1(a) and “BOAT” Fig. 2(a) were used, each with a size of 256 x 256

The first experiment was based on selecting the lower threshold values, embedding 98 bytes in “GIRL” cover-image, as a result and its MSE is 0.006607 and PSNR is 69.930724 dB as shown in Figs. 1(b), Normal threshold values, embedding 86 bytes in “GIRL” cover image and in its MSE is 0.005981 and PSNR is 70.362742 dB are shown in Figs. 1(c) and higher threshold values, embedding 63 bytes in “GIRL” cover image and in its MSE is 0.004303 and PSNR is 71.793112 dB shown in Figs. 1(d).

The second experiment was based on selecting the lower threshold values, embedding 122 bytes in “BOAT” cover-image and its MSE is 0.008316 and PSNR is 68.931638 dB shown in Figs. 2(b), Normal threshold values, embedding 94 bytes in “BOAT” cover image and in its MSE is

0.006424 and PSNR is 70.052782 dB shown in Figs. 2(c) and higher threshold values, embedding 57 bytes in "BOAT" cover image and in its MSE is 0.003952 and PSNR is 72.162605 dB shown in Figs. 2(d).

The Table 2 shows the number of bytes embedded, MSE and PSNR values for the two test images with various threshold values. Our simulation results shows that Mean Squared Error (MSE) is reduced and Peak to Signal Noise Ratio (PSNR) is increased.

The same experiments were carried out using the simple LSB replacement technique. The results of these experiments are as follows: for "BOAT" cover image MSE is 0.501389, PSNR is 51.129059dB and for "GIRL" cover image MSE is 0.50116, PSNR is 51.131042 dB.

The experiments for the data hiding scheme followed in the present experiments produced excellent imperceptible quality when compared with those yielded by simple Least-Significant-Bit replacement methods.

Images showing the difference in pixel between the steak and embedded images for a low threshold value are illustrated in Figure 3(a) for GIRL and 3(b) for BOAT, similarly for Medium Threshold and High Threshold values, the illustrating figures are 4(a), 4(b) and 5(a), 5(b) respectively for GIRL and BOAT.

5. CONCLUSION

Thus the paper presents a novel Steganography scheme capable of concealing a large amount of secret information in a streak image which depends on the threshold value. The proposed scheme has the following feature: it uses a secret key and threshold value to protect the hidden data, and it uses an EX-OR operator to increase the security. A pseudo-random mechanism may be used to achieve secrecy protection.

Fig. 1(a)

Fig 1(b)

Fig. 1(c)

Fig. 1(d)

Fig. 2(a)

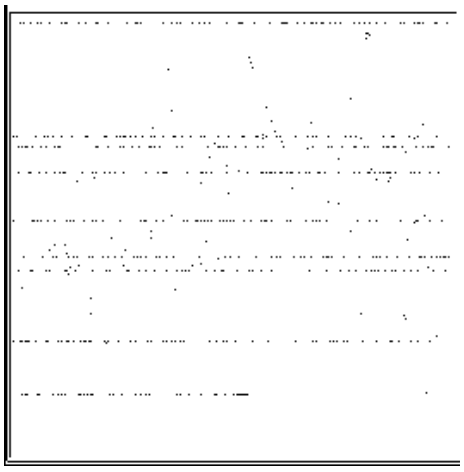
Fig 2(b)

Fig. 2(c)

Fig. 2 (d)

Fig 3(a)

Fig. 3(b)



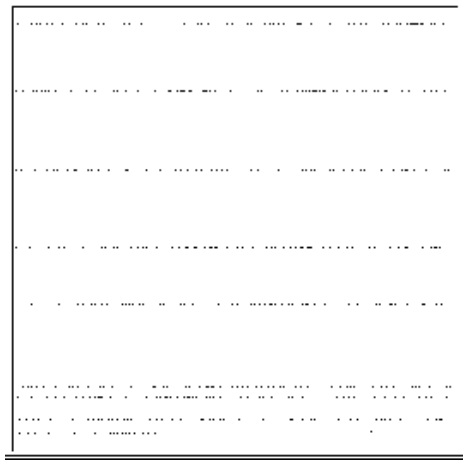
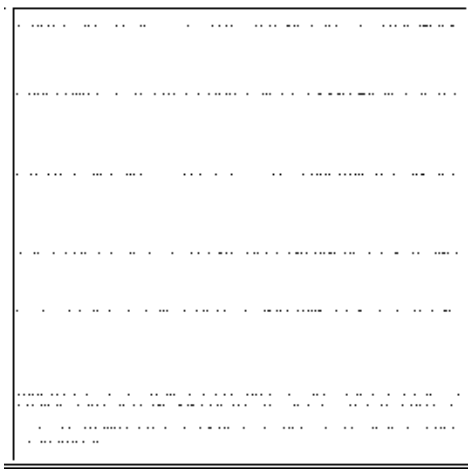


Fig. 4(a)

Fig. 4(b)



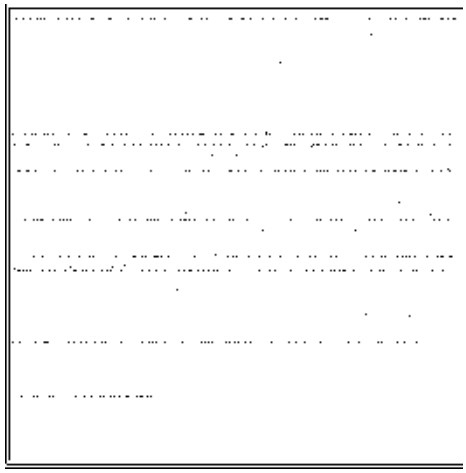
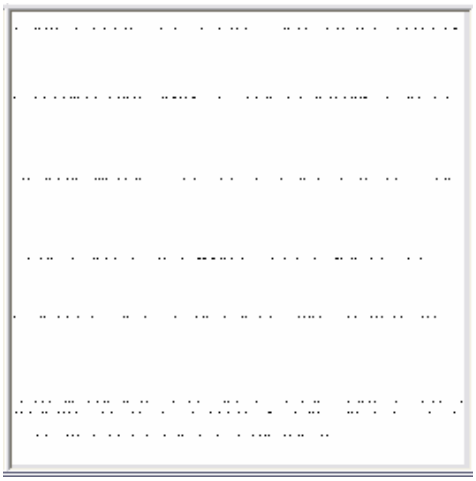


Fig. 5(a)

Fig. 5(b)



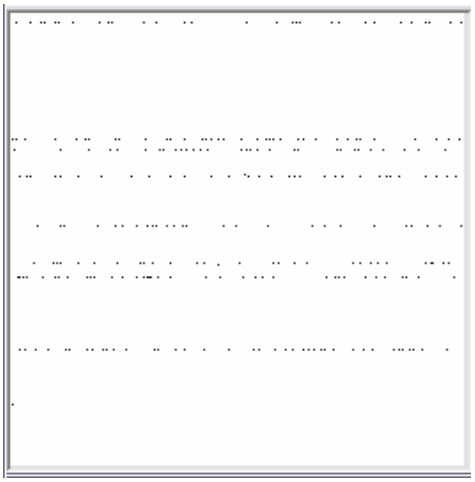


Table 1. Suggested Values of the threshold values

Threshold Value	T1	T2	T3	T4
Lower Values	4	15	30	40
Normal Values	8	20	40	50
Higher Values	15	30	50	80

Table 2. Comparison of number of bytes embedded, MSE and PSNR values for the two test images with various threshold values

Test Image	Threshold Value	No. of Bytes Embedded	MSE	PSNR
GIRL	Lower	98	0.006607	69.930724
	Normal	86	0.005981	70.362742
	Higher	63	0.004303	71.793112
BOAT	Lower	122	0.008316	68.931638
	Normal	94	0.006424	70.052782
	Higher	57	0.003952	72.162605

FIGURE LEGENDS

1. Fig. 1(a) Original Streak GIRL Image
2. Fig. 1(b) After embedding 98 bytes by lower threshold value
3. Fig. 1(c) After embedding 86 bytes by normal threshold value.
4. Fig. 1(d) After embedding 63 bytes by higher threshold value.
5. Fig. 2(a) Original Streak BOAT Image
6. Fig. 2(b) After embedding 122 bytes by lower threshold value
7. Fig. 2(c) After embedding 94 bytes by normal threshold value
8. Fig. 2(d) After embedding 57 bytes by higher threshold value
9. Fig. 3(a) Difference in Pixel between the original and the Embedded Image Using Low Threshold for GIRL.
10. Fig. 3(b) Difference in Pixel between the original and the Embedded Image Using Low Threshold for BOAT.
11. Fig. 4(a) Difference in Pixel between the original and the Embedded Image Using Medium Threshold for GIRL.
12. Fig. 4(b) Difference in Pixel between the original and the Embedded Image Using Medium Threshold BOAT.
13. Fig. 5(a) Difference in Pixel between the original and the Embedded Image Using High Threshold GIRL.
14. Fig. 5(b) Difference in Pixel between the original and the Embedded Image Using High Threshold BOAT.

REFERENCES

1. Eugene T, Lin., and Edward J, Delp, A review of data hiding in digital images, Video and Image Processing Laboratory (VIPER), School of Electrical and Computer Engineering, Purde University, West Lafayette, Indiana.

2. Van Schyndel, R, G., Tirkel,A, Z., and Osborne, C, F.(1994), A digital watermark, Procedings IEEE International Conference on Image Processing, 2, 86-90.

3. Elke Franz., Anja Jerichow., Steffen Möller., Andreas Pfitzmann.,and Ingo Stierand. (1996), Computer based steganography: How it works and why therefore any restrictions on cryptography are nonsense, at best, 7 – 21.

4. Anderson, J, R. (1996), stretching the limits of Steganography, Information Hiding, Springer Lecture Notes in Computer Science, 1174, 39-48.

5. Anderson, R, J., and Petitcolas, F, A, P. (1998), on the limits of Steganography, IEEE J. Select. Areas Commun., 16, 474-481.

6. Bender, W., Gruhl, D., Morimoto, N., and Lu, A. (1996), Techniques for data hiding, IBM Systems Journal, 35, 3-4.

7. Bender, W., Butera, W., Gruhl, D., hwang, R., Palz, F, J., and Pogerb, S, (2000), Applications for Data Hiding, IBM Systems Journal, 39, 547-568.

8. Katzenbeisser, S., and Petitcolas, F, A, P. (2000), "Information Hiding Techniques for Steganography and Digital Watermarking", Artech House. Norwood. MA-02062, 28-29.

9. Abreu, E., Lightstone, M., Mitra, S., and Arakawa, K. (1996), A new efficient approach for the removal of impulse noise from highly corrupted images, IEEE Trans. Image Processing, 5, 1012-1025.

Article received: 2007-07-28