A Digital Watermarking Scheme for Authenticating H.264/AVC Compressed Videos

Po-Chyi Su and Ing-Fan Chen
Department of Computer Science and Information Engineering
National Central University, Jhongli, Taiwan
Email: pochyisu@csie.ncu.edu.tw

***Abstract*—In this research, we propose a digital watermarking scheme under the framework of H.264/AVC. The scheme is mainly used for video authentication to ensure that the frame or shot order in a video be maintained. The watermark signal is embedded into quantization indices of intra-coded frames to achieve the effective watermark embedding/detection and preserve the compact size of video data. The luminance masking of Watson's visual model is employed to guarantee the imperceptibility of watermark signal. The coding procedure of H.264/AVC is taken into account in the design of watermarking procedures for the efficient execution. The experimental results demonstrate the feasibility of the proposed video authentication system.**

I. Introduction

The authentication of multimedia data [1], [2] has drawn tremendous attention these years due to the fact that digital data can be manipulated easily by convenient editing facilities. The conventional methods to authenticate digital data involve calculating a cryptographic checksum or a digest from the digital message and sent along. The authenticator calculates the digest and compare it with the received one to determine if the message has been modified. Although conventional authentication techniques perform well when we require bit-by-bit accuracy, the scenario for digital images and videos may be different. Multimedia users care more about whether the meaning of data has been changed, instead of sparse binary errors, which may be caused by transmission, storage or data processing procedures. Such normal data processing procedures as lossy compression should be allowed since they do not change the content or affect the viewers' usage. In addition, it may be necessary to locate the tampered part of data to help identify the attackers' motivation.

There are two major methodologies of multimedia authentication: signature-based approaches and fragile watermarking. The signatured-based approaches are similar to conventional methodologies in that the digest or signature is generated and appended to the data for authentication but the extracted digest will be required to be resilient to lossy compression but sensitive to malicious attacks. For fragile watermarking, the status or existence of embedded digital watermark can be used to determine if the content has been modified. It should be noted that these methods achieve the so-called "picture-level" authentication so that the tampered pixels within an image or a video frame can be identified. In this research, we focus on digital video and concern more about the correctness of frame or shot order since deleting, inserting or replacing shots are more common attacks to digital videos. The above-mentioned methods may only detect the beginning of attacks and may not be able to locate the attacked frames. We therefore take a different approach to embed the robust watermarks into frames as frame/shot serial numbers to achieve "sequence-level" authentication. Under this scenario, we have the following requirements. First of all, the embedded watermark should be imperceptible and survive transcoding processes, which do not change the video content. Next, the watermark embedding/detection should be efficient since many applications require real-time execution. Besides, the scheme should be able to signal/locate the tampered segment of frames. Furthermore, the watermark can be detected in any video segments so we have to deal with the synchronization issue. Finally, the video size increase due to the watermark embedding process should be limited. Therefore, we will design the watermarking scheme to meet these requirements.

Our strategy is to utilize the traditional spread-spectrum watermarking approach to achieve both the imperceptibility of watermark and the robustness against transcoding. We adopt the approach of watermarking in the compressed domain for efficiency. It should be noted that videos are always compressed to facilitate storage and transmission. The watermarking processes usually have similar procedures with the compression processes so combining both can save a vast amount of computation. Besides, employing a specific compression format is a reasonable choice here since transcoding the video to other formats can be ruled as an illegal manipulation in a proprietary system. We choose H.264/AVC as the underlying video codec because of its decent performance. Most of the existing research works on H.264 video watermarking focus on the low complexity design [3], the high bit-rate information hiding [4], the multiple functions [5], increasing security [6] or robust embedding to resist H.264 lossy compression [7]. The common drawbacks of the existing works are that the issues of increased watermarked video size are overlooked and the synchronization problem coming from detecting the watermark in a cropped video. We think these are quite important in designing a practical video watermarking scheme and should be taken into serious consideration. We will present our watermarking scheme in the following sections. The watermark embedding process will be described in Section II, followed by the detection process in Section III. Experimental results will be shown in Section IV and the conclusive remarks are

given in Section V.

II. Watermark Embedding

We first discuss the data selected for watermarking. As in many existing digital watermarking schemes, the watermark will be embedded in transform coefficients. To better combine the watermarking and coding procedures, we embed the watermark in quantization indices to avoid the watermark from being affected by the quantization step. Besides, we only embed digital watermark in I-frames. The reason is explained as follows. The watermark embedding may introduce redundancy to the host data. As video coding procedures make use of the temporal prediction for efficient compression, the data rate in P-frames or B-frames will be low. Adding redundancy to the residues in these frames will increase the data rate radically and may affect the normal usage of video.

Because a DCT-like transform is used in H.264/AVC, the watermark embedding will be based on Watson's perceptual model [8] to meet the requirement of imperceptibility. Watson's perceptual model helps in determining the amount of watermark energy or Just Noticeable Difference (JND) that is allowed to be embedded into each coefficient. The model basically takes two masking effects into account, *i.e.* the luminance masking and contrast masking. The luminance masking relies on the average luminance of a block and some global setting such the viewing distance and position of the coefficient in the block. The contrast masking has to take the value of individual coefficient into account. In our scheme, we will only use the luminance masking for watermark adaptation without considering the component contrast. The reason is described as follows. Our digital watermark will be embedded into the quantized transform coefficients in I frames, in which H.264/AVC applies the intra prediction to further reduce the data needed for coding and the intra prediction will make the exact values of transform coefficients in the luma block unavailable. The luminance masking in Watson's model only depends on the DC value of transform block. In the encoding process, we can calculate the average value in the pixel domain straightforwardly from the incoming uncompressed raw block so that the luminance masking can be decided. Among the 16 coefficients, the lowest three frequency terms will be excluded from watermarking to further avoid causing unpleasant artifacts. It should be noted the resulting JND has to be divided by the Quantization Parameter (QP) and rounded to be an integer to determine the maximum allowable modification of quantization index.

The spread-spectrum watermarking approach is adopted as mentioned before. We generate W Hadamard sequences taking values $+1$ and -1 as the watermark sequences, $\mathbf{w}^N$. N is the serial number in the range of $[1, W]$ and will be the information we would like to embed in a frame. We sequentially choose a sequence to embed in consecutive I frames. However, the Hadamard sequence has to be scrambled before embedding by a scrambling sequence, $\mathbf{r}$. Therefore, the watermark embedding is as follows. For a quantization index, $q_{i,j,k}$, the watermarked value, $q'_{i,j,k}$, is calculated by

$$q'_{i,j,k} = \begin{cases} 0, & q_{i,j,k} = 0 \\ q_{i,j,k} + \{r_n \times w_n^{N_f}\} \times a^q_{i,j,k}, & \text{otherwise.} \end{cases} \quad (1)$$

N_f is the serial number to be embedded. r_n and $w_n^{N_f}$ are the n^{th} components of the random sequence, $\mathbf{r}$ and the N_f-th watermark sequence, $\mathbf{w}^{N_f}$, respectively, and n depends on the index position i, j, k. $a^q_{i,j,k}$ is the maximum allowed change of quantization index. It should be emphasized that only the non-zero quantization indices are chosen for watermarking to avoid significantly increasing the size of the compressed video.

It is worth noting that using a fixed $\mathbf{r}$ in the entire video may raise certain security concerns. Although using different scrambling sequences will help to increase the difficulty of "guessing" the watermark sequence by attackers, it may also complicate the watermark detection process. For example, if only one segment of the investigated video is chosen for watermark detection, it will be difficult to know exactly which scrambling sequence is used in a specific frame. In our scheme, the scrambling sequence $\mathbf{r}$ is generated by using a hash-like value, h, which is calculated from the video content, as the seed to make the scrambling sequence content adaptive. The other advantage of this strategy is to avoid the so-called "copy attack"[9], in which the attacker can extract noises, which may contain watermark signals, and then tamper the frame content and put the noises back to make a fake frame that may pass the authentication process. A compression resilient hashing is derived to accommodate the applications of digital watermarking. We first apply the shot change detection to determine the frame of shot boundary. The frame can then help to generate a shot hash used for generating the scrambling sequence for an entire shot. It should be noted that we do not make the scrambling sequence depend on every single frame since the scrambling sequence is also required in the decoder side and we do not want the decoder to extract all the frames for watermark detection.

To generate the shot hash, we use our previously proposed method [10] of generating the authentication code in a signature-based authentication scheme. For a 352×288 video frame, we down-sample it to 22×18 and extract the central 16×16 block. We then apply singular value decomposition on this mean-removed block, $\mathbf{X}$, to decompose it as

$$\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T = \sum_{i=1}^{16} \lambda_i^{\frac{1}{2}} \mathbf{u}_i \mathbf{v}_i^T \quad (2)$$

where $\mathbf{u}_i$, $\mathbf{v}_i$ are columns of $\mathbf{U}$, $\mathbf{V}$, representing eigenvectors of $\mathbf{X}\mathbf{X}^T$ and $\mathbf{X}^T\mathbf{X}$, respectively, and $\mathbf{\Lambda}$ is a diagonal matrix with eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m$ on the diagonal line. We choose the first eigenvectors, $\mathbf{u}_1$ and $\mathbf{v}_1$ as the extracted feature of the block. We apply the technique of Vector Quantization (VQ) by training VQ codebooks $\mathcal{C}_u$ for $\mathbf{u}_1$ and $\mathcal{C}_v$ for $\mathbf{v}_1$. The codebook index will then be used as a seed to generate the scrambling sequence.

In our design, the first frame of the scene will be picked to calculate the shot hash value. To be more specific, the

histograms of adjacent frames will be compared and if the difference is larger than a threshold T_c, the scene change is ruled as happening and the hash value calculated from this frame with scene change will be used in watermarking a future I frame. However, it should be worth noting that, given that there are three shots, $\mathbf{S_i}$, $\mathbf{S_j}$ and $\mathbf{S_k}$ and the scene change frame between $\mathbf{S_i}$ and $\mathbf{S_j}$ is $F_{i,j}$, we will use $F_{i,j}$ to generate the scrambling sequence for $\mathbf{S_k}$, instead of $\mathbf{S_j}$, so that we can link the adjacent shots to avoid misses of detections if replacing or deleting shots has been applied.

III. Watermark Detection

The watermark detection is based on the following:

$$\rho^{N_t} = \frac{1}{\parallel \hat{\mathbf{q}} \parallel} \sum_{\hat{q}_{i,j,k} \neq 0} \hat{q}_{i,j,k} \times (r_n \times w_n^{N_t}), \tag{3}$$

where $\hat{q}_{i,j,k}$ is the quantization index in the investigated frame and N_t is the serial number of the tested watermark sequence. All the W sequences will be examined. $\parallel \hat{\mathbf{q}} \parallel$ is the norm of the nonzero quantization indices, which can help to make I and P frames without watermark have similar low responses so that we can better predict the false positive and negative rates of the scheme. N_t with its detection response, ρ^{N_t}, larger than the detection threshold is viewed as the extracted value or serial number of this frame. It should be noted that the watermark adaptation, $a^q_{i,j,k}$ in Eq.(1), is not included in the detection. This omission may seem to be unreasonable since luminance masking in Watson's model may be estimated in the decoder, given that only the DC value, the coefficient location, (i, j), and some global settings are required. However, in order to attain DC values, we have to expand all the full frames due to the use of the intra-prediction in H.264/AVC. The encoder doesn't have this problem since the full raw frames are accessible before compression and calculating the DC value can be done by straightforwardly averaging pixel values. It is also worth noting that the watermark detection will be applied to all the frames, not just I-frames. In other words, we will also detect the watermark in residues of P-frames so that the change of coding structure may not affect watermark detection. As in the encoder, the video frame content will be used to calculate the hash value for generating $\mathbf{r}$ in the decoder. The major difference is that the hash value in the encoder is calculated from the uncompressed video frame while that in the decoder is computed from the expanded video frame. By this design, we may expect that expanding all the frames in the decoder is necessary for watermark detection, which will contradict our objective of efficient execution by integrating watermarking and coding procedures. Therefore, we will try to make this expanding process as rare as possible. Our scene-change detection in the decoder will tend to analyze the data in the compressed domain and the procedure is shown as follows. We first expand two I-frames, I_i and I_j, and then compute their color difference. If the difference is larger than the threshold T_c, a scene change is identified as occurring between I_i and I_j. Next, we calculate the percentage of intra-coding, denoted by $Pr_p^{(I)}$, in all the P-frames between I_i and I_j. The P-frame with the largest $Pr_p^{(I)}$, denoted by P_m, is chosen and $Pr_{p_m}^{(I)}$ is compared to the other threshold T_P. If $Pr_{p_m}^{(I)} > T_P$, we expand P_m and compare its histogram with those of I_i and I_j to determine the actual scene change. Else, we expand the frame right before I_j and compare histograms to determine the exact frame of shot boundary. In other words, only I frames and predicted frames with possible scene change will be expanded to the full frame size for hash-value calculation. Since the number of such frames is not large, in terms of efficiency, this procedure should be acceptable. By this design, the watermark detection will not be able to proceed on the fly with the decoding process so the quantization indices of a frame have to be stored for subsequent watermark detection.

TABLE I
THE COMPARISONS OF THE ORIGINAL AND WATERMARKED VIDEOS

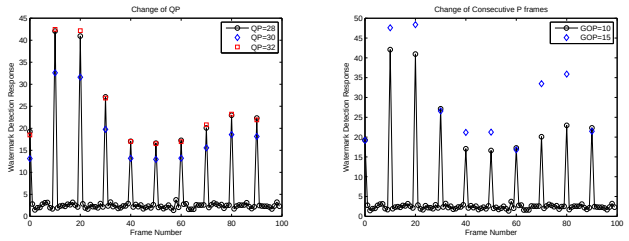
	Original H.264 video		*Watermarked H.264 video*	
Video	Bit-rate	PSNR	Bit-rate	PSNR
Monitor	423Kbps	37.91dB	433Kbps	37.56dB
Stefan	1469Kbps	35.58dB	1504Kbps	35.04dB
Foreman	547Kbps	37.07dB	566Kbps	36.69dB
Akiyo	381Kbps	38.42dB	399Kbps	37.83dB

IV. Experimental Results

In the experiments, we adopt the H.264/AVC baseline mode to make the encoding faster. One I frame is followed by 9 P frames so the watermark is embedded every ten frames. The PSNR drop and video size increase are shown in Table.I. Four CIF videos with 300 frames are tested. It should be noted that we do not turn on the rate control and set QP=28 so that we can see the effects of watermarking more clearly. The results are satisfactory as the bit-rate increase and quality degradation are limited. Since only the luminance masking of Watson's model is used in weighting the embedded watermark, the perceptual quality is not severely affected.

Next, we will test the robustness of the watermark against the allowable transcoding procedures and then apply manipulations of frames to demonstrate the ability of locating tampered frames in the proposed scheme. Two transcoding processes are considered, including the change of QP and the different number of consecutive P frames following an I frame. The results are shown in Fig.1. The circles marked in Fig.1(a) indicate the peak values of watermark detection responses in each frame without any transcoding and the watermark is correctly detected. Then we change the QP from 28 to 30 and 32 and the peak values are shown as diamond and square marks, respectively. Again the watermark can be detected easily. It maybe a bit surprising that larger QP values, which will decrease the video sizes, may have even larger responses as shown in Fig.1(a). The larger responses actually come from the use of normalization factor, $\parallel \hat{\mathbf{q}} \parallel$, in Eq.(3). We then change the number of P frames during transcoding. It is a necessary test since the transcoding may not use the same, in MPEG language, GOP size. Besides, the attack may make an original I frame be processed by P frame due to possible

frame deletions. Here we change the GOP size from 10 to 15 and Fig.1(b) shows that the responses of GOP=15 in multiples of 10 frames (marked as diamonds) are high enough to detect the correct serial numbers.



(a) (b)

Fig. 1. The watermark detections for transcoding by (a) changing the QP and (b) changing the GOP size.

Finally, we concatenate 5 video segments, including "TableTennis", "Monitor", "Container", "Foreman" and "Stefan", each with 100 frames, as 5 shots. The GOP size is set as 30 for better viewing. And then we apply video editing including replacing, inserting and deleting a video shot. Fig.2(a) shows the detection result of no attack. Fig.2(b), (c) and (d) demonstrate the results of replacing, inserting and deleting the 3rd video shot. We can see that two shots are affected in each case since we always use the previous shot change frame to calculate the shot hash. The different extracted serial numbers will inform us of the case of manipulation.

V. Conclusion

A digital video watermarking scheme is proposed for authenticating H.264/AVC compressed videos. The procedures of watermarking are integrated with the coding routines in a close manner to achieve efficiency. Experimental results demonstrate the feasibility of the proposed approach. It should be noted that the watermarking scheme can be coupled with the signatured-based approaches to construct a more well-rounded video authentication system.

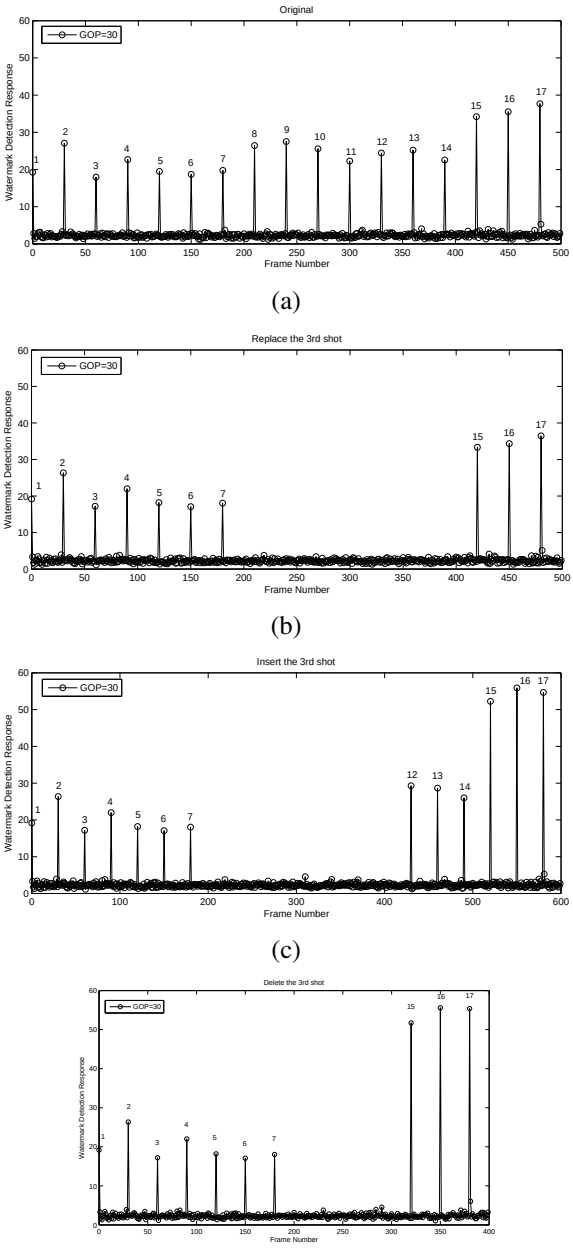


Fig. 2. The watermark detections for (a) the 5-shot video without attack and the cases of (b) replacing, (c) inserting the 3rd shot with other video segment and (d) deleting the 3rd shot.

References

[1] B. Zhu, M. Swanson, and A. Tewfik, "When seeing isn't believing - current multimedia authentication technologies and their applications," in *IEEE Signal Processing Magazine*, vol. 21, Mar. 2004, pp. 40–49.

[2] F. Bartolini, A. Tefas, M. Barni, and I. Pitas, "Image authentication techniques for surveillance applications," in *Proc. IEEE*, vol. 89, no. 10, Oct. 2001, pp. 1403–1418.

[3] J. Zhang, A. Ho, G. Qiu, and P. Marziliano, "Robust video watermarking of H.264/AVC," in *IEEE Transactions on Circuits and System-II: Express Briefs*, vol. 54, no. 2, February 2007, pp. 205–209.

[4] M. Yang and N. Bourbakis, "A high bitrate information hiding algorithm for digital video content under H.264/AVC compression," in *IEEE Int. Sym. On Circuits and Systems*, vol. 2, August 2005, pp. 935–938.

[5] G. Qiu, P. Marziliano, A. Ho, D. He, and Q. Sun, "A hybrid watermarking scheme for H.264/AVC video," in *Proceedings of the 17th International Conference on Pattern Recognition, ICPR*, vol. 4, August 2004, pp. 865–868.

[6] M. Noorkami and R. M. Mersereau, "Compressed-domain video watermarking for H.264," in *Proceedings of the International Conference on Image Processing, ICIP*, vol. 2, September 2005, pp. 890–893.

[7] G.-Z. Wu, Y.-J. Wang, and W.-H. Hsu, "Robust watermark embedding/detection algorithm for H.264 video," in *Journal of Electronic Imaging*, vol. 14, Jan.-Mar. 2005.

[8] A. B. Watson, "DCT quantization matrices visually optimized for individual images," in *Proc. SPIE, Human Vision, Visual Processing, and Digital Display*, vol. 1913, Bellingham, WA, 1993, pp. 202–216.

[9] C.-S. Lu, H.-Y. M. Liao, and M. Kutter, "Denoising and copy attacks resilient watermarking by exploiting knowledge at detector," in *IEEE Transatctions on Image Processing*, vol. 11, no. 3, 2002.

[10] P.-C. Su, C.-C. Chen, and H.-M. Chang, "Towards effective content authentication for digital videos by employing feature extraction and quantization," in *IEEE Trans. on Circuits and Systems for Video Technology, to appear*.