Hiding Data using Motion Vector and Linear Block Codes in Video Steganography

Rucha Bahirat[#1], Amit Kolhe[*2]

[#]Department of Electronics & Telecommunication,

CSVTU
RCET Bhilai, Chhattisgarh. INDIA

[1]rucha.bahirat@gmail.com

[2]amitkolhe@sify.com

***Abstract*— A video steganography scheme based on motion vector and linear block codes has been proposed in this paper. In our proposed method we are embedding the secret message into the video signal during the process of video compression. In the sender side during the H.264 compression process, secret data is embedded in the motion vector using a new approach named linear block parity coding. In this approach, instead of original message an encrypted message by data matrix coding algorithm is hidden into a video. At the receiver side the secret message can be extracted directly without using original video sequence. This method provides more security than conventional approaches. The computational complexity will comparatively low with other methods. The proposed method is analyzed in terms of Peak Signal to Noise Ratio (PSNR).**

***Keywords*— Video steganography, motion vectors, H.264, linear block codes, parity coding.**

I. INTRODUCTION

Information attacks are showing the weakness of information security due to the rapid growth of globalization. The main aim of these attacks is to retrieve the information by illegal that shows the fault in the security system. Steganography is the art and science of writing hidden messages in such a way that no body, except from the transmitter and intended receiver, suspects the existence of the message [1-2]. The secrete message is hidden into the cover media. The cover media may be any digital media, an image file, an audio or any video file. The word steganography is of Greek origin and means "concealed writing" from the Greek words steganos (στεγανός) meaning "covered or protected", and graphei (γραφή) meaning "writing". There are two steps in steganography, they are embedding and extraction. In embedding process transmitter hide the secrete message into cover media using a secrete key. In extraction process the receiver uses the same secrete key which is used by the transmitter to extract the secrete message from the stego file.

In last few years, video steganography techniques are nearly related with the video compression methods. Motion vectors are calculated by those compression standards to remove the redundancies between successive frames. There are different methods that have been proposed for hiding message into digital videos. There are many method based on motion vectors have been proposed for MPEG algorithms [3-4]. Zhang employed the characteristics of MPEG codes to embed information into INTER frame [5]. Zhou et al proposed a video watermarking technology to hide information by least modification of motion vector in a compressed video bitstream [6]. All the above methods did not take the full advantages of the motion vector. In this paper an advance method has been proposed.

The rest of the paper is arranged as follows. In section II we describe the new video steganographic scheme i.e. linear block code parity coding and data matrix coding. Section III gives the experimental results of the algorithms. In section IV, a conclusion is drawn finally.

II. THE PROPOSED STEGANOGRAPHIC METHOD

A. Linear Block Codes

Linear block codes have the property of linearity, i.e. the addition of any two codewords is also a codeword, and they are applied to the source bits in blocks, hence the name linear blocks codes. In block code there are k bits or symbols are input and n bits or symbols are output. We allocate the code as an ($n; k$) code. We are using bits, fundamentals from the field *GF (2).* If we select k input bits, then there are 2^k different messages. A block code C which have length n with 2^k codeword is called a linear ($n; k$) code only if its 2^k code words form a k-dimensional subspace of the vector space of all n-tuples over the field *GF (2).* In general, with a larger field, a block code C of length n with q^k code words is called a linear ($n; k$) code if and only if its q^k codeword form a k-dimensional subspace of the vector space of all n-tuples over the field *GF (q).*

The (n; k) linear block code contains 2^k columns and 2^{n-k} rows in a standard array. No vectors are identical in the same row, each vector comes out just once in the standard array.

For a given (n; k) linear code C, the standard array is made as follows.

A. Starting with the codeword 0, write down all the 2^k codewords of C as a row.
B. Select one of the vector e1 of minimum weight from the remaining vectors of GF (2) in a sequence, which we will call a coset leader. Under the vector 0 in the standard array Add e1, and complete the columns by adding e1 to the codewords of C.

C. Select a sequence of minimum weight e2 from the remaining vectors of GF (2). Add weight e2 to the vector e1 in the standard array, and fill the columns by summing e2 to the codewords of C.

D. Repeat this procedure until all vectors of GF (2) haven been placed in the array.

The following properties have to be satisfied by the standard array:-

- There are 2^k columns and 2^{n-k} rows; therefore, an (n; k) block code is able to correcting 2^{n-k} error patterns (the coset leaders).
- The difference of any two vectors in the same row is a codeword in C, all vectors in the same row have the same syndrome.
- Two vectors in the same row are not the same.
- Each vector comes out only one time in the standard array.

Standard array for a code *C* is a look-up table for all cosets of *C* and the first column has a minimum weight vector for each coset. These minimum weight vectors are called coset leaders of *C* for a given standard array. Coset leaders can be acted as the correctable error patterns when individual use the respective standard array to decode the received vectors. The syndrome *s* of a coset is only depend on the error pattern *e* and is independent of the transmitted codeword.

B. Data matrix coding

A Data matrix code is a two-dimensional matrix barcode consisting of black and white "cells" or modules arranged in either a rectangular or square pattern. The message to be encoded may be numeric or text information. Normal data size is from a a small number of bytes to 1556 bytes. Depending on the number of cells in the matrix we determine the length of the encoded data. Reliability can be increased by the Error correction codes: even if one or more cells is damaged so it is illegible, the data can still be read. A Data Matrix symbol can accumulate up to 2,335 alphanumeric characters.

Data Matrix symbols are rectangular in shape and usually square and are composed of "cells": little squares that represent bits. Depending on the coding used, a "light" cell represents a 0 and a "black" cell is a 1, or vice versa. All Data Matrix is contains two solid adjacent borders in an "L" shape (called the "finder pattern") and two other borders consisting of alternating dark and light "cells" or modules (called the "timing pattern"). Inside these limits are columns and rows of cells encoding data. To locate and orient the symbol the finder pattern is used while the timing pattern provides a count of the number of rows and columns in the symbol. The number of cells (rows and columns) increases, as more data is encoded in the symbol. Each code is unique. Symbol sizes vary from 10×10 to 144×144 in the new version ECC 200, and from 9×9 to 49×49 in the old version ECC 000 - 140.

B. Embedding-

1. Discard the surrounding macro-block in the current inter frame.
2. Compute the motion vector magnitudes in the remaining macro-blocks.
3. Using the predefine threshold T to select a set S for candidate motion vectors (MV).
4. Compute the phase angle of each motion vector in S.
5. Segment the circle into eight sectors, and if the motion vector phase angle is belong to the shadow place, the motion vector represent data 0, otherwise, the motion vector represent data 1.
6. The N motion vectors are grouped together in n, denoted by r.
7. The secret information are grouped together in (n-k), denoted by a, and encrypt it using Data Matrix coding.
8. Compute the syndrome s of r, it is a combination of motion vector r, and Hybrid code (Linear block Code & parity coding) consistent monitoring matrix.
9. Compute b from s and a, a look-up table which describing the relationship between coset leaders and syndromes, searching for the coset leader corresponding with b, assume to be e_b
10. Compute the public information after embedding secret data, from r ande_b.
11. The secret key is the Hybrid code (Linear block Code & parity coding) consistent monitoring matrix H and the method of segmentation and representation.

C. Extraction-

The receiver gets the steganography video and recovers the secret information using the secret key. The extraction process is as follows:

1. The steps 1-6 is the same with the embedding process, thus the receiver gets r'.
2. Computes', decrypt it using Linear Block coding to make s', s' is the secret information which the receiver wants to extract from the steganography video.

D. Hybrid Linear block Code & parity coding –

In the second layer of data security, secret text is embedded into an image using Linear Block Parity coding method. In this method, the image is the combination of Black and white pixel where black pixel is represented as '0' and white pixel is as '1'. It means, only one bit representation is possible for each pixel. In building block parity data hiding method, first, considered an original image (I) represented by a matrix. It is partitioned into m x n blocks. To control the image quality after data hiding should be considered the following algorithm. [7].

E. Data Hiding Algorithm

The original image 'I' is partitioned into m × n blocks. For simplicity, we assume that size of 'I' is multiple of m × n. The data hiding is achieved by modifying some bits of 'I'. Below we show how to hide one bit of original information into m× n host block, say 'ai'.

Step 1: Find the parity of 'I'.
Step 1.1: If ((Sum (Ii) mod 2) = =0)
Then "Even Parity".
Step 1.2: If ((Sum (Ii) mod 2) ≠0)
Then "Odd Parity".

Step 2: If ((Parity (I) ==”Even”) and (ai = = 1)) || ((Parity (I) = =”Odd”) and (ai = =0)) then go to step 4 otherwise “No change”.
Step 3: Find Neighbour (I).
Step 4: Find the location of highest value from Neighbour (I) to hide the data.
Step 5: Complement the bit in the position which we found in the step 4.

Embed the cipher text using the embedding algorithm [8]. The resultant image is called as stego image which is participating in the communication. The results are discussed in the following section.

III. EXPERIMENT RESULTS

To test the performance of the algorithms, we uses two video sequences have been, they are Foreman (352x288, 300 Frames), Flower (352x288, 250 Frames), Stefan (352x288,90) and mobile(352x288,300). In the experiment, our proposed method is applied to the H.264 compressing process.

A. *Embedding Capacity*

In the experiment, we choose threshold value *T=10* and divides circle into eight segments. In this algorithm the *(n,k)* linear block codes can embed n-k bits per n bits, and the variation is at most 2 bits. So the steganographic algorithm proposed by this paper can embedding n-k bits secret messages per n motion vectors, the embedding capacity of our method is nearly take up 2/3 of the total numbers of motion vectors.

B. *Message Encryption*

In this method we are choosing secrete message “TEXT” and then applying Data Matrix Coding. The message after encryption is shown in Fig. 1.

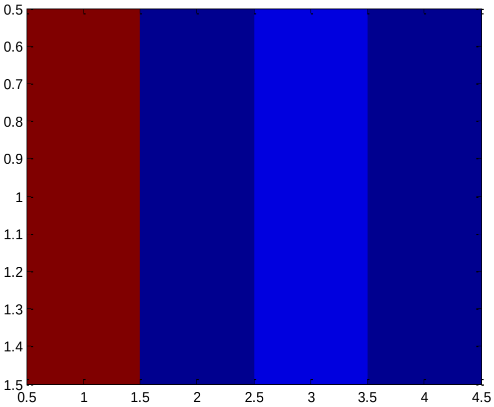

Fig. 1.Secrete Encrypted Message

C. *Visual Imperceptible*

In virtue of our method has a lower motion vector modification rate, so the quality of the cover media has not greatly declined after embedding secret messages during the process of H.264 compressing. In the experiment, we are calculating the mean Peak Signal-to-Noise Ratio *(PSNR)* of the video sequence and the experiment results are shown in table I.

Generally speaking, if the value of PSNR is more than 30dB, then people are difficulty to notice the difference between the cover image and stego-image. In TABLE I, the mean PSNR of Flower and Foreman video sequence are all more than 30dB, thus the changes caused by embedding is so small that people can not sensitive of the difference. Therefore, our method can maintain good video quality and can get a satisfied performance as well.

TABLE I
THE MEAN PSNR & MSE

Cover Video File Name	**No of frames**	**Secrete Text Message**	**Average PSNR**
Flower	250	Text	39.95
Foreman	300	Text	35.53
Mobile	300	Text	36.34
Stefan	90	Text	36.24

IV. CONCLUSION

In this paper, a new steganographic method using motion vectors and linear block parity codes was proposed. We embedding *(n-k)* bits in per *n* motion vectors and only the weight of e_b motion vectors are modified at most. In this way, our algorithm is improved the embedding efficienfy. Experimental results show that our proposed scheme can embed large amounts of information and can maintain good video quality as well. The large embedding capacity and visual imperceptible after embedding process made our scheme can satisfy the request of covert communications.

REFERENCES

1) Min Wu and Bede Liu, “Data hiding in image and video. I. Fundamental issues and solutions,” IEEE Trans. Image Processing, vol. 12, no. 6, pp.685-695, June 2003.
2) Min Wu and Bede Liu, “Data hiding in image and video .II. Designs and applications,” IEEE Trans. Image Processing, vol. 12, no. 6, pp.685-695, June 2003.
3) Kutter M., Jordan F. and Ebrahimi T.: Proposal of a watermarking technique for hiding/retriving data in compressed and decompressed video, Technical reportM2881, ISO/IEC document,JTC1/Sc29/WG11,1997.
4) Ding Y F,Long W C.Data Hiding for Digital Video with Phase of Motion Vector.In:Proc Int Symp on Circuits and Systems.Kosisland, 2006, pp.1422-1425.
5) J.Zhang, H. Maitre, L. Ling, “Embedding watermark in MPEG video sequence,” *IEEE Fourth Workshop on Multimedia Signal Processing*, pp. 535-540, oct. 2001.
6) Z. Zhuo, Y. Nenghai and L. Xuelong, “ A novel video watermarking scheme in compression domain based on fast motion estimation” in Proc. ICCT 2003, vol. 2, pp. 1878-1882, Apr.2003.
7) F. A. P. Petitcolas, R. J. Anderson, M. G. Kuhn (1999) “Information hiding – survey”, Proc. IEEE, vol. 87, No. 7, pp. 1062 – 1078.

8) Yueyun Shang, "A New Invertible Data Hiding in Compressed Videos or Images", Third International Conference on Natural Computation (ICNC 2007), Vol. 4, pp. 576-580, Haikou, Aug. 2007.

9) Yueyun Shang, "A New Invertible Data Hiding in Compressed Videos or Images", Third International Conference on Natural Computation (ICNC 2007), Vol. 4, pp. 576-580, Haikou, Aug. 2007.

10) Venkatraman S., Ajith Abraham and Marcin Paprzycki, “Significance of Steganography on Data Security”, International Conference on Information Technology:Coding and Computing (ITCC’04), Vol. 2, April 2004.