Verification of MPEG Video Contents by Random Signal Hiding

Hsin-Yu Chen (陳信宇) and Wen-Hsiang Tsai (蔡文祥)
Department of Computer & Information Science
National Chiao Tung University
1001 Ta Hsueh Rd., Hsinchu, Taiwan 300, R. O. C.
Tel: 886-3-5720631
Email:whtsai@cis.nctu.edu.tw

Abstract

Advanced information technologies have brought us plenty of convenience, yet through their use, MPEG videos can also be acquired or tampered with ease, resulting possibly in unauthorized uses or modifications. Therefore, it is compulsory to develop appropriate methods for video authentication. For the purpose of verifying the fidelity and integrity of MPEG videos, a video authentication method based on information hiding techniques is proposed in this study. The method not only can check whether a video has been tampered with, but also can show where and how the tampering was conducted.

Keywords: Video authentication, fidelity and integrity verification, information hiding.

1. Introduction

In this study, a method for content verification of MPEG videos is proposed. Digital videos can be easily modified nowadays using a lot of video editing software. Therefore, how to verify the integrity and fidelity of video contents is a very important issue. For instance, if a video were to be used by the court as evidence, to judge whether a suspect is guilty, the video would have to be authenticated first to make sure that modifications have not been made to it. In addition, because MPEG videos are usually transmitted across networks for many applications, such as environment surveillance, net meeting, videophoning, etc., these videos can be acquired and tampered with easily. Therefore, it is necessary to verify at the receiver site that the content of the received video is original and has not been modified. In recent years, a number of methods have been proposed for video authentication. Two typical approaches are the digital signature technique [1, 2] and the digital watermarking technique [3-6].

In this study, a video verification system is proposed. The basic task of such a system is to prove whether a given video has been tampered with or not. However, it is an even more essential requirement that the verification system can tell us where and how tampering was conducted in the given video. The proposed video verification system not only can check whether a video has been tampered with by a malicious user, but also can mark the tampered regions and recognize the tampering types.

Because a video stream may be regarded to possess three dimensions: two spatial ones and a temporal one, tampering manipulations in the video can be categorized into two different types: *spatial tampering* and *temporal tampering*. Spatial tampering means any modification on the image frame content, and temporal tampering means any manipulation performed on the image frame sequence. In this study, temporal tampering of videos is categorized further into three types: *cropping*, *replacement*, and *insertion*. Cropping means deletion of some video frames by a malicious user. Insertion means addition of some fake video frames into the original video sequence. And Replacement means deletion of some video frames, followed by insertion of some other fake ones.

To detect spatial tampering, some random signals, called *authentication signals*, generated according to a user's key are embedded in each frame of the video. For I frames, authentication signals are embedded into the coefficients of the DCT domain. For P and B frames, authentication signals are embedded into the motion vectors in the frames. From our analysis of temporal tampering, two features are proposed in this study for use in detecting temporal tampering in the proposed method. One is the *index of the GOP of the video*. The other is the *number of the inter-coded frames in the GOP*. Both features will be embedded into the I frames of a video for the purpose of tampering detection.

In the remainder of this paper, the proposed random signal embedding method is described in Section 2, and the proposed video verification method is stated in Section 3. And in Section 4 some experimental results will be shown. Some conclusions are made in the last section.

2. Embedding Random Signals in MPEG Video

In this section, the proposed signal embedding method will be described.

2.1 Process for Embedding Random Signals in I Frames

In the proposed signal embedding process for I frames, two DCT coefficients, having the same quantization step size within the MPEG intra-quantization table of an 8×8 luminance block, are selected as a pair to embed an authentication signal. Embedding is made possible by adjusting the relative values of the coefficient pair. Since the quantization step size of the two selected DCT coefficients are equal, the relative sizes between them will not be affected even when the coefficients are re-quantized. That is, the embedded authentication signals are robust to survive moderate image recompression.

In this study, the index of the i-th GOP of the input video is denoted as G_i, and the number of the inter-coded frames of the (i-1)-th GOP is denoted as N_i. G_i and N_i are embedded into some pre-defined macroblocks of the i-th I frame in the same fashion as embedding authentication signals, as mentioned previously. In order to extract these two types of features precisely in the verification process, the proposed system duplicates them many times before embedding them to reduce the probability of misrepresentation.

***Algorithm* 1**: Signal embedding process for I frames.

Input: an I frame F, a user's key R, G_i, and N_i.

Output: a protected I frame F'.

Steps:

1. Denote the binary form of G_i as $G_i = g_1g_2\ldots g_{L_1}$, where L_1 is the length of G_i. Duplicate G_i K times to form a new binary string G'_i.
2. Denote the binary form of N_i as $N_i = n_1n_2\ldots n_{L_2}$, where L_2 is the length of N_i. Duplicate N_i K times to form a new binary string N'_i.
3. For each 8×8 luminance block B, combine the input key R and the position P of B in F to form a seed for a random number generator to produce an authentication signal S.

4. Select the two DCT coefficient which have the same quantization step size as a pair $P_1 = (C_1, C_2)$ to embed S. Before embedding S, compute $diff_1 = |C_1 - C_2|$. Embed S into P_1 according to the following two types of rules.
   - When $diff_1 \leq T_1$:

     $$\begin{cases} \textit{if } S \textit{ is odd then set } C_1 > C_2 \ \& \ |C_1 - C_2| = T_1; \\ \textit{if } S \textit{ is even then set } C_2 > C_1 \ \& \ |C_1 - C_2| = T_1; \end{cases}$$

   - When $diff_1 > T_1$:

     $$\begin{cases} \textit{if } S \textit{ is odd and } C_1 < C_2, \\ \textit{then set } C_1 = M_1 + (T_1/2) \ \& \ C_2 = M_1 - (T_1/2); \\ \textit{if } S \textit{ is even and } C_2 < C_1, \\ \textit{then set } C_1 = M_1 - (T_1/2) \ \& \ C_2 = M_1 + (T_1/2); \end{cases}$$

   where M_1 is the mean of C_1 and C_2 calculated as $M_1 = (C_1 + C_2)/2$, and T_3 is a pre-defined threshold value.
5. If the block B is one of the pre-defined blocks selected to embed G_i or N_i, then select the two DCT coefficient which have the same quantization step size as a pair $P_2 = (C_3, C_4)$ to embed a bit b of G'_i or N'_i. Before embedding b, compute $diff_2 = |C_3 - C_4|$. Embed b into P_2 according to the following two types of rules.
   - When $diff_2 \leq T_1$:

     $$\begin{cases} \textit{if } b = 1, \textit{ then set } C_3 > C_4 \ \& \ |C_3 - C_4| = T_1; \\ \textit{if } b = 0, \textit{ then set } C_4 > C_3 \ \& \ |C_3 - C_4| = T_1; \end{cases}$$

   - When $diff_2 > T_1$:

     $$\begin{cases} \textit{if } b = 1 \ \& \ C_3 < C_4, \\ \textit{then set } C_3 = M_2 + (T_1/2) \ \& \ C_4 = M_2 - (T_1/2); \\ \textit{if } b = 0 \ \& \ C_4 < C_3, \\ \textit{then set } C_3 = M_2 - (T_1/2) \ \& \ C_4 = M_2 + (T_1/2); \end{cases}$$

   where M_2 is the mean of C_3 and C_4 calculated as $M_2 = (C_3 + C_4)/2$.

2.2 Process for Embedding Random Signals in P and B Frames

Since inter-coded frames are encoded by motion compensation prediction, embedding authentication signals in the motion vectors for authenticating the fidelity of inter-coded frames can utilize efficiently the information in the video bitstream.

In the proposed signal embedding process for each P or B frame of an input video, every two non-overlapping adjacent macroblocks in a P or B frame are selected to form a pair for embedding an authentication signal. However, not each pair is proper for embedding an authentication signal. The principles of selecting proper pairs are presented in the following.

For each pair of macroblocks (MB_i, MB_j) in a P frame, there are two candidates for embedding an authentication signal. One is (H_{fi}, H_{fj}); and the other is (V_{fi}, V_{fj}), where H_{fi} and V_{fi} are the horizontal and vertical components of the forward motion vector in the macroblock MB_i, and H_{fj} and V_{fj} are the horizontal and vertical components of the forward motion vector in the macroblock MB_j. In this study, two principles of how to select a proper pair are proposed. First, motion vectors whose magnitudes are large should be selected. Secondly, the difference between the two components in a candidate pair must be small. The details of the proposed selection process are described in the following.

For each pair of macroblocks (MB_i, MB_j) in a B frame, there are four candidates: (H_{fi}, H_{fj}), (V_{fi}, V_{fj}), (H_{bi}, H_{bj}), and (V_{bi}, V_{bj}), where H_{bi} and V_{bi} are the horizontal and vertical components of the backward motion vector in the macroblock MB_i, and H_{bj} and V_{bj} are the horizontal and vertical components of the backward motion vector in the macroblock MB_j.

***Algorithm* 2**: Signal embedding process for P and B frames.

Input: a P or B frame F, and a user's key R.

Output: a protected P or B frame F'.

Steps:

1. If the input frame F is a P frame, then perform Step 1.1; otherwise, perform Step 1.2.

   1.1 For each pair (MB_i, MB_j) of non-overlapping adjacent macroblocks in F, select (H_{fi}, H_{fj}) and (V_{fi}, V_{fj}) as two candidates for embedding an authentication signal. And use the following rule to judge whether (H_{fi}, H_{fj}) is proper to embed an authentication signal:

$$\begin{cases} H_{fi} > T_2, \\ H_{fj} > T_2, \\ \left| H_{fi} - H_{fj} \right| \le 1, \end{cases} \tag{1}$$

   where T_2 is a pre-defined threshold value. If proper, then select (H_{fi}, H_{fj}) to embed an authentication signal; otherwise, use the following rule to judge whether (V_{fi}, V_{fj}) is proper to embed an authentication signal:

$$\begin{cases} V_{fi} > T_2, \\ V_{fj} > T_2, \\ \left| V_{fi} - V_{fj} \right| \le 1. \end{cases} \tag{2}$$

   If proper, then select (V_{fi}, V_{fj}) to embed an authentication signal. If the selected pair B consists of the horizontal components, then denote it as (H_i, H_j). On the contrary, if the selected pair consists of the vertical components, then denote it as (V_i, V_j).

   1.2 For each pair (MB_i, MB_j) of non-overlapping adjacent macroblocks in the input B frame F, there are four candidates: (H_{fi}, H_{fj}), (V_{fi}, V_{fj}), (H_{bi}, H_{bj}), and (V_{bi}, V_{bj}). The selection process for each pair of macroblocks (MB_i, MB_j) in a B frame is similar to the process used for P frames. Just use Equations (1) and (2), and (3) and (4) below sequentially to judge which candidate can be selected to embed an authentication signal:

$$\begin{cases} H_{bi} > T_2, \\ H_{bj} > T_2, \\ \left| H_{bi} - H_{bj} \right| \le 1. \end{cases} \tag{3}$$

$$\begin{cases} V_{bi} > T_2, \\ V_{bj} > T_2, \\ \left| V_{bi} - V_{bj} \right| \le 1. \end{cases} \tag{4}$$

   If the selected pair B consists of the horizontal components, it is denoted as (H_i, H_j). On the contrary, if the selected pair consists of the vertical components, it is denoted as (V_i, V_j).

2. Combine the input key R and the position P of the selected pair B in F to form a seed for a random number generator to produce an authentication signal S.
3. If B is (H_i, H_j) from the first step, use the following rule to embed S:

$$\begin{cases} \textit{if S is odd, then set } H_i > H_j \ \&\ |H_i - H_j| = 1; \\ \textit{if S is even, then set } H_j > H_i \ \&\ |H_i - H_j| = 1. \end{cases}$$

   On the contrary, if B is (V_i, V_j), use the following rule to embed S:

$$\begin{cases} \textit{if S is odd, then set } V_i > V_j \ \&\ |V_i - V_j| = 1; \\ \textit{if S is even, then set } V_j > V_i \ \&\ |V_i - V_j| = 1. \end{cases}$$

3. Content Verification

In this section, the proposed video content verification method will be described.

3.1 Process for Verification of Integrity and Fidelity of I Frames

Using the embedded signals as described in the last section, not only the fidelity but also the integrity of each I frame can be verified by the proposed method, since an authentication signal is embedded in each 8×8 luminance block of the I frame. This is useful for detecting spatial tampering. In addition, two pre-defined features can be extracted from each I frames to detect whether *temporal tampering* has been attempted inside the input video. One feature is the index of each GOP, denoted as G'_i; and the other is the number of the P and B frames in each GOP, denoted as N'_i.

***Algorithm* 3**: Content verification process for I frames.

Input: the i-th I frame F and a user's key R.

Output: the verified I frame and a verification report.

Steps:

1. For each 8×8 luminance block B of the input I frame F, combine the input key R and the position P of B in F to form a seed for a random number generator to produce a random signal S.
2. Use the pre-defined pair $P_1 = (C_1, C_2)$ of the DCT coefficients to verify the existence of an embedded authentication signal according to the following rule:

$$\begin{cases} \textit{if } S \textit{ is odd } \& \ C_1 \le C_2, \\ \textit{then label this block as unauthentic}; \\ \textit{if } S \textit{ is even } \& \ C_2 \le C_1, \\ \textit{then label this block as unauthentic}; \end{cases}$$

3. If the block B is one of the pre-defined blocks selected to embed the index G'_i of each GOP, then use the pre-defined pair $P_2 = (C_3, C_4)$ of the DCT coefficients to extract a bit $g'(j)$ as part of the bitstream g' according the following rule:

$$g'(j) = \begin{cases} 1, & \text{if } C_3 > C_4; \\ 0, & \text{otherwise}; \end{cases}$$

where $1 \le j \le L_1 \times K$, and L_1 is the length of the binary form of G'_i, and K is the number of copies used originally.

4. If the block B is one of the pre-defined blocks selected to embed the number N'_i of the P and B frames in each GOP, then use the pre-defined pair $P_2 = (C_3, C_4)$ of the DCT coefficients to extract a bit $n'(j)$ as part of the bitstream n' according the following rule:

$$n'(j) = \begin{cases} 1, & \text{if } C_3 > C_4; \\ 0, & \text{otherwise}; \end{cases}$$

where $1 \le j \le L_2 \times K$, and L_2 is the length of the binary form of N'_i.

5. After processing each luminance block, employ the following three steps to verify each macroblock MB.
   - 5.1 If two or more of the four luminance blocks of MB are considered unauthentic, then consider MB as suspicious.
   - 5.2 If two or more of the 4-neighbors of a suspicious MB are considered suspicious, then consider MB as unauthentic. The 4-neighbor relationship is illustrated in Figure 1.
   - 5.3 Mark unauthentic macroblocks as tampered regions.
6. After extracting all bits of g', perform majority voting to get a result as follows:

$$V(m) = \sum_{a=0}^{K-1} g'(a \times L_1 + m),$$

where $1 \le m \le L_1$. Then, reconstruct $g(m)$ by the following rule:

$$g(m) = \begin{cases} 1, & \text{if } V(m) > \dfrac{K}{2}; \\ 0, & \text{otherwise}, \end{cases}$$

where $1 \le m \le L_1$. And convert the binary string $g(m)$ into a decimal value G'_i.

7. After extracting all bits of n', perform majority voting to get a result as follows:

$$V(m) = \sum_{a=0}^{K-1} n'(a \times L_2 + m),$$

where $1 \le m \le L_2$. Then reconstruct $n(m)$ by the following rule:

$$n(m) = \begin{cases} 1, & \text{if } V(m) > \dfrac{K}{2}; \\ 0, & \text{otherwise,} \end{cases}$$

where $1 \le m \le L_2$. And convert the binary string $n(m)$ into a decimal value N'_i.

8. If some macroblocks of F are considered unauthentic, decide that spatial tampering has been attempted inside the video. Then use the following rule to determine the temporal tampering type *TTT*:

$$\begin{cases} \textit{if } (G'_i - G'_{i\text{-}1}) \ne 1 \ \& \ E_i = 0, \textit{ then TTT is cropping}; \\ \textit{if } (G'_i - G'_{i\text{-}1}) \ne 1 \ \& \ E_i \ne 0, \\ \quad \textit{then TTT is replacement}; \\ \textit{if } (G'_i - G'_{i\text{-}1}) = 1 \ \& \ E_i \ne 0, \textit{ then TTT is insertion}; \\ \textit{if } (G'_i - G'_{i\text{-}1}) = 1 \ \& \ E_i = 0 \ \& \ (N_i - N'_i) \ne 0, \\ \quad \textit{then TTT is cropping}; \end{cases}$$

where G'_i and $G'_{i\text{-}1}$ are the GOP indexes extracted from the i-th and (i-1)-th I frame, respectively; N'_i is the number of the P and B frames extracted from the i-th I frame; N_i is the number of the P and B frames in the (i-1)-th GOP of the input video; and E_i is the number of the unauthenticated frames before the i-th I frame.

3.2 Process for Verification of Fidelity of P and B Frames

In the content verification process of a P or B frame, a characteristic of inter-coded frames in the MPEG standard can be utilized to verify the fidelity of a frame, that is, the property that the number of the intra-coded macroblocks in P or B frames is usually small. If a P or B frame is manipulated illegally, then most of the macroblocks in the frame will become intra-coded ones, resulting in a great increase of the number of such macroblocks, because the illegal manipulation will cause the frame to become quite different from its reference frame. Therefore, the proportion of the number N_{intra} of the intra-coded macroblocks to the number N_{all} of the total macroblocks in the frame can be utilized for verifying the fidelity of the frame first. More specifically, if the proportion of N_{intra} to N_{all} is high, then this frame is decided to be unauthentic.

In the second verification step, the number N_{sel} of the pairs of macroblocks which satisfy the conditions specified by Equations (1) through (4) presented previously is checked. If N_{sel} is greater than a pre-selected threshold value, it means that the number of the authentication signals embedded in the frame is large enough, which may be used to verify the fidelity of the frame. On the contrary, if N_{sel} is smaller than a pre-selected threshold value, it indicates that the authentication signals embedded in the frame are insufficient for reliable fidelity verification. In this case, a method based upon a *temporal reference relation* is proposed in this study for verifying the frame. The method is presented as follows. For a P frame, if its forward reference frame is authentic, then the current frame is decided to be authentic; otherwise, the frame is decided to be unauthentic, and then the proposed verification system will mark as tampered those regions in the current frame whose corresponding regions in its forward reference frame were marked tampered.

For a B frame, the number N_f of the forward-coded macroblocks of the frame and the number N_b of the backward-coded macroblocks of the frame must be compared first to decide whether the frame is similar to its forward reference frame or to its backward reference frame. If N_f is greater than N_b, it means the frame is similar to its forward reference frame. In this case, if its forward reference is authentic, then the frame is decided to be authentic; otherwise, the frame is decided to be unauthentic, and then the proposed verification system will mark as tampered those regions in the current frame whose corresponding regions in its forward reference frame were marked tampered. On the contrary, if N_f is smaller than N_b, it means the frame is similar to its backward reference frame. In this case, if its backward reference frame is authentic, then the frame is decided to be authentic; otherwise, the frame is decided to be unauthentic, and then the proposed verification system will mark as tampered those regions in the current frame whose corresponding regions in its backward reference frame were marked tampered. A reason for using this method is that tampering, taking place in a brief amount of time, should occur within neighboring regions in a series of frames that are similar to each other. If not so, tampering should be easily detected.

***Algorithm* 4**: Content verification process for P and B frames.

Input: a P or B frame F and a user's key R.

Output: the verified P or B frame.

Steps:

1. Count the number N_{intra}, N_f, and N_b of the intra-coded, forward-coded, and backward-coded macroblocks of the input P or B frame F, respectively. In the mean time, count the number N_{all} of all the macroblocks of F. Judge whether the input frame is authentic by the following rule:

   $$\begin{cases} \textit{if } (N_{intra}/N_{all}) > T_3, \\ \quad \textit{then the input frame is unauthentic}; \\ \textit{otherwise, continue the next step}; \end{cases}$$

   where T_3 is a pre-defined threshold value.

2. For each pair of non-overlapping adjacent macroblocks of F, use the selection process presented in Section 2.2 to select the pairs used to embed authentication signals. If the selected pair consists of the horizontal components, it is denoted as (H_i, H_j). On the other hand, if the selected pair consists of the vertical components, it is denoted as (V_i, V_j). And then count the number N_{sel} of the selected pairs. If N_{sel} is larger than a pre-defined threshold value T_6, perform Step 3; otherwise, perform Step 4.

3. For each selected pair B, combine the input key R and the position P of B in F to form a seed for a random number generator to produce a random signal S. If the selected pair B in Step 2 is (H_i, H_j), then examine the relation between S and (H_i, H_j) and count the number N_w of unauthentic pairs according to the following rule:

   $$\begin{cases} \textit{if } S \textit{ is odd } \& \ H_i \leq H_j, \textit{ then } N_w = N_w + 1; \\ \textit{if } S \textit{ is even } \& \ H_j \leq H_i, \textit{ then } N_w = N_w + 1. \end{cases}$$

   If the selected pair in Step 2 is (V_i, V_j), then examine the relation between S and (V_i, V_j) and count the number N_w of unauthentic pairs according to the following rule:

   $$\begin{cases} \textit{if } S \textit{ is odd } \& \ V_i \leq V_j, \textit{ then } N_w = N_w + 1; \\ \textit{if } S \textit{ is even } \& \ V_j \leq V_i, \textit{ then } N_w = N_w + 1. \end{cases}$$

   After verifying each selected pair, decide whether the frame F is authentic by the following rule:

   $$\begin{cases} \textit{if } (N_w/N_{sel}) > T_4, \\ \quad \textit{then the input frame is unauthentic}; \\ \textit{otherwise, the input frame is authentic}, \end{cases}$$

   where T_4 is a pre-defined threshold value.

4. If the input frame F is a P frame, then perform Step 4.1; on the other hand, if F is a B frame, then perform Step 4.2.

   4.1 If the forward reference frame R_f of F is considered authentic in the previous verification process, then decide F to be authentic; otherwise, unauthentic and mark as tampered those regions in F whose corresponding regions in R_f were marked tampered.

4.2 First, compare N_f with N_b. If N_f is larger than N_b, perform Step 4.2.1; otherwise, Step 4.2.2.

4.2.1 If the forward reference frame R_f of F is considered authentic in the previous verification process, then decide F to be authentic; otherwise, unauthentic and mark as tampered those regions in F whose corresponding regions in R_f were marked tampered.

4.2.2 If the backward reference frame R_b of F is considered authentic in the previous verification process, then decide F to be authentic; otherwise, unauthentic and mark as tampered those regions in F whose corresponding regions in R_b were marked tampered.

4. Experimental Results

In our experiments, a video with frame size 352×240 was used as the input. Four frames of the input video are shown in Figure 2. The four corresponding frames of the resulting video after the proposed signal embedding process was performed are shown in Figure 3. The PSNR values of them are shown in Table 1, from which we can see that the authentication signals can be embedded into MPEG videos imperceptibly by applying the proposed method. Figure 4 shows certain modification results of the four frames by commercial software. Figure 5 is the verification result of these modified frames, in which the yellow regions represent the tampered regions. Moreover, the tampering was recognized to be of the type of spatial tampering. From these figures we can see that the tampered regions can be identified efficiently and the tampering types can be recognized correctly by the proposed method.

5. Conclusions

In this paper, a method for fidelity and integrity verification of MPEG videos has been proposed. Both spatial and temporal tamperings can be detected by the proposed method. Spatial tampering is detected by a scheme of embedding random signals generated according to a key into the frequency coefficients as well as the motion vectors of a video. Any malicious modification performed on the image frames of the video will destroy the embedded signals; therefore, spatial tampering can be detected by the proposed method. On the other hand, temporal tampering is detected by a scheme of embedding the temporal information of a video into the I frames of it. Any malicious manipulation performed on the video frame sequence will change the embedded temporal information; therefore, temporal tampering can be detected by the proposed method. Good experimental results prove the feasibility of the proposed methods.

6. References

[1] Jana Dittmann et al., “Content-based Digital Signature for Motion Pictures Authentication and Content-Fragile Watermarking,” *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, Florence, Italy, vol. 2, pp. 209-213, June 1999.

[2] M. Schneider and S. F. Chang, “A Robust Content Based Digital Signature for Image Authentication,” *Proceedings of IEEE International Conference on Image Processing*, Lausanne, Switzerland, vol. 3, pp. 227-230, Sept. 1996.

[3] D. A. Winne et al., “Spatial Digital Watermark for MPEG-2 Video Authentication and Tamper Detection,” *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, USA, vol. 4, pp. 3457-3460, May 2002.

[4] B. G. Mobasseri et al., “Content Authentication and Tamper Detection in Digital Video,” *Proceedings of IEEE International Conference on Image Processing*, Vancouver, BC, Canada, vol. 1, pp. 458-461, Sept. 2000.

[5] P. Yin and H. H. Yu, “A Semi-fragile Watermarking System for MPEG Video Authentication,” *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, USA, vol. 4, pp. 3461-3464, May 2002.

[6] R. Du and J. Fridrich, “Lossless Authentication of MPEG-2 Video,” *Proceedings of IEEE International Conference on Image Processing*, New York, USA, vol. 2, pp. 893-896, Sept. 2002.

Table 1 The PSNR values of the resulting video.

	f_0 (I)	f_1 (B)	f_2 (B)	f_3 (P)
PSNR	34.0	34.1	34.1	34.0

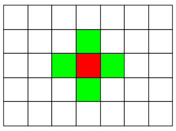
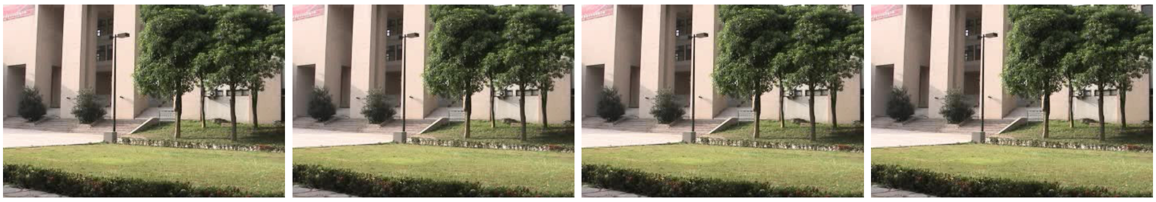
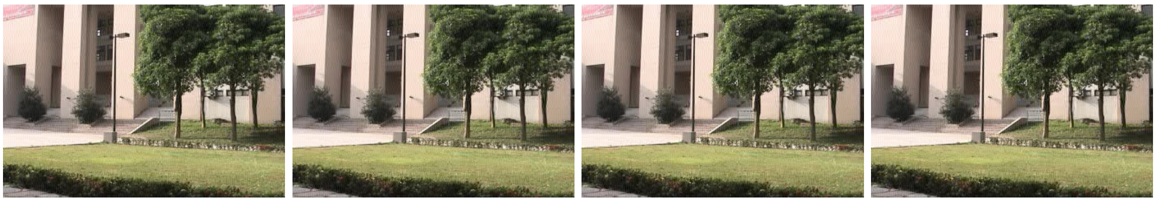


Figure 1 Illustration of 4-neighbor relationship (The four green macroblocks are the 4-neighbors of the red one).



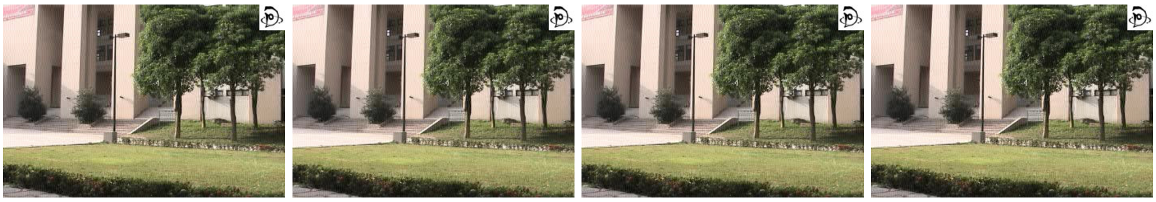
(a) (b) (c) (d)

Figure 2 Four frames of the original video. (a) The first frame (I frame). (b) The second frame (B frame). (c) The third frame (B frame). (d) The 4th frame (P frame).



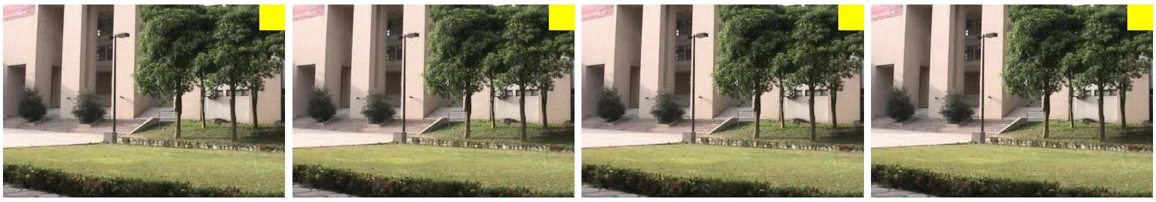
(a) (b) (c) (d)

Figure 3 Four frames of the resulting video. (a) The first frame (I frame). (b) The second frame (B frame). (c) The third frame (B frame). (d) The 4th frame (P frame).



(a) (b) (c) (d)

Figure 4 Four frames of the resulting video that has been modified. (a) The first frame (I frame). (b) The second frame (B frame). (c) The third frame (B frame). (d) The 4th frame (P frame).



(a) (b) (c) (d)

Figure 5 Four frames of the verified video of a modified video. (a) The first frame (I frame). (b) The second frame (B frame). (c) The third frame (B frame). (d) The 4th frame (P frame).