USING SECOND LAST BIT AS AN INDICATOR FOR IMAGE STEGANOGRAPHY

Manish Pandey[1], Shreyash Gajbhiye[2]

Abstract—As in today's generation, the transfer of information is usually done through the internet, the biggest concern is to maintain the secrecy and security of the transferrable data. Steganography is considered one of the good methods of transferring data with highest forms of secrecy, also as the sizes of images permit us to hide messages long enough thus this form of transferring data is considered as one of the best forms. In our article we hereby mention a replacement method of Image steganography encoding.

According to our proposed method of image steganography we first identify what the present pixel represents consistent with our general method of representation. This can be done by observing the 2 nd last bit of the binary representation of the pixel value and the 2nd last bit of the binary representation of the pixel + 1 value. Once we identify that then we use our mentioned scheme of altering the last two bits such that they produce minimal error for the stego image. We do this encoding until all the message bits are encoded. For decoding what we do is just identify the pixel value in binary at that pixel and find out what it represents according to our general method of representation table.

I. INTRODUCTION

Steganography refers to the hiding information i.e. images, texts or audio files in another image file. At the sender's side, the image used for embedding the secret message is called cover image, and the secret information that needs to be protected is called a message.

1Manish Pandey, Assistant Professor, Department of Computer Science and Engineering, NIT Bhopal
2Shreyash Gajbhiye, Student, Department of Computer Science and Engineering, NIT Bhopal

As soon as data is embedded using some appropriate embedding algorithm, then it is called a stego image. This stego image is transferred to the receiver, and then it extracts out the secret message using an extraction algorithm. Sometimes the hidden message can also be encrypted to make it even more difficult for an attacker to decode[1] . The current project aims to use a newly proposed method of steganography for an image with another image using spatial domain technique. This hidden information is often retrieved only through proper and a specific decoding algorithm.

II. LITERATURE SURVEY

The area of steganography has been recently blooming in the computer science research community. Nowadays the most famous kind of steganography used is image steganography, that is embedding a message in an image. One such method used for image steganography is LSB[2] . This method basically tweaks a little with the classic LSB and other similar methods [3][4][5]. .In this method the last 2 bits of a pixel in an image are replaced with 2 bits of a message. Thus, the maximum difference between a pixel value of the original image and the stego image can be 3.

This method is definitely effective and gives an average PSNR value around 45 - 60 (depending on the size of the message). But the very simplistic nature of the implementation of this method makes

it kind of easy for an attacker to identify this.

In this paper we have suggested a similar method, which makes the implementation still easy but not so easy to judge for an attacker. Also does this method manage to provide a better result by providing a lesser PSNR value than the regular LSB method.

III. METHODOLOGY AND WORK DESCRIPTION

The method that we've proposed uses the second last little bit of a pixel as an indicator. Our general method of representation is such the message that's being represented by a pixel is identified as follows:

$$M(Pi) = 2^{nd} \text{ last bit of } Pi + 2^{nd} \text{ last bit of } (Pi + 1)$$

To understand it in a better way let us take an example. Say we have to find out what the pixel value of 64 represents according to our method?

To find out what 64 represents, we first convert 64 and 64 + 1 into bit representation.

Now, we take the value of the second last bit of both 64 and 65 and combine i.e Second last bit of 64 + second last bit of 65 ⇒ "0" + "0" = "00"
Thus, 64 represents the message "00"

position	8th	7th	6th	5th	4th	3rd	2nd	1st
64	0	1	0	0	0	0	0	0
64 +1 = 65	0	1	0	0	0	0	0	1

Similarly to find out for 65

Second last bit of 65 + second last bit of 66 ⇒ "0" + "1" = "01"
This is the way how we identify what value is represented by a pixel value by using the 2nd last bit as an indicator.

position	8th	7th	6th	5th	4th	3rd	2nd	1st
65	0	1	0	0	0	0	0	1
65 +1 = 66	0	1	0	0	0	0	1	0

USING THIS REPRESENTATION FOR STEGANOGRAPHY:

Suppose the message to be hidden in an image I = (132,69,32,244) is "01101111".

Breaking the message in 4 sections of 2 bits each.
m = (01 , 10 , 11 , 11)

Let,

I1 = 132	m1 = 01
I2 = 69	m4 = 10
I3 = 32	m3 = 11
I4 = 244	m4 = 11

For I1 = 132 and m1 = 01

position	8th	7th	6th	5th	4th	3rd	2nd	1st
132 - 1 = 131	1	0	0	0	0	0	1	1
132	1	0	0	0	0	1	0	0
132 + 1 = 133	1	0	0	0	0	1	0	1
132 + 2 = 134	1	0	0	0	0	1	1	0
132 + 3 = 135	1	0	0	0	0	1	1	1

By our method "01", out of the following numbers is represented by 133.
Therefore I'1 = 133

Similarly for, I2 = 69 and m2 = 10

position	8th	7th	6th	5th	4th	3rd	2nd	1st
69 - 1 = 68	0	1	0	0	0	1	0	0
69	0	1	0	0	0	1	0	1
69 + 1 = 70	0	1	0	0	0	1	1	0
69 + 2 = 71	0	1	0	0	0	1	1	1
69 + 3 = 72	0	1	0	0	1	0	0	0

By our method "10", out of the following numbers is represented by 71.
Therefore I'2 = 71

Similarly, for I3 = 32 and m3 = 11

position	8th	7th	6th	5th	4th	3rd	2nd	1st
32 - 1 = 31	0	0	0	1	1	1	1	1
32	0	0	1	0	0	0	0	0
32 + 1 = 33	0	0	1	0	0	0	0	1
32 + 2 = 34	0	0	1	0	0	0	1	0
32 + 3 = 35	0	0	1	0	0	0	1	1

By our method out of the following numbers "11" is represented by 34.
Therefore I'3 = 34

Similarly, for I4 = 244 and m4 = 11

position	8th	7th	6th	5th	4th	3rd	2nd	1st
244 - 1 = 243	1	1	1	1	0	0	1	1
244	1	1	1	1	0	1	0	0
244 + 1 = 245	1	1	1	1	0	1	0	1
244 + 2 = 246	1	1	1	1	0	1	1	0
244 + 3 = 247	1	1	1	1	0	1	1	1

By our method out of the following numbers 246 represents "11".
Therefore I'4 = 246
Thus we have,

ORIGINAL IMAGE ⇒ (132, 69, 32, 244)
STEGANOGRAPHIC IMAGE by our method ⇒ (133, 71, 34, 246)
STEGANOGRAPHIC IMAGE by LSB ⇒ (133, 70, 35, 247)

MSE for our method = 3.25
MSE for LSB method = 5

While decoding the image on the other end, we can use the 7th bit of the pixel to identify what the message bits are.

IV. CALCULATIONS

The efficiency is checked on the basis of two parameters, that is, PSNR (peak signal to noise ratio) and MSE (mean square error). Obtained values show the high efficiency of the proposed method:

$$\mathrm{MSE} = \frac{1}{R \times C} \sum_{i=1}^{R} \sum_{j=1}^{C} \left(x_{ij} - x'_{ij} \right)^2$$

Where R and C represent the dimensions of the image matrix, Xij represents the original image, and X'ij represents the stego image where I represents the maximum possible value of the pixel in an image. PSNR is measured in decibel.

$$\mathrm{PSNR} = 10 \log_{10} \left[\frac{I^2}{\mathrm{MSE}} \right] (\mathrm{dB})$$

V. RESULTS

We have implemented this method and compared it with the regular LSB method and it appears to give a 5% rise in efficiency in terms of PSNR value. The following are the results observed on varying size of message length by using our proposed method and the LSB method of steganography.
NOTE : All the PSNR values are in Decibel (dB).

LENGTH OF MESSAGE = 32000 characters

PSNR FOR 7th BIT	PSNR FOR LSB	DIFFERENCE	BETTER PERFORMANCE IN %
56.49714	53.73938	2.75777	5.13174
56.94541	54.11973	2.82568	5.22116
57.60440	54.83120	2.77320	5.05771
61.85253	59.08430	2.76823	4.68522
65.28524	62.48908	2.79616	4.47464
65.96371	63.19688	2.76683	4.37812
53.29341	50.78210	2.51131	4.94527
64.15118	61.39124	2.75993	4.49565
57.25279	54.46507	2.78772	5.11836
67.09968	64.31997	2.77971	4.32168
62.41962	59.64952	2.77010	4.64396
62.32956	59.53238	2.79718	4.69859
65.82837	63.06002	2.76835	4.39003
65.34128	62.58379	2.75749	4.40607
64.10806	61.34228	2.76578	4.50877
56.49680	53.69577	2.80103	5.21649
64.90310	62.14118	2.76192	4.44458
64.90310	62.14118	2.76192	4.44458
60.72889	57.95725	2.77165	4.78223
51.08259	49.71622	1.36638	2.74835
62.10452	59.37109	2.73343	4.60397
56.49473	53.51730	2.97743	5.56349
53.51900	52.39088	1.12812	2.15327
57.79248	56.58949	1.20299	2.12581
AVERAGE			
60.58323	58.00447	2.57876	4.43999

LENGTH OF MESSAGE = 64000 characters

PSNR FOR 7th BIT	PSNR FOR LSB	DIFFERENCE	BETTER PERFORMANCE IN %
53.48014	50.72921	2.75093	5.42278
53.93472	51.10786	2.82686	5.53117
54.57933	51.86213	2.71719	5.23926
58.83480	56.05944	2.77536	4.95075
62.28196	59.49462	2.78734	4.68503
62.95792	60.18310	2.77482	4.61063
50.32086	47.77247	2.54839	5.33444
61.13908	58.38181	2.75727	4.72283
54.26599	51.48800	2.77799	5.39541
64.07871	61.30819	2.77053	4.51902
59.40698	56.63496	2.77202	4.89453
59.31314	56.50821	2.80493	4.96375
62.82444	60.05302	2.77142	4.61495
62.33666	59.59320	2.74346	4.60365
61.09133	58.33536	2.75597	4.72435
53.49379	50.70911	2.78468	5.49148
61.89910	59.12722	2.77188	4.68799
61.89910	59.12722	2.77188	4.68799
57.72513	54.94748	2.77765	5.05510
48.76231	47.15037	1.61194	3.41871
59.10030	56.45170	2.64860	4.69180
53.46126	50.52808	2.93318	5.80505
50.50870	49.38058	1.12812	2.28454
54.72642	53.50484	1.22158	2.28312
AVERAGE			
57.60092	55.01826	2.58267	4.69243

LENGTH OF MESSAGE = 128000 characters

PSNR FOR 7th BIT	PSNR FOR LSB	DIFFERENCE	BETTER PERFORMANCE IN %
50.45206	47.70985	2.74220	5.74767
50.91798	48.05610	2.86187	5.95527
51.54641	48.79498	2.75143	5.63875
55.81454	53.04182	2.77272	5.22742
59.25998	56.48332	2.77666	4.91589
59.94780	57.16852	2.77928	4.86156
47.30685	44.74236	2.56448	5.73167
58.13805	55.37165	2.76640	4.99605
51.24565	48.46538	2.78026	5.73659
61.07112	58.30139	2.76973	4.75071
56.40297	53.62647	2.77651	5.17749
56.29452	53.50238	2.79214	5.21872
59.81225	57.04502	2.76723	4.85096
59.32146	56.57817	2.74329	4.84867
58.08174	55.31613	2.76561	4.99964
50.43952	47.71988	2.71964	5.69917
58.88604	56.11319	2.77285	4.94154
58.88604	56.11319	2.77285	4.94154
54.70817	51.93735	2.77082	5.33493
46.81862	44.76636	2.05226	4.58437
56.08296	53.40309	2.67988	5.01821
50.40393	47.52264	2.88129	6.06298
47.58253	46.44003	1.14251	2.46018
51.63762	50.49413	1.14349	2.26460
AVERAGE			
54.62745	52.02973	2.59772	4.99852

LENGTH OF MESSAGE = 256000 characters

PSNR FOR 7th BIT	PSNR FOR LSB	DIFFERENCE	BETTER PERFORMANCE IN %
47.43424	44.68023	2.75401	6.16383
47.89706	45.06768	2.82938	6.27806
48.52141	45.77339	2.74802	6.00352
52.81277	50.04222	2.77055	5.53643
56.24707	53.47568	2.77139	5.18252
56.93164	54.15956	2.77208	5.11835
44.32252	41.73374	2.58878	6.20308
55.12661	52.36027	2.76633	5.28327
48.22896	45.46934	2.75962	6.06920
58.05956	55.29408	2.76549	5.00141
53.39418	50.62177	2.77242	5.47673
53.28585	50.50648	2.77937	5.50299
56.79870	54.03323	2.76547	5.11809
56.30670	53.53635	2.77034	5.17469
55.07153	52.30512	2.76640	5.28897
47.40334	44.84388	2.55946	5.70748
55.87322	53.10221	2.77101	5.21826
55.87322	53.10221	2.77101	5.21826
51.69697	48.92788	2.76909	5.65954
44.47390	42.13280	2.34110	5.55649
53.07563	50.33928	2.73635	5.43581
47.39988	44.56920	2.83068	6.35121
45.09812	43.92894	1.16918	2.66154
48.48053	47.48783	0.99269	2.09042
AVERAGE			
51.65890	49.06222	2.59668	5.30417

VI. CONCLUSION

The above discussed steganography method allows high capacity of knowledge to be hidden inside the grey carrier image. Each pixel stores two bits of message bit inside the pixel, whereas other methods like LSB allow just one little bit of message hiding inside every pixel. Our method doesn't entertain its dependency over the 8th bit as that's found within the case of LSB method.

As far as error cares , a maximum change of +2 or −2 is entertained while transferring the stego image. In the same manner, messages can be extracted at the receiver side by using the same method on the stego image. One of the major demands of the good steganography method is to provide good PSNR and MSE values. Our method provides high PSNR and low MSE values when compared with other methods.

VII. REFERENCES

[1]Naveen Verma, Preeti Sondhi, Gargi Kalia, K., 2019. Paper on LSB Based Steganography to Enhance Image Security. International Journal of Trend in Scientific Research and Development (IJTSRD).

[2]Bhaskar, Abhijeet and Upendra Kumar, T., 2019. Image Steganography Using Modified LSB. International Journal of Engineering Research & Science (IJSER).

[3] Thenmozhi, M.J. and Menakadevi, T., 2016. A New Secure Image Steganography Using Lsb And Spiht Based Compression Method. International Journal of Engineering Research & Science (IJOER), 2(3), pp.81-85.

[4]Shabnam, S. and Hemachandran, K., 2016. LSB based Steganography using Bit masking method on RGB planes. IJCSIT) International Journal of Computer Science and Information Technologies, 7(3), pp.1169-1173.

[5]Datta, B., Mukherjee, U. and Bandyopadhyay, S.K., 2016. LSB Layer Independent Robust Steganography using Binary Addition. Procedia Computer Science, 85, pp.425-432.