Video Files Using Flexible Macro Block Ordering

Anitha.T, Ganesh Shankar.S

Abstract— This paper proposes two data hiding approaches using compressed MPEG video. The first approach hides message bits by modulating the quantization scale of a constant bitrates video. A payload of one message bit per macro block is achieved. A second order multivariate regression is used to find an association between macro block-level feature variables and the values of a hidden message bit. The regression model is then used by the decoder to predict the values of the hidden message bits with very high prediction accuracy. The second approach uses the flexible macro block ordering feature of H.264/AVC to hide message bits. Macro blocks are assigned to arbitrary slice groups according to the content of the message bits to be hidden. A maximum payload of three message bits per macro block is achieved. The proposed solutions are analyzed in terms of message extraction accuracy, message payload, excessive bitrates and quality distortion. Comparisons with previous work reveal that the proposed solutions are superior in terms of message payload while causing less distortion and compression overhead

Index Terms— Data hiding,flexible macro block ordering,MPEG coding,multi variant regression,packet loss.

IJSER

1 Introduction

DATA hiding techniques can be used to embed a secret message into a compressed video bit stream for copyright protection, access control and transaction tracking. Some data hiding techniques to assess the quality of compressed video in the absence of the original reference. The quality is estimated based on computing the degradations of the extracted hidden message. Data hiding is also used for error detection and concealment in applications of video transmission. Edge orientation information and number of bits of a block are hidden in the bit stream for that purpose. Data hiding techniques can be used to embed a secret message into a compressed video bit stream for copyright protection, access control and transaction tracking. Some data hiding techniques to assess the quality of compressed video in the absence Of the original reference. The existing solution rely on hiding message bits in discrete cosine transform (DCT) coefficients, motion vectors(MV), quantization scale or prediction modes. The first approach the quantization scale of a CBR video is either incremented and decremented according to approaches using compressed MPEG Video. A second-order multivariate regression is used to associate macro block level features with the hidden message bit.

2 Some system analysis

The existing solutions rely on hiding message bits in discrete cosine transform (DCT) coefficients, motion vectors (MV), quantization scale or prediction modes. Examples of data hiding using DCT coefficients include the use the parity of the quantized coefficients to hide a message. They utilized zero-length codes to insert a dummy value at certain locations to indicate message bits. In MV technique, the phase angles of MV are used to hide messages. This paper proposes two data hiding approaches using compressed MPEG video. The first approach hides message bits by modulating the quantization scale of a constant bit rate video. A payload of one message bit per macro block is achieved. The regression model is then used by the decoder to predict the values of the hidden message bits with very high prediction accuracy. The second approach uses the flexible macro block ordering feature of H.264/AVC to hide message bits. Macro blocks are assigned to arbitrary slice groups according to the content of the message bits to be hidden. A maximum payload of three message bits per macro block is achieved. The proposed solutions are analyzed in terms of message extraction accuracy, message payload, and excessive bit rate and quality distortion. Comparisons with previous work reveal that the proposed solutions are superior in terms of message payload while causing less distortion and compression overhead. The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process it lists the requirements of a particular software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective. The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its user interface, hardware and software requirements. It defines how the client, team and audience see the project and its functionality. The proposed system has the advantage of the high message payload, less video distortion and excessive overhead. The methods used in proposed system are very efficient in hiding data and extracting the data. Apart from the advantage of increase message payload, excessive bit rate and quality distortion the proposed solution overcome the packet loss in MPEG Video.

- *Anitha.T is currently working as Assistant Professor in Information Technology in Sree Sastha Institute of Engineering and Technology,Chennai, India,*
  E-mail: anitha.thiyagarajan@yahoo.com.
- *Ganesh Shankar.S is currently working as Assistant Professor in Information Technology in Sree Sastha Institute of Engineering and Technology, Chennai, India, E-mail: sganeshshanker@gmail.com*

3 OTHER IMPLEMENTATION

3.1 Frame Selection

Data hiding techniques can be used to embed a secret message into a compressed video bit stream for copyright protection, access control and transaction tracking. Some data hiding techniques to assess the quality of compressed video in the absence of the original reference. The quality is estimated based on computing the degradations of the extracted hidden message. Data hiding is also used for error detection and concealment in applications of video transmission. Edge orientation information and number of bits of a block are hidden in the bit stream for that purpose. In this project our first module is select the frame from using the given input video. First of the video is split the audio and video separately. Thereafter the video part will be converting into the N number of frames. In future these frames are used to block selection process.

3.2 Data Hiding using Quantization Technique

To hide a message using quantization scale modulation, the message is first converted into a binary stream of bits. During the MPEG encoding of individual macro blocks, the message bits are read one at a time. For each coded macro block, the quantization scale is either incremented or decremented based on the corresponding message bit. Clearly, if the original quantization scale was either the lowest or largest allowable values then no modification is applied. By modulating the quantization scale of a constant bit rate video a payload of one message bit per macro block is achieved.

3.3 Data Hiding using FMO Technique

One of the limitations of the quantization scale modulation solution of the previous section is related to the message payload where only one message bit can be hidden per macro block. This section introduces a second solution that benefits from a higher message bit rate through the use of FMO. A coded picture is divided into one or more slices. This feature is important to suppress error propagation within a picture due to the nature of variable length coding. Each slice group contains one or more slices and macro blocks can be assigned in any order to these slices. The assignment of macro blocks to different groups is signaled by a syntax structure called the “slice group id”. To hide a message into the H.264/AVC bit stream, the message is first read into chunks of bits. If macro blocks are coded per picture, then message bits can be used to allocate the macro blocks to slice groups. By using this technique a maximum payload of three message bits per macro block is achieved.

3.4 Extracting the Data

In Quantization Technique, the question that remains is how to extract the message from the bit stream. This problem can be solved by extracting macro block-level feature variables during the encoding process. Once the whole message is hidden we end up with a feature matrix and a message vector. We will then treat the feature matrix as predictors and the message bits as a response variable and use multivariate regression to compute a prediction model. Once computed, the prediction model can be used to predict the message bit hidden in a given macro block based on its feature variables. In FMO technique, to extract the message bits, each time a picture is decoded, the macro block to slice group mapping syntax structure is used to read message bits and append them to the extracted message.

4 ARCHITECTURAL OVERVIEW

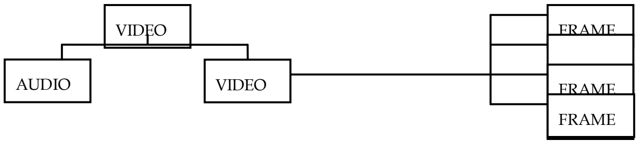


Fig. 1. Video divided into frames

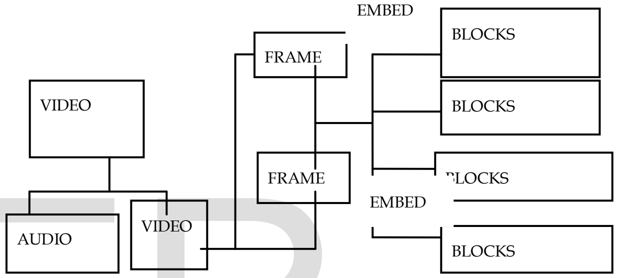


Fig. 2. Video frames are divided into blocks

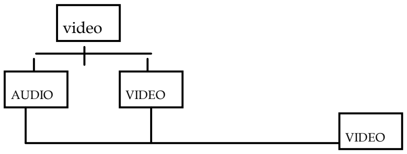


Fig..3. Video frames are merging into videos

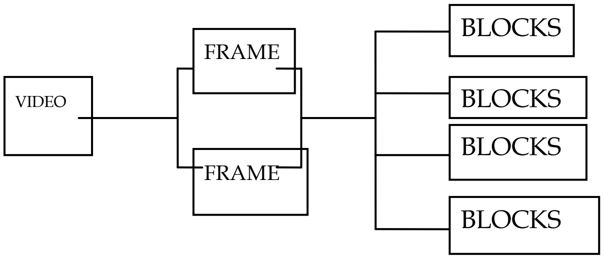


Fig..4. Frames are Extracted into blocks

IJSER

5 Other Coding and Testing

5.1 Coding Standard

Once the design aspect of the system is finalizes the system enters into the coding and testing phase. The coding phase brings the actual system into action by converting the design of the system into the code in a given programming language. Therefore, a good coding style has to be taken whenever changes are required it easily screwed into the system.

Coding standards are guidelines to programming that focuses on the physical structure and appearance of the program. They make the code easier to read, understand and maintain. This phase of the system actually implements the blueprint developed during the design phase. The coding specification should be in such a way that any programmer must be able to understand the code and can bring about changes whenever felt necessary. Some of the standard needed to achieve the above-mentioned objectives are as follows:

Program should be simple, clear and easy to understand.
Naming conventions
Value conventions
Script and comment procedure
Message box format
Exception and error handling

5.2 Naming Conventions

Naming conventions of classes, data member, member functions, procedures etc., should be self-descriptive. One should even get the meaning and scope of the variable by its name. The conventions are adopted for easy understanding of the intended message by the user. So it is customary to follow the conventions.

5.3 Member Function and Data member name

Member function and data member name begins with a lowercase letter with each subsequent letters of the new words in uppercase and the rest of letters in lowercase.

5.4 Value Convention

Value conventions ensure values for variable at any point of time. This involves the following:

Proper default values for the variables.
Proper documentation of flag values.

5.5 System Testing

Testing is performed to identify errors. It is used for quality assurance. Testing is an integral part of the entire development and maintenance process. The goal of the testing during phase is to verify that the specification has been accurately and completely incorporated into the design, as well as to ensure the correctness of the design itself. For example the design must not have any logic faults in the design is detected before coding commences, otherwise the cost of fixing the faults will be considerably higher as reflected. Detection of design faults can be achieved by means of inspection as well as walkthrough.

Testing is one of the important steps in the software development phase. Testing checks for the errors, as a whole of the project testing involves the following test cases:

- Static analysis is used to investigate the structural properties of the Source code.
- Dynamic testing is used to investigate the behavior of the source code by executing the program on the test data.

5.6 Unit Testing

Unit testing is conducted to verify the functional performance of each modular component of the software. Unit testing focuses on the smallest unit of the software design (i.e.), the module. The white-box testing techniques were heavily employed for unit testing.

5.7 Functional Testing

Functional test cases involved exercising the code with nominal input values for which the expected results are known, as well as boundary values and special values, such as logically related inputs, files of identical elements, and empty files.

Three types of tests in Functional test:

- Performance Test
- Stress Test
- Structure Test

5.8 Security Testing

Security testing attempts to verify the protection mechanisms built in to a system well, in fact, protect it from improper penetration. The system security must be tested for invulnerability from frontal attack must also be tested for invulnerability from rear attack. During security, the tester places the role of individual who desires to penetrate system.

5.9 White box Testing

This testing is also called as Glass box testing. In this testing, by knowing the specific functions that a product has been design to perform test can be conducted that demonstrate each function is fully operational at the same time searching for errors in each function. It is a test case design method that uses the control structure of the procedural design to derive test cases. Basis path testing is a white box testing.

Basis path testing:

- Flow graph notation
- Cyclometric complexity
- Deriving test cases
- Graph matrices Control

5.10 Black box Testing

In this testing by knowing the internal operation of a product, test can be conducted to ensure that “all gears mesh”, that is the internal operation performs according to specification and all internal components have been adequately exercised. It fundamentally focuses on the functional requirements of the software.

The steps involved in black box test case design are:

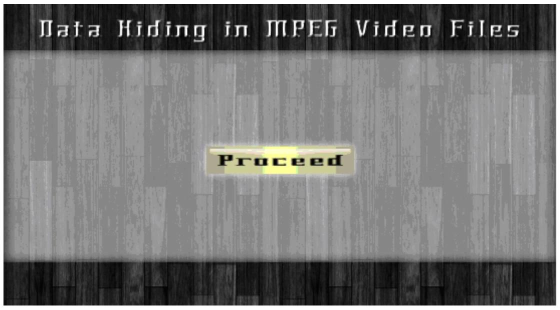
- Graph based testing methods

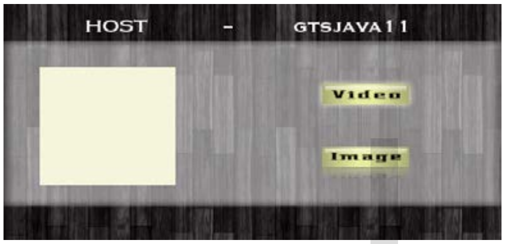
IJSER

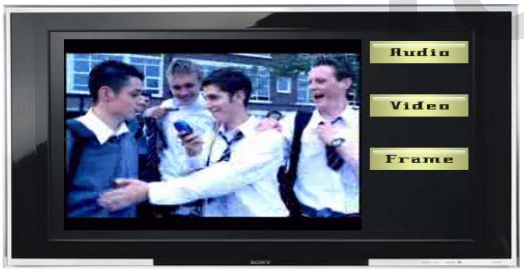
- Equivalence partitioning
- Boundary value analysis
- Comparison testing

6 Some Modules

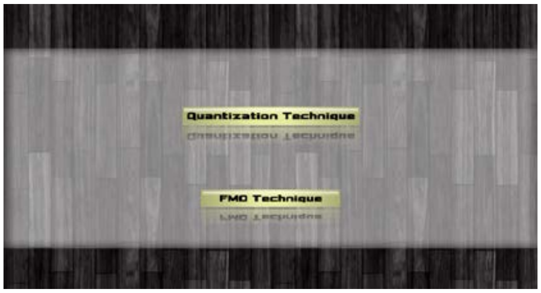
6.1 Module-1

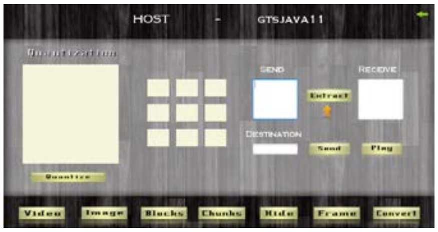


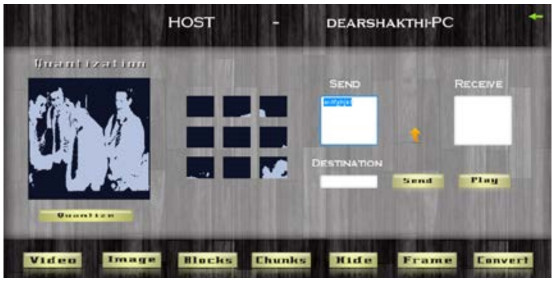




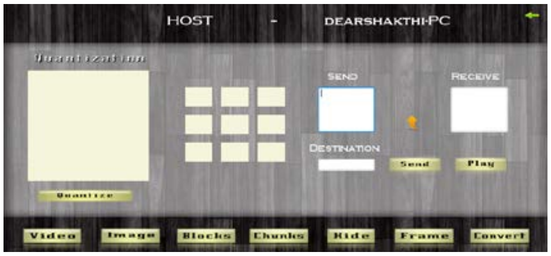
6.2 Module-2

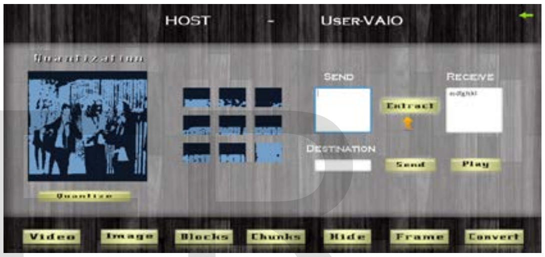






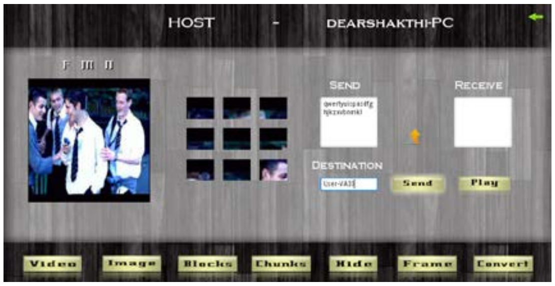
6.3 Module-3





IJSER

6.4 Module-4



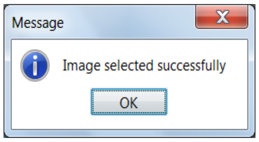
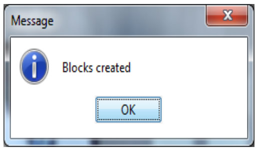
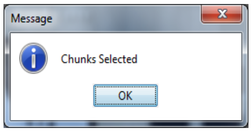

Message

Image selected successfully


Message

Blocks created


Message

Chunks Selected

7 Conclusion

In the existing system there is chance of getting in-ordered and loss of message. Also the quality of the video decreases as it buffers. In the proposed system by using FMO and Multivariate regression we overcome the above mentioned problems.

References

[1] M. Carli, M. Farais, E. D. Gelasca, R. Tedesco, and A. Neri, “Quality assessment using data hiding on perceptually important areas,” in Proc. IEEE Int. Conf. Image Processing, ICIP, Sep. 2005, pp. III-1200-3–III-1200-3.

[2] S. Kapotas and A. Skodras, “A new data hiding scheme for scene change detection in H.264 encoded video sequences,” in Proc. IEEE Int. Conf. Multimedia Expo ICME, Jun. 2008, pp. 277–280.

[3] A. Yilmaz and A. Aydin, “Error detection and concealment for video transmission using information hiding,” Signal Processing: Image Communication, vol. 23, no. 4, pp. 298–312, Apr. 2008.

[4] K. Nakajima, K. Tanaka, T. Matsuoka, and Y. Nakajima, “Rewritabledata embedding on MPEG coded data domain,” in Proc. IEEE Int.Conf. Multimedia and Expo, ICME, Jul. 2005, pp. 682–685.

[5] C. Xu, X. Ping, and T. Zhang, “Steganography in compressed videostream,” in Proc. Int. Conf. Innovative Computing, Information and Control, ICICIC’06, 2006, vol. II, pp. 803–806.

[6] H. A. Aly, “Data hiding in motion vectors of compressed video basedon their associated prediction error,” IEEE Trans. Inform. Forensics Security, vol. 6, no. 1, pp. 14–18, Mar. 2011.

[7] K. Wong, K. Tanaka, K. Takagi, and Y.Nakajima, “Complete video quality-preserving data hiding,” IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 10, Oct. 2009.

[8] Y. Hu, C. Zhang, and Y. Su, “Information hiding based on intra prediction modes H.264/AVC,” in Proc. IEEE Int. Conf. Multimedia and Expo, ICME, Jul. 2007, pp. 1231–1234.

[9] H. A. Aly, “Data hiding in motion vectors of compressed video based on their associated prediction error,” IEEE Trans. Inform. Forensics Security, vol. 6, no. 1, pp. 14–18, Mar. 2011.

[10] K. Wong, K. Tanaka, K. Takagi, and Y. Nakajima, “Complete video quality-preserving data hiding,” IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 10, Oct. 2009.

[11] Y. Hu, C. Zhang, and Y. Su, “Information hiding based on intra prediction modes H.264/AVC,” in Proc. IEEE Int. Conf. Multimedia and Expo, ICME, Jul. 2007, pp. 1231–1234.

[12] G. Yang, J. Li, Y. He, and Z. Kang, “An information hiding algorithm based on intra-prediction modes and matrix coding for H.264/AVC video stream,” Int. J. Electron. Commun., vol. 65, no. 4, pp. 331–337, Apr. 2011.

[13] Y. Hu, C. Zhang, and Y. Su, “Information hiding based on intra prediction modes H.264/AVC,” in Proc. IEEE Int. Conf. Multimedia and Expo, ICME, Jul. 2007, pp. 1231–1234.

[13] K. Solanki, U. Madhow, B. S. Manjunath, S. Chandrasekaran, and I.El-Khalil, “’Print and Scan’ resilient data hiding in images,” IEEE Trans. Inform. Forensics Security, vol. 1, no. 4, pp. 464–478, Dec.2006.

[14] X.-P. Zhang, K. Li, and X. Wang, “A novel look-up table design method for data hiding with reduced distortion,” IEEE Trans. Circuits Syst. Video Technol., vol. 8, no. 6, pp. 769–776, Jun. 2008.

[15] M. Xiaojing, L. Zhitang, T. Hao, and Z. Bochao, “A data hiding algorithm for H.264/AVC video streams without intra-frame distortion drift,” IEEE Trans. Circuits Syst. Video Technol., vol. 20, no. 10, pp.1320–1330, Oct. 2010.

[16] S. Lyu and H. Farid, “Steganalysis using higher-order image statistics,” IEEE Trans. Inform. Forensics Security, vol. 1, no. 1, pp. 111–119, Mar. 2006.

[17] U. Budhia, D. Kundur, and T. Zourntos, “Digital video steganalysis exploiting statistical visibility in the temporal domain,” IEEE Trans. Inform. Forensics Security, vol. 1, no. 4, pp. 502–516, Dec. 2006.

[18] Y. Li, H.-X. Chen, and Y. Zhao, “A new method of data hiding based on H.264 encoded video sequences,” in Proc. IEEE Int. Conf. Signal Processing, ICSP, Oct. 2010, pp. 1833–1836.

[19] K. Nakajima, K. Tanaka, T. Matsuoka, and Y. Nakajima, “Rewritable data embedding on MPEG coded data domain,” in Proc. IEEE Int. Conf.

[20] S. Lyu and H. Farid, “Steganalysis using higher-order image statistics,”IEEE Trans. Inform. Forensics Security, vol. 1, no. 1, pp. 111–119, Mar. 2006.

[21] K. Nakajima, K. Tanaka, T. Matsuoka, and Y. Nakajima, “Rewritabledata embedding on MPEG coded data domain,” in Proc. IEEE Int.Conf. Multimedia and Expo, ICME, Jul. 2005, pp. 682–685.

IJSER