LOSSLESS AUTHENTICATION OF MPEG-2 VIDEO

Rui Du[a] *Jessica Fridrich*[b]

[a]MTL Systems, Inc. Beavercreek, Ohio
[b]Department of ECE, SUNY Binghamton, New York

ABSTRACT

In authentication using watermarking, the original media needs to be slightly modified in order to embed a short media digest in the media itself. Lossless authentication watermark achieves the same goal with the advantage that the distortion can be erased if media authenticity is positively verified. In this paper, we extend lossless data embedding methods originally developed for the JPEG image format to digital video in the MPEG-2 format. Two new lossless watermarking methods for authentication of digital videos are presented. In the first approach, each frame contains its hash embedded losslessly, while in the second approach we embed a hash of a group of frames in B frames only. Implementation issues, such as real-time performance, are also addressed.

1. INTRODUCTION

Analog surveillance video cameras, reconnaissance cameras, and industrial cameras will soon be replaced with digital hardware. Digitized video will provide higher quality and offer other advantages of digital media, such as convenient sharing and storage, easy editing and enhancement, and direct access. However, digital video and any other multimedia digital objects can be easily modified using video editing software. Thus, the question of authenticity and data integrity of digital video may become critical, for example in cases when a digital video clip is presented as evidence in the court. Authentication watermarks provide one possible solution to this problem.

In authentication using watermarking, a short media digest, such as the cryptographic hash, is embedded in the media itself rather than attached to it in a header or a separate file [1–3]. This has the advantage that the media can authenticate itself without accessing any side information. The embedded digest is invisibly hidden in the media and does not substantially increase the media file size and stays with the media even after a lossless format conversion. Note that the original video needs to be slightly modified in order to embed the digest. This slight loss of quality and artifacts visibility is difficult to evaluate and quantify especially for digital video [4]. The lack of understanding the visibility of watermarks in video may prevent wider spread of watermarking technology to applications that require high quality video, such as military or medical digital multimedia files.

The concept of invertible (or lossless) authentication and data embedding enables us to “undo” the modification due to data embedding [5]. After a positive integrity check, the original video can be losslessly recovered. In this paper, we extend lossless data embedding methods originally developed for the JPEG image format to digital video in the MPEG-2 format. In the next section, we describe an approach in which a frame hash is embedded in each frame. In Section 3, an alternative approach is explained in which the hash is calculated from a group of pictures between two reference frames and the hash is embedded in B frames only. Implementation issues, such as real-time performance, are addressed in Section 4.

2. AUTHENTICATION BY FRAMES

In contrast to images, a video stream has three dimensions — two spatial dimensions and a temporal one. Malicious users may swap frames or drop some frames to modify the video stream. To detect this type of tampering, the authentication code should include a hash of the frame content and the frame index. By extracting the frame index, one can re-order the tampered video frames and detect the dropped frames. In the first method for MPEG lossless authentication, the hash is calculated for each individual frame. The hash and the frame index are embedded into the same frame using Method 2 described in [5]. The chrominance blocks of both intra and non-intra macro-blocks are used for embedding. To reduce the complexity of hash computation, the hash is computed from the non-zero DCT coefficients instead of the pixel values from the whole frame. The input MPEG data is first decoded with a Huffman decoder (to obtain the quantized DCT coefficients). The non-zero DCT coefficients are used for calculating the hash. Then, a selected quantization factor is halved and the corresponding DCT coefficients are multiplied by 2. As a result, the LSBs of those coefficients will all become zeros, which can be used for lossless embedding of authentication bits (for details, see [5]). Finally, the modified DCT coefficients are encoded with a Huffman encoder to produce the new MPEG video stream with embedded authentication information. By working with the quantized DCT coefficients instead of the pixels, we avoid having to perform quantization, DCT transform, and motion compensation (the most time consuming operation).

Another advantage of using DCT coefficients instead of pixels is that authentication of P or B frames does not depend on the authentication results of previous reference frames. If I or P frames are tampered, the pixel values of future frames will also change because they are encoded based on these tampered reference frames. In this case, authentication by pixels cannot determine whether or not the subsequent frames have been tampered. The reason for choosing non-zero DCT coefficients rather than the Huffman code, which is shorter than the sequence of non-zero DCT coefficients and appears more efficient for hash calculation, lies in the verification processing.

During the verification step, the embedded authentication bits are first extracted from the selected DCT coefficients. Then, the quantization coefficients and DCT coefficients are losslessly recovered. From the recovered non-zero DCT coefficients, a new hash is calculated. The calculated hash is compared with the extracted hash to verify the authenticity of the video frame. The extracted frame index is used to detect tampering with the temporal sequence of the frames. If Huffman code were used for hash calculation, we would need to use the Huffman compression during recovery and verification to calculate the hash. This would increase the complexity of verification and recovery.

Figure 3 shows the PSNR (Peak Signal to Noise Ratio) between the original MPEG video stream and the MPEG video stream with authentication information embedded. PSNR is calculated according to Eq. (1) and (2), where P_{orig} and P_{stego} are the original and watermarked frames, and M and N are frame dimensions. We show the results for the first 10 frames of the test MPEG movie clip 'Hummingbird'. Because I and P frames are used as reference frames for subsequent B and P frames, the distortion introduced by data embedding in these frames will spread to subsequent frames until the next I frame is encountered. Notice that the PSNR for the 6th frame is higher than that for the 4th frame because the 6th frame uses an I frame as a backward reference frame.

$$MSE = \frac{1}{MN} \sum_{i=0}^{N} \sum_{j=0}^{M} \left(P_{orig}(i,j) - P_{stego}(i,j) \right)^2 \quad (1)$$

$$PSNR = 10 \times \log\left(255^2 / MSE\right) \quad (2)$$

3. AUTHENTICATION BY GROUPS OF FRAMES

In the previous authentication method, the distortion due to data embedding could become perceptible when the distance between two I frames is too large. Although this is not typically true for MPEG video streams (the distance between two I frames is 15 frames), it would frequently happen to two-way communication standards, such as the ITU-H.263, because for those formats the random access ability is unnecessary and the distance between two I frames could be 100 frames or more. To avoid the spread of distortion, the second method embeds authentication information into B frames only because B frames are not used as reference frames for other frames' encoding and the distortion in B frames due to data embedding will not spread to subsequent frames. Authentication by groups of frames was proposed to address this issue.

Figure 2 shows the scheme for MPEG authentication by groups of frames. A group of frames includes B frames and their reference frames. All non-zero DCT coefficients from a group of frames form the input for hash calculation. The authentication data is formed by the index of the group of frames and the hash, which are both embedded into the B frames. As shown in Figure 2, the reference frames enter the hash calculation for both adjacent groups of frames. In our implementation, only chrominance blocks of non-intra macro-blocks are used for data embedding.

Unlike authentication by frames, authentication by groups of frames introduces less distortion but obviously cannot pinpoint tampering to individual frames. However, this can be quite acceptable for a video stream because the ability to locate the tampering up to a group of frames may be sufficient in most cases. Also, the improved PSNR between two I frames for this second method is an important asset especially for videos, where the perceptibility of artifacts is not well understood.

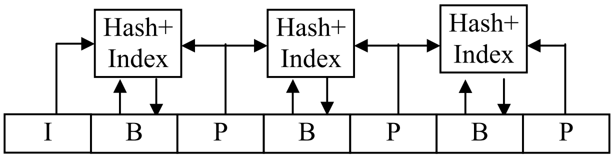

Figure 1 Authentication by groups of frames

Figure 4 shows the *PSNR* for authentication by groups of frames for the first ten frames of the "Hummingbird'. At the first sight, the P frames for the original video and the authenticated video should be the same. However, due to the definition of inverse quantization, the process of reconstruction of non-intra blocks is not completely lossless. Eq. (3) is the definition of inverse quantization in MPEG-2. In this formula, $Q(i, j)$ is the quantization coefficient, $QD(t, i, j)$ denotes the t^{th} frame's quantized DCT coefficients, $D(i, j)$ are the de-quantized DCT coefficients, and S is the scale factor which is adjusted by the rate control,

$$D(i,j) = ((2 \times Q(i,j) + k) \times QD(t,i,j) \times S)/32\,, \quad (3)$$

where $k = sign(Q(i, j))$ for non intra blocks, and $k = 0$ for intra blocks. For lossless embedding, $Q(i, j)$ is divided by 2 and $QD(t, i, j)$ is multiplied by 2. Unfortunately, this operation can not give the same $D(i, j)$ as Eq. (3) because k is not zero for non-intra blocks. This leads to the difference in P frames.

4. ACHIEVING REAL TIME PERFORMANCE

Video processing, such as authentication and data embedding, are in general time and memory consuming operations. While the authentication complexity is not as crucial for a single image, low complexity of the authentication for video may be one of the most important requirements in practice. In particular, if the authentication algorithm can perform in real time, the usefulness of the authentication methods will increase dramatically because no need for pre-decoding is necessary and the original video may be played as its integrity is being evaluated. To achieve real-time performance for our proposed lossless video authentication, three speedup techniques were used in the implementation. The first speedup technique is reducing the usage of branch code. The second speedup technique uses the pre-fetch instruction, and the third technique uses the SIMD (Single Instruction Multi Data) instructions. Of all the three speedup techniques, the pre-fetch has the largest contribution because it reduces the time used for memory access by more than ten CPU clocks. We tested five MPEG-2 video streams on a PIII 550 machine with 128M memory and the Windows98 operating system (see Table 1). Screen shots from the five test video streams are shown in Figure 2. Full versions of all five test video streams can be downloaded from the following web site: http://bingweb.binghamton.edu/~rdu.

Figure 2 Test MPEG-2 video streams: House, Dance, Ski, Hummingbird, and Rocket

Video	Dimension	By frames		By groups of frames	
		Authentication	Verification	Authentication	Verification
House	720×480, 61	13.84	4.77	12.11	4.75
Dance	704×480, 24	13.16	3.29	11.19	4.53
Ski	720×576, 31	13.69	4.42	12.53	3.97
Hummingbird	352×240, 89	22.16	13.34	20.14	12.82
Rocket	160×120, 226	109.54	64.66	99.86	59.85

Table 1 Processing speed (frames per second) for five test video clips

5. CONCLUSIONS AND SUMMARY

In this paper, we propose a lossless authentication watermark for MPEG-2 video. Two methods are presented: authentication by frames and authentication by groups of frames. In authentication by frames, frame hash is embedded losslessly in each frame, while in authentication by groups of frames, the hash of the group of frames between two reference frames is hashed and the authentication code embedded in B frames of the group. The first approach provides better temporal localization while the second approach introduces less distortion. The algorithms can perform in real-time enabling concurrent authentication while playing the video.

6. ACKNOWLEGMENTS

The work on this paper was supported by Air Force Research Laboratory, Air Force Material Command, USAF, under a research grant number F30602-00-1-0521. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of Air Force Research Laboratory, or the U. S. Government.

7. REFERENCES

[1] S.-F. Chang and C.-Y. Lin, "Issues and Solutions for Authenticating MPEG Video", *Proc. SPIE Photonics West*, vol. 3657, 1999, pp. 54–65.

[2] T. Kalker, G. Depovere, J. Haitsma, M.J. Maes, "Video Watermarking System for Broadcast MOnitoring", *Proc. SPIE Photonics West*, vol. 3657, 1999, pp. 103–112.

[3] J. Dittmann and M. Steinbach, "Combined Video and Audio Watermarking: Embedding Content Information in Multimedia Data",*Proc. SPIE Photonics West*, vol. 3971, 2000, pp. 455–464.

[4] W. Macy and M.J. Holliman, "Quality Evaluation of Watermarked Video", *Proc. SPIE Photonics West*, vol. 3971, 2000, pp. 486–501.

[5] J. Fridrich, M. Goljan, and R. Du. "Invertible Authentication Watermark for JPEG Images." *ITCC 2001*, Las Vegas, Nevada, April 2–4, 2001, pp.

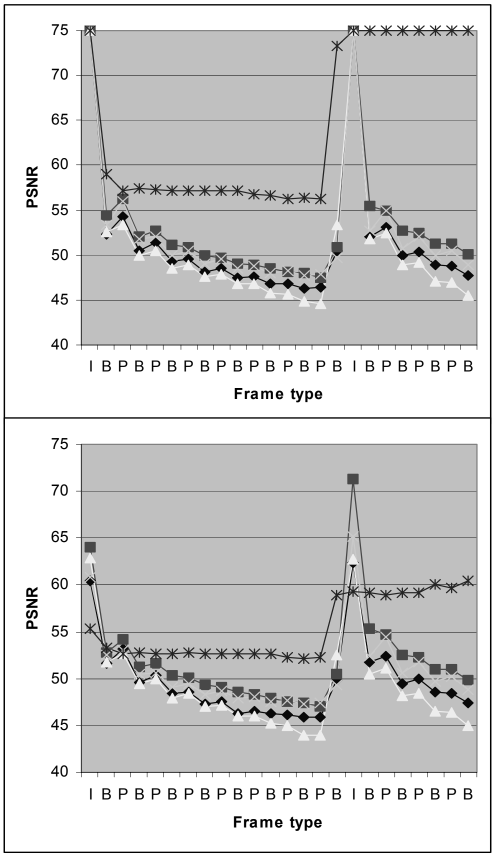

Figure 3 and 4 PSNR for authentication by frames (above) and by groups of frames (below). The symbols denote 5 different video clips from Fig. 2.