

LOSSLESS VISIBLE WATERMARKING FOR COPYRIGHT PROTECTION OF H.263 VIDEOS*

¹Yi-Lin Wang (王彞琳)¹ and Wen-Hsiang Tsai (蔡文祥)^{1, 2}

¹Dept. of Computer Science, National Chiao Tung University, Hsinchu, Taiwan

²Dept. of Computer Science & Information Eng., Asia University, Taichung, Taiwan

Abstract

A lossless visible watermarking method is proposed for copyright protection of H.263 videos which are transmitted over mobile phones. The method can be employed to embed a given watermark image into an H.263 video, resulting in a watermarked video with a visible watermark in the video frames. Relevant watermarking information is also embedded in the video so that the original video can be recovered losslessly without the use of the original watermark in the recovery process. To prevent illegal removal of the visible watermark, a user-specified key is adopted for randomizing the content of the embedded watermark. Good experimental results show the feasibility of the proposed method.

Keywords: copyright protection, data hiding, H.263, lossless, mobile phone, watermarking.

1. INTRODUCTION

Nowadays, many techniques of data hiding in videos have been proposed [1-4]. The purpose is to achieve the functions of copyright protection, covert communication, information authentication, etc. Chen and Tsai [1] proposed a method to hide data in the median frequencies of the DCT domain in the I-frame and proper motion vectors in the P-frame and the B-frame of MPEG videos. A method of embedding information in H.263 videos was proposed by Song and Liu [3] in which information is embedded by changing the half-pixel motion estimation in every macroblock coded in the INTER mode.

With the advance of mobile computing, more and more mobile phones are provided with the capability of video recording. People use this function to record interesting scenes in their daily life. When users produce videos, they often want to share them with others on the Internet. Because of the limit of the storage of mobile phones, a user may copy a video clip

into a personal computer and does not keep it in his/her mobile device.

When users share their videos on the Internet, they do not want to see them suffering from copyright-infringing downloads by illegal users. In order to protect the copyright of videos, a lossless visible watermarking method is proposed in this study. The method embeds a given watermark image into an H.263 video, resulting in a watermarked video with a visible watermark in the video frames. Because relevant watermarking information is also embedded in the video, the original video can be recovered losslessly without using the original watermark in the recovery process. In addition, a user-specified key is adopted for randomizing the content of the embedded watermark to prevent illegal removal of the visible watermark. Good experimental results show the feasibility of the proposed method.

In the remainder of this paper, we first review the standard of the H.263 video in Section 2. Then, we describe the proposed lossless visible watermarking method in Section 3. Then, in Section 4 we describe how to remove the visible watermark to recover the original video. Some experimental results are shown in Section 5, followed by conclusions in Section 6.

2. REVIEW OF H.263 VIDEO FORMAT

The most popular multimedia container format for mobile devices is the 3GP defined by the 3rd Generation Partnership Project (3GPP) for use on mobile phones of the third-generation technology (3G) [5]. The file extension of 3GP is .3gp or .3gp2. One of the video compression techniques adopted in the 3GP format is ITU-T Recommendation H.263 [6]. In this study, we adopt the 3GP file with the H.263 video compression technique as our video file.

An H.263 video consists of a hierarchical structure with four layers. From top to bottom the layers are: Picture, Group of Blocks (GOB), Macroblock, and Blocks. There are five standardized picture formats: CIF, 4CIF, 16 CIF, Sub-QCIF and QCIF.

The data of each picture consists of a picture header which indicates the frame type of this picture. There are various types of frames in the H.263 format, including intra-coded frame (I), predictive-coded frame (P), bi-directionally predictive-coded frame (B), and PB frame which consist of two pictures coded as one unit: one P-

* This work is supported partially by the NSC Project Advanced Technologies and Applications for Next Generation Information Networks (II) with Project No. NSC93-2752-E-009-006-PAE and partially by the NSC National Digital Archives Program with Project No. NSC93-2422-H-009-001.

frame and one B-frame. The structure of an H.263 video is described in Figure 1.

H.263 pictures are divided into regions of macroblocks which are in order from the left border of the picture to the right border of the picture in the same row, each of which is called a Group of Blocks (GOB). Figure 2 shows the arrangement of GOBs in a CIF, 4CIF, 16 CIF, Sub-QCIF, and QCIF picture format.

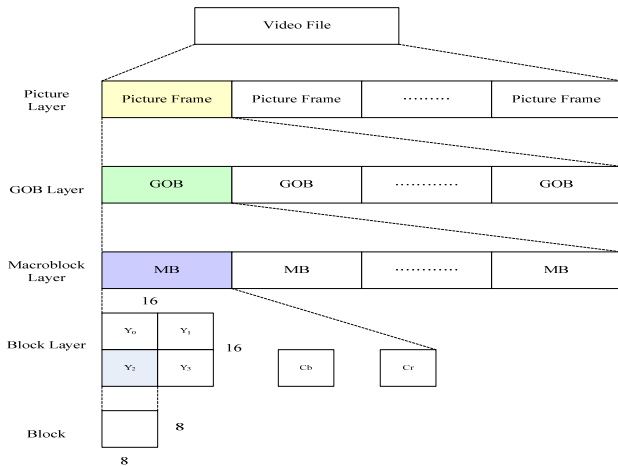


Figure 1. Structure of H.263 video stream.

In the H.263 standard, not only the transform coding method similar to the JPEG is applied to reduce the spatial redundancy, but also a motion-compensation algorithm is adopted to reduce the temporal redundancy between frames. A macroblock (MB) is adopted as a motion-compensation unit. There are different types of macroblocks in each type of frames. **Table 1** shows the relationships between frames and macroblocks.

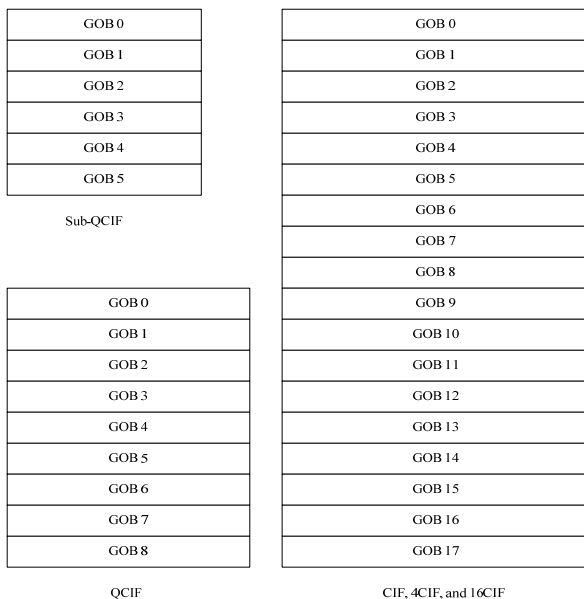


Figure 2 Arrangement of GOBs in H.263 format.

Table 1 Relationships between frames & macroblocks.

	Intra-coded MB	Forward-coded MB	Backward-coded MB	Bidirectionally-interpolated MB
I frame	•			
P frame	•	•		
B frame	•	•	•	•
PB frame	•	•	•	•

3. PROPOSED LOSSLESS VISIBLE WATERMARKING METHOD

A watermark to be embedded is assumed to be a binary image (a black and white logo, for instance). Each pixel in the watermark is referred to in the sequel as a black watermark pixel or a white one according to its binary value 1 or 0, respectively. The idea of the proposed embedding process for I-frames is to use each macroblock to embed a watermark pixel.

The embedded watermark pixels can be losslessly removed by checking the embedded watermarking information in the recovery process. The method of embedding a watermark depends on the random signals generated by a user-specified key. Without getting the correct key, the video cannot be recovered losslessly and correctly.

Because the P, B and P-B frames are coded by referencing to the forward or backward I-frames, the watermarked I-frames will influence the appearance of what they display. The change of the I-frames will cause damages to the video content.

The proposed method of embedding watermarking information is to insert a token which fits in with the H.263 encoding but will not be shown in a normal encoding procedure into the corresponding macroblock of the first I-frame. More details are described in the following sections.

3.1 Watermark Image Embedding Process

In the process of embedding a watermark image, the macroblock of each I-frame are taken as a unit. Each macroblock includes six blocks of 8×8 pixels, four luminance blocks, and two chrominance blocks. In order to show black watermark pixels in a macroblock, we modify the DC coefficient of every block in the macroblock. If the embedded watermark pixel is a white pixel, no change will be made in the DC coefficient in the macroblock. The details are described as an algorithm in the following.

Algorithm 1: Visible watermark embedding.

Input: An I-frame F except the first one in the video, a user-specified key K , and a binary watermark W .
Output: A watermarked I-frame F' with a visible watermark.

Steps:

1. Use the input key K as a seed for a random number generator to produce two random sequence S and S' where S is a sequence of $m \times n$ elements with its

value being in the range of 0 through 255 and S' is a sequence of $m \times n$ elements which are either 0 or 1, where m is the number of GOBs in every frame and n is the number of macroblocks in every GOB.

2. Denote S and S' as follows:

$$S = \{S_{ij} | i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n\};$$

$$S' = \{S'_{ij} | i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n\}.$$

3. For each macroblock in the input I-frame F , modify the DC coefficient C_{dc} of every block in this macroblock as follows:

$$\text{set } C_{dc} = C_{dc} \oplus S'_{ij} \text{ if } W_{ij} = 1 \text{ and } S_{ij} = 1;$$

$$\text{set } C_{dc} = C_{dc} \otimes S'_{ij} \text{ if } W_{ij} = 1 \text{ and } S_{ij} = 0;$$

$$\text{keep } C_{dc} \text{ unchanged, otherwise,}$$

where \oplus and \otimes mean exclusive-OR and the exclusive-NOR operations, respectively, on bit sequences.

3.2 Watermarking Information Embedding Process

In the proposed method, we use the first I-frame of a video to record the watermarking information for lossless recovery. Hence, the watermarked video can be recovered without the use of the original watermark image. We use a *recording token*, which fits in with the H.263 format but is not be used in normal encoding, to record a black watermark pixel.

In H.263 coding, some codes of LEVEL in FLC are not used, such as $(00000000)_2$ and $(10000000)_2$. Therefore, we use the token which is a combination of the ESCAPE code, the LAST with its code being $(1)_2$, the RUN with its code being $(000000)_2$, and the LEVEL with its code being $(00000000)_2$. The structure of the recording token is shown in Figure 3 and a detailed algorithm is described as follows.

Algorithm 2: Watermarking information embedding.

Input: The first I-frame F of a video, the watermark image W , and the recording token T .

Output: A recorded I-frame F' .

Steps:

1. For every macroblock in F , take the first luminance block as the recording block.
2. Find the last coefficient code C in the block and modify the block data as follows:

if $W_{ij} = 1$, *then change* C *into the non-last code and insert the recording token* T ;
otherwise, make no change to C .

	ESCAPE	LAST	RUN	LEVEL
Binary format	0000011	1	000000	00000000

Figure 3. Structure of the recording token.

4. RECOVERY OF WATERMARKED VIDEO BY REMOVING VISIBLE WATERMARK

In the recovery process, the watermarking information can be extracted from the first I-frame and

then the video content can be recovered with a correct input key. A detailed algorithm of the recovery is described as follows.

Algorithm 3: Recovery of watermarked video.

Input: A user-specified key K and a watermarked video V .

Output: A recovered video V' .

Steps:

1. Use the input key K as a seed for a random number generator to produce two random sequence S and S' where S is a sequence of $m \times n$ elements with its value being in the range of 0 through 255 and S' is a sequence of $m \times n$ elements which are either 0 or 1, where m is the number of GOBs in every frame and n is the number of macroblocks in every GOB.
2. Denote S and S' as follows:

$$S = \{S_{ij} | i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n\};$$

$$S' = \{S'_{ij} | i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n\}.$$

3. For the first I-frame in T_v , search and remove the recording token and record the watermarking information in W which is an $m \times n$ matrix. If there is a recording token in the corresponding macroblocks, set W_{ij} as 1; else, as 0.
4. For every macroblock M_{ij} of the following I-frames in T_v , modify the DC coefficient C_{dc} of the blocks as follows:

$$\text{set } C_{dc} = C_{dc} \oplus S'_{ij} \text{ if } W_{ij} = 1 \text{ and } S_{ij} = 1;$$

$$\text{set } C_{dc} = C_{dc} \otimes S'_{ij} \text{ if } W_{ij} = 1 \text{ and } S_{ij} = 0;$$

$$\text{keep } C_{dc} \text{ unchanged, otherwise.}$$

5. After every I-frame is processed, the recovered video V' is the desired result.

5. EXPERIMENTAL RESULTS

In our experiments, a watermarked video is produced using Algorithm 1 and Algorithm 2 with the input video being an H.263 video downloaded from the Internet and the key used is "123456." Six frames extracted from the watermarked video are shown in Figure 4. Figure 4(a) is the first I-frame after the watermarking information embedding process as stated in Algorithm 2. Frames extracted from the recovered video using the correct key and an incorrect one are shown in Figure 5 and Figure 6, respectively, with the recovered videos produced using Algorithm 3 with the input being the watermarked video.

In Figure 5, the recovered video is compared pixel by pixel to the original video, and it is verified that the pixel values of the recovered video are identically to that of the original video. In Figure 6, we observe that there is noise in the watermarked region after the recovery process with an incorrect key. The "watermark image" is clearly visible and influences the observability of the video frames after such a watermark removal attempt.

6. CONCLUSIONS

In this paper, a lossless watermarking technique for videos has been proposed. Users can share their videos on the Internet with more security using the proposed method. Even when the web server is cracked by hackers, the watermarked videos are still defended by the proposed watermark technique for copyright protection. Furthermore, the watermarked videos can be shared among people who know the protected key of the videos. Without the correct key, the watermarked videos cannot be recovered losslessly. There are two main advantages of the proposed technique. First, the watermarking information has been embedded into watermarked videos. Hence, the watermark image is not necessary for recovery process. Second, the complexity of recovery of the watermarked videos without the correct key is up to $(256)^n$, where n is the number of macroblocks whose DC values are changed by the watermark embedding process. It is more difficult to recover the watermarked videos illegally than the traditional watermarking technique for videos.

REFERENCES

- [1] H. Y. Chen and W. H. Tsai, "Verification of MPEG Video Contents by Random Signal Hiding," *IPPR Conference on Computer Vision, Graphics, and Image Processing*, Kinmen, Taiwan, August 16-18, 2003, pp. 692-701.
- [2] J. Song and K. J. R. Liu, "A Data Embedding Scheme for H.263 Compatible Video Coding," *Proceedings IEEE International Symposium on Circuits and Systems 1999*, Orlando, U.S.A., May 30-June 02 1999, Volume 4, pp. 390-393.
- [3] J. Meng and S. F. Chang, "Embedding Visible Video Watermarks in the Compressed Domain," *Proceedings of IEEE International Conference on Image Processing*, Chicago, IL, U. S. A., Oct. 1998, Volume 1, pp. 474-477.
- [4] "Universal Mobile Telecommunications System (UMTS); Transparent end-to-end streaming service; 3GPP file format (3GP)," ETSI TS 126 244 V6.3.0, Mar. 2005.
- [5] "Video Coding For Low Bitrate Communication," DRAFT ITU-T Recommendation H.263, May 1996.
- [6] Z. N. Li and M. S. Drew, *Fundamentals of Multimedia*, Pearson Education, U. S. A., 2004.
- [7] D. Cross and B. G. Mobasseri, "Watermarking for Self-authentication of Compressed Video," *Proceeding of IEEE International Conference on Image Processing*, New York, U. S. A., Sept. 2002, Volume 2, pp. 913-916.
- [8] Y. Takishima, M. Wada, and H. Murakami, "Reversible Variable Length Codes," *IEEE Transactions on Communications*, 1995, Volume 43, Issue 234, pp. 158-162.
- [9] K. H. Rahouma, "Utilization of multiple block cipher hashing in authentication and digital signatures," *Proceedings of IEEE International Conference on Networks*, Las Vegas, U. S. A., 5-8 September 2000, pp. 257 - 261.

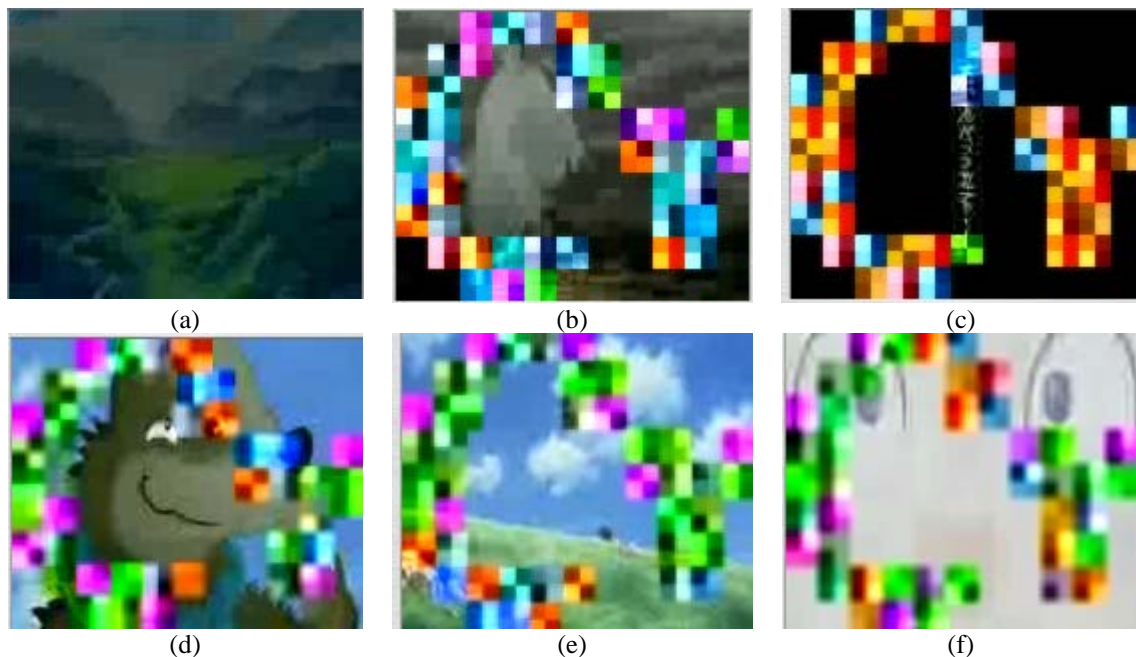


Figure 4 Some frames extracted from the watermarked videos. (a) The first I-frame with the watermark information embedded. (b) (c) (d) (e) (f) Some frames after watermarking with user input key.



Figure 5. Illustration of the recovered video with the correct key. (a) (b) (c) (d) (e) (f) Some frames extracted from the recovered video with the correct key.

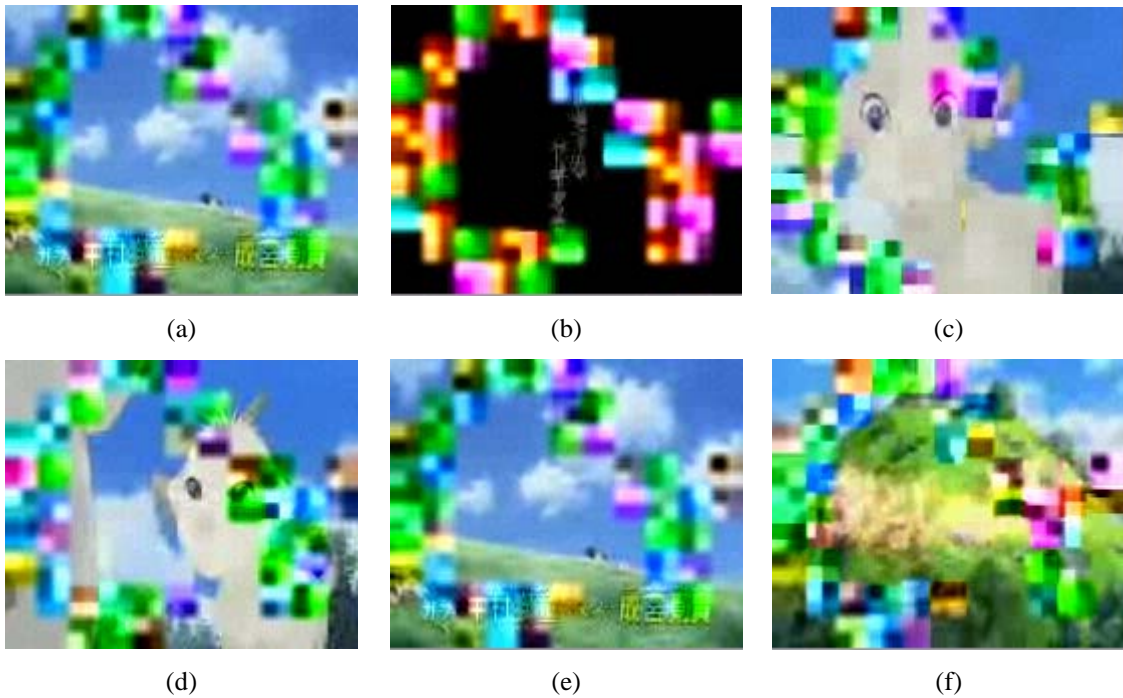


Figure 6. Illustration of the recovered video with an incorrect key. (a) (b) (c) (d) (e) (f) Some frames extracted from the recovered video with the incorrect key.