IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 6, Ver. IV(Nov – Dec. 2014), PP 01-06
www.iosrjournals.org

Video Steganography Using LSB Matching Revisited Algorithm

R. Shanthakumari[1] and Dr.S. Malliga[2]
[1] *Department of Information Technology, Kongu Engineering College, India.*
[2]*Department of Computer Science and Engineering, Kongu Engineering College, India.*

Abstract: *Video Steganography deals with hiding secret data or information within a video. In this paper, a spatial domain technique for LSB Matching Revisited algorithm (LSBMR) has been proposed, where the secret information is embedded in the cover frames. LSB Matching Revisited(LSBMR) algorithm selects the embedding regions according to the size of secret message and the difference between two consecutive pixels in the cover image. For embedding rates is lower , only sharper edge regions are used while keeping the other smoother regions as they are. In the proposed approach, LSB Matching Revisited algorithm is used to embed the secret message into the video. Hence large amounts of data can be embedded and also preserving higher visual quality of stego images at the same time.The proposed method is analyzed in terms of both Peak Signal to Noise Ratio (PSNR) compared to the original cover video as well as the Mean Squared Error (MSE) measured between the original and steganographic files averaged over all video frames.*
Key Words: *Frames, LSBMR , Threshold ,Video Steganography,Visual Quality.*

I. Introduction

Steganography is hiding private or secret data within a carrier in invisible manner. It derives from the Greek word steganos, meaning covered or secret and graphy (writing or drawing) [1]. The medium where the secret data is hidden is called as cover medium which can be an image, video or an audio file. Any stego algorithm removes the redundant bits in the cover media and inserts the secret data into the space. Higher the quality of video or sound, more redundant bits are available for hiding. Application of Steganography varies from military, industrial applications to copyright and Intellectual Property Rights (IPR). By using lossless steganography techniques, messages can be sent and received securely [2]. Traditionally, steganography was based on hiding secret information in image files. But modern work suggests that there has been growing interest among research fraternity in applying steganographic techniques to video files as well [3], [4]. The advantage of using video files in hiding information is the added security against the attack of hacker due to the relative complexity of the structure of video compared to image files.

Video based steganographic techniques are broadly classified into temporal domain and spatial domain. In frequency domain, images are transformed to frequency components by using FFT,DCT or DWT and then messages are embedded in some or all of the transformed coefficients. Embedding may be bit level or block level. Moreover, in spatial domain the bits of the message can be inserted in intensity pixels of the video in LSB positions. The advantage of the method is that the amount of data (payload) that can be embedded is more in LSB techniques. However most of the LSB techniques are prone to attacks as described in [5] and [6]. This makes research fraternity interested in designing new methods.

In this paper an LSBMR algorithm is proposed in spatial domain. Application of the algorithm is illustrated with AVI (Audio Video Interleave) file as a cover medium. The rest of the paper is arranged as follows, section 2 does Literature survey of the recent steganographic techniques. In section 3 the proposed video steganographic technique has been described.The proposed algorithm is in section 4. Section 5 gives results and performance evaluation of LSBMR technique. Conclusion and future work are presented in Section 6.

II. Literature Survey

Several steganographic methods have been proposed in literature and most of which are performed in pixel domain. However major contribution is in the domain of Image steganography. The existing methods are mainly based on LSB where LSBs of the cover file are directly changed with message bits. In [7] a robust image steganography technique based on LSB insertion and RSA encryption technique has been used. Masud et al [8] proposed an LSB technique for RGB true color image by enhancing the existing LSB substitution techniques to improve the security level of hidden information. Other Examples of LSB schemes can be found in [9], [10]. Whereas EzStego developed by Machado [11] embed information into an image in the GIF format. It sorts the palette to ensure the difference between two adjacent colors is visually indistinguishable. Tseng and Pan [12] presented a data hiding scheme in 2-color images, it embeds the information in any bit where at least one of the adjacent bits is the same as the original unchanged bit. Kawaguchi et. al. [13] proposed bit plane complexity segmentation (BPCS) method to embed information into the noisy areas of the image. These techniques are not

limited to the LSB. Video steganography of late has also gained significance for researchers. Various techniques of LSB exists, where [14] proposes that the data is first encrypted using a key and then embedded in the carrier AVI video file in LSB keeping the key of encryption in a separate file called key file. Whereas in [15] selected LSB steganography algorithm is proposed. Other steganography techniques in uncompressed raw video are illustrated in [16], [17] and [18]. Steganography techniques for compressed video stream can be found in [19], [20] and [21].Video steganography scheme based on motion vectors and linear block codes has been proposed in [22].

III. Proposed Technique

LSB Matching Revisited (LSBMR) technique for Video Steganography has been proposed. The flow diagram of the encoding and decoding is given in Fig 1and Fig 2.

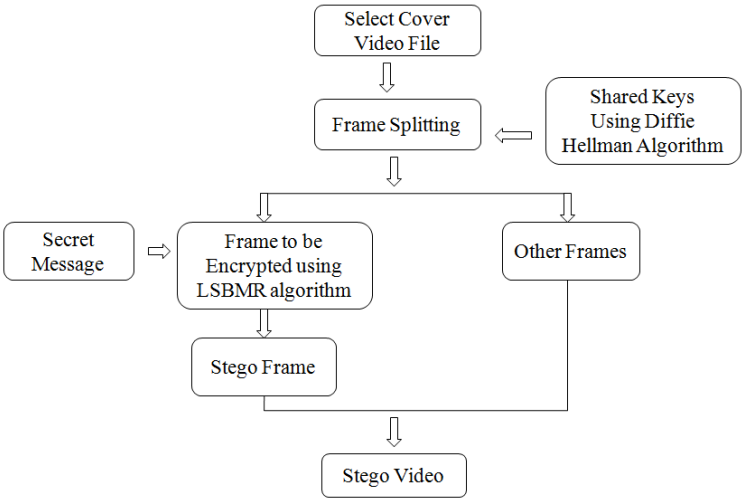

Fig 1: Block diagram of LSBMR Video Steganography technique - Encoding

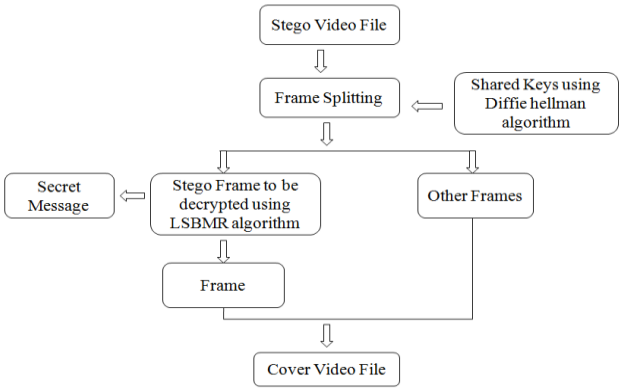

Fig 2: Block diagram of LSBMR Video Steganography technique – Decoding

A Video Stream (AVI) consists of collection of frames and the secret data is embedded in these frames as payload. The cover video is then broken down into frames. Now the proposed LSB Matching Revisited technique is applied to conceal the data in the carrier frames. The size of the message does not matter in video steganography as the message can be embedded in multiple frames. After concealing data in multiple frames of the carrier video, frames are then grouped together to form a stego video, which is now an embedded video. In the receiver side, the reverse steps are used to decode the secret data. During decoding, the stego video is again broken into frames. The secret message is extracted from the stego video.

Reducing distortion between the cover image and the stego image is an important issue for steganography. Most of the steganographic methods usually use randomly selected pixels for data embedding. These pixels are selected without considering adjacent pixel values. In such cases the probability of embedding in the smooth regions will be high. Generally, the sharper regions have more complicated statistical features and random characteristics than that at the smoother ones. It is expected that detectable and visual artifacts would be left very low in the sharper regions after data embedding. It makes the detection more difficult. In this paper, data embedded in the video using LSBMR algorithm is analyzed. The details of data embedding and data extraction algorithms are as follows.

3.1 LSBMR Algorithm

The proposed algorithm, both for encoding and decoding are given in this section. Encoding technique is given in section 3.1.1 and decoding technique in section 3.1.2.

3.1.1. Algorithm for Encoding

Step 1 : Dividing Video into Frames

The cover video file is decomposed into number of frames in which the secret message will be hidden. Shared key is used to select the frame for hiding the message.

Step 2 : Calculating the key using Diffie Hellman Algorithm

The Diffie-Hellman key exchange method allows two parties who have no prior knowledge of each other to jointly establish a shared secret key over a secure communication channel. Consider, g as the base, n as a very large prime number or generator. Both the numbers n and g are selected such that n > g and g is the primitive root of n. x and y are secret keys of A and B respectively.

Step 2.1 : A computes g^x mod n where x is A's secret key and sends along with n and g to B

Step 2.2 : B on receiving the message, extracts n and g.

Step 2.3 : B computes g^y mod n and sends to A.

Step 2.4 : At A, ((g^y mod n)^x) mod n is calculated which is equivalent to g^xy mod n

Similarly at B, ((g^x mod n)^y) mod n is calculated which is equivalent to g^xy mod n. Thus the received key is same at both the ends. This session key was used to encrypt the data which can be transmitted successfully. Communication channel eavesdropper will end up with a value which doesn't mean anything.

Step 3 : Embedding the text

In the data embedding stage, the scheme first initializes some parameters, which are used for subsequent data preprocessing and region selection, and then estimates the capacity of those selected regions. If the regions are large enough for hiding the given secret message, then data hiding is performed on the selected regions. Finally, it does some post processing to obtain the stego image.

Step 3.1: The cover image of certain size is divided into non-overlapping blocks of pixels. For each small block, we rotate it by a random degree in the range of, as determined by a secret key. The resulting image is rearranged as a row vector by raster scanning. Then the vector is divided into non-overlapping embedding units with every two consecutive pixels, these pixels can be used to generate the pseudorandom number which can be either an even or an odd number. Two benefits can be obtained by the random rotation. First, it can prevent the detector from getting the correct embedding units without the rotation key which improves security. Second, both horizontal and vertical edges (pixel pairs) within the cover image can be used for data hiding.

Step 3.2: According to the scheme of LSBMR, 2 secret bits can be embedded into each embedding unit. Therefore, for a given secret message, the threshold for region selection can be determined as follows. Let be the set of pixel pairs whose absolute differences are greater than or equal to a parameter t, $EU(t)=\{ (x_i,x_{i+1}) \| x_i-x_{i+1} | \geq t , (x_i,x_{i+1}) \in V\}$.Then we calculate the threshold T by T=arg max $\{ 2X | EU(t) | \geq M | \}$ where $t\{0,1,....,31\}$,$|M|$ is the size of the secret message M , and EU(t)| denotes the total number of elements in the set of EU(t).

Step 3.3: Performing data hiding on the set of $EU(t)=\{(x_i,x_{i+1})\|x_i-x_{i+1}\geq t,(x_i,x_{i+1})V\}$,
We deal with the above embedding units in a pseudorandom order determined by a secret key key_2. For each unit, we perform the data hiding according to the following four cases.

Case #1: $LSB(x_i) = m_i$ & $f(x_i,x_{i+1}) = m_{i+1}$
$(x'_i,x'_{i+1}) = (x_i,x_{i+1})$;

Case #2: $LSB(x_i) = m_i$ & $f(x_i,x_{i+1}) \neq m_{i+1}$
$(x'_i,x'_{i+1}) = (x_i,x_{i+1}+r)$;

Case #3: $LSB(x_i) \neq m_i$ & $f(x_i -1,x_{i+1}) = m_{i+1}$
$(x'_i,x'_{i+1}) = (x_i-1,x_{i+1})$;

Case #4: $LSB(x_i) \neq m_i$ & $f(x_i,x_{i+1}) \neq m_{i+1}$
$(x'_i,x'_{i+1}) = (x_i+1,x_{i+1})$;

Where m_i and m_{i+1} denote two secret bits to be embedded. The function is defined as f(a,b)= LSB([a/2] + b).r is a random value in {-1, +1} and (x'_i,x'_{i+1}) denotes the pixel pair after data hiding.

Step 4: After data hiding, the resulting image is divided into non-overlapping B_Z X B_Z blocks. The blocks are then rotated by a random number of degrees based on key. The process is very similar to Step 1 except that the random degrees are opposite. Then we embed the two parameters (t, B_Z) into a preset region which has not been used for data hiding.

There are two parameters in the proposed approach. The first one is the block size B_Z for block dividing in data preprocessing, another is the threshold t for embedding region selection. B_Z is randomly selected from the set of{1,4,8,12}, t belongs to {0,1,...,31} and can be determined by the image contents and the secret message M.

Here, an example is shown. Assume that we are dealing with an embedding unit $(x_i,x_{i+1}) = (62,81)$, $m_i = 1$, $m_{i+1} = 0$, T = 19. It is easy to verify that $|x_{i+1}-x_i| = 19 \geq T$ and
$LSB(62) = 0 \neq m_i$, $LSB([(62)/2]+81) = 1 \neq m_{i+1}$

Therefore, we invoke Case #4 and obtain
$(x'_i,x'_{i+1}) = (x_i+1,x_{i+1}) = (63,81)$.

3.1.2. Algorithm for Decoding

Step 1: To extract data, we first extract the side information, i.e., the block size B_Z and the threshold t from the stego image. Then do exactly the same things as Step 1 in data embedding.

Step 2: The stego image is divided into Bz X Bz blocks and the blocks are then rotated by random degrees based on the secret key key_1 .The resulting image is rearranged as a row vector V. Finally, the embedding unit is obtained by dividing V into non overlapping blocks with two consecutive pixels.

Step 3: Travel the embedding units whose absolute differences are greater than or equal to the threshold T according to pseudorandom order based on the secret key key_2, until all the hidden bits are extracted completely.

Step 4: Data extraction, for each qualified embedding unit, say(x'_i,x'_{i+1}) , where $x'_{i+1}-x'_i| \geq T$, we extract the two secret bits m_i,m_{i+1}as follows:

$m_i = LSB(x''_i)$, $m_{i+1} = LSB([x'_i/2]+x'_{i+1})$.

For instance, we are dealing with the unit $(x'_i,x'_{i+1}) = (63,83)$ with T = 19. We eventually get the secret bits by $m_i = LSB(63) = 1$, $m_{i+1} = LSB([63/2]+83) = 0$.

IV. Results And Performance Evaluation

Any Steganography technique is characterized mainly by two attributes, imperceptibility and capacity. Imperceptibility means the embedded data must be imperceptible to the observer (perceptual invisibility) and computer analysis (statistical invisibility). The performance of the proposed technique is evaluated using video stream rhinos.avi. The perceptual imperceptibility of the embedded data is indicated by comparing the original image or video to its stego counterpart so that their visual differences, if any, can be determined. Additionally, as an objective measure, the Mean squared Error (MSE) and Peak Signal to Noise Ratio (PSNR) between the stego frame and its corresponding cover frame are analysed.

The quantities are given as below,

$$MSE = \frac{1}{m \times n} \sum_{i=1}^{m-1} \sum_{j-1}^{n-1} (I - I') \text{^} 2$$

where, MSE is Mean Squared Error, m and n are height and width, I(i, j) represents original frame and I'(i, j) represents corresponding stego frame.

PSNR = 10 * log 10 (MAX^2 / MSE)

where, PSNR is Peak Signal to Noise Ratio, MAX is peak signal level for a grey scale image it is taken as 255. The cover file video details are given in Table 1 and results are tabulated in Table 2.

Table 1. Cover Video File details

S. No	Cover video file information			
	Name of Resolution (W1 *H1)	**Video file Resolution (W*H)**	**Frame/sec.**	**No .of frames**
1	rhinos.avi	320 * 240	15	105

Table 2. Results obtained from LSB and LSBMR techniques

Name of the Video file	No. of bits embedded	Results obtained using LSB		Results obtained using LSBMR	
		MSE	PSNR	MSE	PSNR
rhinos.avi	136	0.00082	79	0.00065	80
	200	0.00013	77	0.00010	78
	264	0.00016	76	0.00013	77
	312	0.00019	75	0.00016	76
	392	0.00024	74	0.00020	75
	552	0.00033	73	0.00028	74

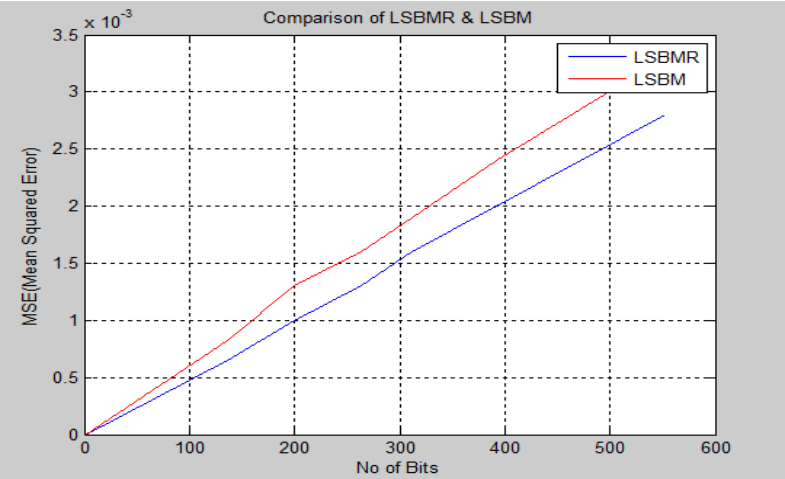

Fig 3 Comparison of LSB and LSBMR based on MSE

Fig 3. compares the MSE values of LSB algorithm with the LSBMR algorithm. In the LSB algorithm due to high replacement rate, MSE value is high. So it lacks from security**.** In case of LSBMR algorithm due to low replacement rate, MSE value is low which makes it secure when compared to LSB algorithm. In our proposed approach, intruder may not be able to identify the presence of the secret message inside the frame. Also, the comparison with the original video never gives the original secret message. This in turn provides additional security.

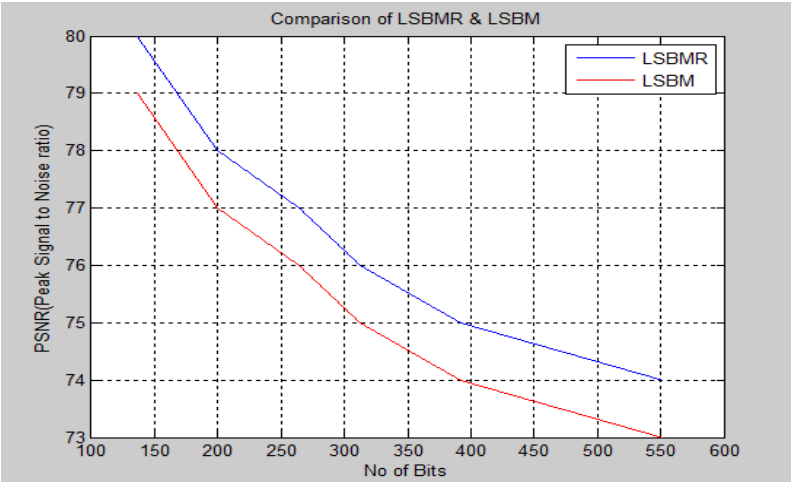

Fig 4 Comparison of LSB and LSBMR based on PSNR

Fig 4. compares the PSNR values of LSB algorithm with the LSBMR algorithm Peak Signal to Noise Ratio are plotted in the graph. When the embedding unit increases, PSNR value decreases.

V. Conclusion And Future Enhancement

The proposed scheme addresses two problems that were identified in the existing approach which were Lack of Security and Low Embedding rate. Embedding text in video is more secure when compared to an image. It is due to the fact that the intruder may not be able to identify the presence of the secret message inside the frame. Even the comparison with the original video never gives the original secret message. This in turn provides additional security. In future, it is expected that the idea can be extended by embedding the text in the different frames of same video. Since the video consists of many number of frames, the text can be embedded in many methods like embedding in the consecutive frames based on the key, in the frames with the sequence number of multiples of key, in the frames with the sequence number of powers of key, etc., thereby providing a technique to embed large amounts of data with additional security and making it difficult for the steganalysers to detect the secret data.

References

[1]. E. Cole and R.D. Krutz, Hiding in Plain Sight: Steganography and the Art of Covert Communication, Wiley Publishing, Inc., ISBN 0-471-44449-9, 2003.

[2]. Katzenbeisser and Fabien A. P. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking, Artech House Books, ISBN 1-58053-035-4, 1999.

[3]. D. Stanescu, M. Stratulat, B. Ciubotaru, D. Chiciudean, R. Cioarga and M. Micea, Embedding Data in Video Stream using Steganography, in 4th International Symposium on Applied Computational Intelligence and Informatics, SACI-2001, pp. 241-244, IEEE, 2007.

[4]. Feng Pan, Li Xiang, Xiao-Yuan Yang and Yao Guo, Video Steganography using Motion Vector And Linear Block Codes, in IEEE 978-1-4244-6055-7/10/, pp. 592-595,2010.

[5]. A. Westfield, and A. Pfitzmann, Attacks on Steganographic Systems, in Proceedings of 3rd Info.Hiding Workshop, Dresden, Germany, Sept. 28−Oct. 1, pp. 61-75, 1999.

[6]. J. Fridrich, R. Du, and L, Meng, Steganalysis of LSB Encoding in Color Images, in Proceedings of ICME 2000, Jul.-Aug. 2000, N.Y., USA.

[7]. Fillatre. L, Designing of Robust Image Steganography Technique Based on LSB Insertion and Encryption, IEEE Transactions on Signal Processing, Volume 60, Issue:2, pp. 556-569, Feb, 2012

[8]. Masud K. S.M. Rahman, Hossain, M.L., A new approach for LSB based image steganography using secret key, in Proceedings of 14th International Conference on Computer and Information Technology (ICCIT-2011), pp.-286-291, Dec. 2011.

[9]. Hema Ajetrao, Dr. P.J.Kulkarni and Navanath Gaikwad, A Novel Scheme of Data Hiding in Binary Images, in International Conference on Computational Intelligence and Multimedia Applications, Vol.4, pp. 70-77, Dec. 2007.

[10]. Sachdeva S. and Kumar A, Colour Image Steganography Based on Modified Quantization Table,in Proceedings of Second International Conference on Advanced Computing & Communication Technologies (ACCT-2012), pp. 309-313, 2012.

[11]. R. Machado, http://www.securityfocus.com/tools/586/scoreit, .EzStego., Nov. 1996. [last accessed on 16-04-2012]

[12]. Y. C Tseng and H. K Pan, Data Hiding in 2-color Image, in IEEE Transactions on computers,Vol. 51, No. 7, pp. 873-878, July 2002.

[13]. E. Kawaguchi and R. O. Eason, Principle and applications of BPCS-Steganography, in Proceedings of SPIE Int'l Symp. on Voice, Video, and Data Communications, pp. 464-473, 1998.

[14]. Mritha Ramalingam, Stego Machine Video Steganography using Modified LSB Algorithm, in World Academy of Science, Engineering and Technology 74 2011, pp. 502-505, 2011.

[15]. Juan Jose Roque and Jesus Maria Minguet, SLSB: Improving the Steganographic Algorithm LSB, in the 7th International Workshop on Security in Information Systems (WOSIS 2009),Milan, Italy, pp.1-11, 2009.

[16]. A.K. Bhaumik, M. Choi, R.J. Robles and M.O. Balitanas, Data Hiding in Video ,in International Journal of Database Theory and Application Vol. 2, No. 2, pp. 9-16, June 2009.

[17]. J. J. Chae, B. S. Manjunath, Data Hiding in Video, Proceedings of the 6th IEEE International Conference on Image Processing, pp.311-315, 1999.

[18]. Melih Pazarci, Vadi Dipcin, Data Embedding in Scrambled Digital Video, in Proceedings of the 8th IEEE International Symposium on Computers and Communication, pp. 498-503, 2003.

[19]. A. Giannoula, D. Hatzinakos, "Compressive Data Hiding for Video Signals", in Proceedings of International Conference on Image Processing, pp. I529- I532, 2003.

[20]. Giuseppe Caccia, Rosa Lancini, Data Hiding in MPEG2 Bit Stream Domain, in Proceedings of International Conference on Trends in Communications, pp.363-364, 2001.

[21]. Jun Zhang, Jiegu Li, Ling Zhang, Video Watermark Technique in Motion Vector, in Proceedings of XIV Brazilian Symposium on Computer Graphics and Image Processing, pp.179-182, 2001.

[22]. Feng Pan, Li Xiang, Xiao-Yuan Yang and Yao Guo, Video steganography using motion vector and linear block codes, in Proceedings of IEEE International Conference on Software Engineering and Service Sciences (ICSESS- 20100), pp. 592-595, 2010.