Video Authentication with Self Recovery

Mehmet U. Celik[a], Gaurav Sharma[b], A. Murat Tekalp[a] and Eli Saber[b]

[a]University of Rochester, Rochester, NY, USA
[b]Xerox Corporation, Webster, NY, USA

ABSTRACT

Digital video has become increasingly susceptible to spatio-temporal manipulations as a result of recent advances in video editing tools. In this paper, we propose a secure and flexible fragile digital video authentication watermark which also enables the self-recovery of video content after malicious manipulations. In the proposed block-based method, the watermark payload of a block is composed of two parts: authentication and recovery packets. The authentication packet is a digital signature with a special structure and carries the spatio-temporal position of the block. The digital signature guarantees the authenticity and integrity of the block as well as the recovery packet, whereas the localization information prevents possible cut& paste attacks. On the other hand, the recovery packet contains a highly compressed version of a spatio-temporally distant block. This information enables the recovery of the distant block, upon detection of tampering by its authentication packet. A spatio-temporal interleaving scheme and a simple multiple description coding mechanism increase the probability of self recovery by diffusing recovery information throughout the sequence. Finally, watermark payload is embedded by least significant bit modulation.

Keywords: Fragile watermark, video authentication, self recovery, LSB modulation, digital signatures, multiple description coding

1. INTRODUCTION

Using commonly available hardware and software, it is now feasible to perform digital video editing not only to insert, delete or replace groups of frames but also to insert or delete objects from those frames without introducing visible artifacts. The ease and extent of such manipulations highlights the need for authentication and integrity verification mechanisms in applications such as video for court evidence. Potential security loopholes of shared information networks, e.g. Internet, on which digital video is stored and distributed further underscore this need.

Multimedia integrity and authenticity can be guaranteed through the use of *digital signatures* and/or watermarks. A *digital signature* is a data string which associates a message (in digital form) with some originating entity.[1] Digital watermarking[2–4] may be utilized in general to verify authenticity and integrity of multimedia content. The use of watermarks for digital video authentication typically affords additional functionality such as spatio-temporal tamper localization and direct embedding of the authentication information in the video data. Authentication watermarks can be classified as either *fragile* or *semi-fragile*. Fragile watermarks, as the name implies, are designed to identify any alteration of pixel values. On the other hand, semi-fragile watermarks try to differentiate between content preserving (non-malicious) processes, e.g. transcoding or frame-rate changes, and malicious manipulations, e.g. removal objects from a scene or change of sequence of events. The use of watermarks for authentication may further facilitate the recovery of tampered segments of video via *self-embedding*.[5] *Self-embedding* refers to the process where a compressed copy of the image/video is embedded into itself during watermarking. During watermark verification, this embedded copy is extracted to reconstruct portions of the image/video where tampering is detected. This latter process is known as *self-recovery*.

In this paper, a fragile video authentication watermark with spatio-temporal localization and self-recovery properties is presented. The proposed method builds on recent work by Fridrich,[6] where an elegant solution

Send correspondence to M.Celik: E-mail: celik@ece.rochester.edu, Telephone: 1 585 275-8122, Address: Electrical and Computer Engineering Department, University of Rochester, Rochester, NY, 14627-0126, USA, WWW: http://www.ece.rochester.edu/projects/iplab

Security and Watermarking of Multimedia Contents IV, Edward J. Delp III, Ping Wah Wong, Editors, Proceedings of SPIE Vol. 4675 (2002) © 2002 SPIE · 0277-786X/02/$15.00

to image authentication is proposed. Extending this solution to digital video, our algorithm operates on pixel blocks of video frames. A similar digital signature structure is used to combine the block hash with block positioning information. Moreover, we incorporate self-recovery properties proposed earlier for images[5] and video.[7] We improve the effectiveness of both methods by utilizing a multiple description coding scheme and a novel interleaving strategy. Furthermore, we exploit the key frame concept in the context of recovery for compressed video where watermarking channel capacity is low. As a result, the proposed algorithm not only improves the current state-of-the-art methods, but also provides a flexible framework for further improvements.

This paper is organized as follows: In Section 2 we summarize the relevant previous work on image and video authentication marks. Proposed general framework is explained in Section 3. An instantiation of the algorithm, together with possible attack scenarios and results are presented in Section 4. Finally, conclusions are drawn in Section 5.

2. BACKGROUND

2.1. Image Authentication Watermarks

Early work on authentication of visual content focussed on still images based on the immediate threat due to the relative simplicity of tampering on still images. In this section, we will review some of the relevant work on fragile still image authentication watermarks.[8–12]

A well known algorithm among public key fragile watermarks is Wong's scheme,[10] which embeds a digital signature of most significant bits of a block of the image into least significant bits of the same block. Despite the elegance of the algorithm and cryptographic security of the digital signatures, its blockwise independence was exploited by Holliman and Memon with a counterfeiting attack.[13] The attacker constructs a vector quantization codebook using blocks from a set of watermarked images. The image to be counterfeited is then approximated using this codebook. Since each block is authenticated by itself, the counterfeit image appears authentic to the watermarking algorithm. Recently, Fridrich[6] presented a modification of the Wong's scheme which prevents such attacks on the scheme. Since our proposed scheme builds on this idea let us explain it in detail.

Counterfeit Resistant Authentication

In general, a digital signature is a public-key encrypted version of the hash of a data block. $S = \mathcal{E}(\mathcal{H}(D), Key)$ where $\mathcal{E}$ and $\mathcal{H}$ denote encryption and hashing functions, and D is the data to be authenticated; resulting digital signature is denoted by S. During verification $\mathcal{H}(D)$ is compared with $\mathcal{D}(S, Key)$, where $\mathcal{D}$ is the decryption function corresponding to $\mathcal{E}$.

In Wong's scheme,[10] digital signature S is computed using the MSBs of a pixel block. Resulting signature is embedded in LSBs of the same block.

$$S = \mathcal{E}(\mathcal{H}(MSB), Key); \tag{1}$$

The vulnerability of the scheme to counterfeiting attacks[13] can be eliminated by customizing the digital signature as proposed by Fridrich.[6] In particular, spatial location of the block (X,Y) together with a unique image ID is embedded in the signature.

$$S = \mathcal{E}(\mathcal{H}(MSB) \oplus [ID||X||Y||ID||X||Y||...], Key); \tag{2}$$

where $\oplus$ and $||$ denote bitwise exclusive or operation and concatenation, respectively. During watermark verification, embedded image ID and block position information is extracted.

$$[ID||X||Y||ID||X||Y||...] = Decrypt(S, Key) \oplus Hash(MSB) \tag{3}$$

Symmetry of the extracted information together with the consistency of the image ID throughout the image and correctness of block positions authenticate a block, preventing counterfeiting attacks. If pixel values within a block are altered, signature decryption and/or hashing steps will alter extracted information which will indicate tampering.

2.2. Video Authentication Watermarks

Authentication of video is very similar to that of still images. In particular, we can evaluate a video as a sequence of still images where each image (frame) is authenticated individually. Any tampering within frames will be detected by the image verification scheme. In addition to spatial manipulations, a common class of attacks on digital video is re-indexing attacks, where the sequence of events is tampered. Since any ordering of authentic frames is not an authentic video sequence, the temporal location of frames should also be verified by the watermark. Dittman et al.[14] addresses this problem by embedding the SMPTE time code in each frame. Any alterations in the sequence of events is detected by checking the timing information. Mobasseri et al.[7] uses the concept of frame pairs to verify the integrity of the video sequence. In particular, a coarsely quantized and encrypted version of a frame is inserted into the LSBs of the other frame in the frame pair. Re-indexing could result in corruption of frame pair correspondences indicating a manipulation. In the absence of temporal manipulations, the encrypted copy of the frame also acts as self-recovery information, i.e. spatial manipulations can be undone using the extracted copy. A major disadvantage of this method is the pseudo-random permutation matrix used to identify the frame pairs. Removal/insertion of frames at two different temporal neighborhoods completely impairs the synchronization of frame pairs, resulting in the loss of verification and self-recovery abilities. Reverse tracing employed to regain synchronization when frame removals are concentrated in a single temporal neighborhood is not sufficient in such cases.

3. PROPOSED ALGORITHM

In this section, a novel solution to digital video authentication is outlined. In particular, we propose a block-based algorithm which operates on pixel blocks of each frame. Our scheme utilizes a digital signature similar to that of Fridrich's[5] and a novel self-recovery mechanism. The block-based solution has two advantages over full-frame solutions. It is capable of pin-pointing manipulated regions while validating the rest of the frame. Moreover, all verified regions of the frame carry trustworthy information which can be used to improve the recovery quality of the manipulated regions.

3.1. Watermark Embedding

Watermarking process can be considered in two phases: preparation of the watermark payload and payload insertion. In the proposed block-based scheme watermark payload of a block consists of two parts; authentication and recovery information (4). During verification, authentication information is used to verify the integrity of a block; whereas tampered blocks are reconstructed using the recovery information extracted from trustworthy, i.e. verified, blocks.

$$Payload = [AuthenticationPacket || RecoveryPacket] \tag{4}$$

3.1.1. Self-Recovery and Recovery Information

An approximation to the original content can be recovered at the watermark detector after limited manipulations using *self-embedding*[5] watermarks. In general, a compressed copy of the video is inserted into itself and during recovery phase tampered regions are reconstructed using this embedded copy, i.e. *self-recovery.*

Key frame concept

Despite the elegance of the self-embedding concept, it is not always possible to embed a full description of the content into itself. In most cases, watermarking channel capacity is much lower than the bandwidth required by the description*. Especially when the host data is highly compressed, an efficient representation of the content is necessary. Video summarization has already been established using *key frames* in various applications, e.g. video indexing. Here, we re-introduce this concept in the context of self-recovery. In particular, we propose self-embedding only selected key frames in instances when the watermarking channel capacity is low. Algorithms[15] exist for automatic extraction of key frames from video sequences.

Multiple Description Coding

A property of encryption and hashing functions utilized in digital signatures in authentication watermarks is

*A known exception is possible with high capacity uncompressed host data and lossy compression of the description

the *avalanche effect*. Avalanche effect refers to the significant change at the output of the function, when a small change (even a single bit) occurs at the input. Despite being desirable for the security of the digital signatures against cryptanalytic attacks, this property results in the complete loss of the encrypted data in case of a tampering. In that respect, channel losses faced by the encrypted descriptions of the key frames is similar to packet losses encountered in transmission over information networks. Thus, we may improve the self-recovery quality of the system using multiple description coding,[16–18] which have been proposed for transmitting images and video over information networks. In particular, it is possible to recover a low quality version of the key frame even-if some of the recovery packets are lost during tampering. The multiple description coding schemes provide a graceful trade-off between resilience against packet losses and recover quality.

Interleaving

The analogy between packet losses on an information network and loss of payload data due to manipulations of the video content can be further extended to the error characteristics. In a given information network, packet losses typically occur in bursts when a particular section of the network is congested. Likewise, modifications made on a video sequence are not arbitrary and are generally confined in certain spatio-temporal neighborhoods. Thus, the interleaving schemes, which provide better error resilience on information networks, can also be utilized during self-embedding.

A simple interleaving strategy that yields the maximum distance between two consecutive samples of the original signal can be obtained by linear congruence.
Let $x \in \{0, 1, 2, 3, ..., N-1\}$ and $y \in \{0, 1, 2, 3, ..., N-1\}$ denote the original and interleaved index values, where N is the total number of samples.

$$y = mx \qquad (mod N) \tag{5}$$

gives an interleaving distance of m. We require m and N be relatively prime for the transformation to be one-to-one, and $N > 2m$ for maximum distance between consecutive samples. We may further include a pseudo random shift, $s < N$ as a secret key.

$$y = mx + s \qquad (mod N) \tag{6}$$

This one dimensional interleaving scheme can be applied to blocks in a video sequence as well. Basically, the interleaving can be applied to the 1-D sequence obtained by scanning the block indices in a frame in zig-zag order and concatenating the resulting sequences across frames in accordance with the frame sequencing. However this does not guarantee a minimum distance between adjacent blocks in three dimensions. A generalized form of the scheme is to employ interleaving in all three dimensions. Let $\mathbf{x}$, denote the position vector of a block in three dimensions. Interleaving equation can be rewritten as,

$$\mathbf{y} = \mathbf{T}\mathbf{x} + \mathbf{s} \qquad (mod \mathbf{N}) \tag{7}$$

where the addition and modulo operations are component-wise. Since it is harder to guarantee one-to-one and minimum distance properties in this general form, we may simplify the case by assuming independent interleaving along each dimension. We should then require $\mathbf{T}$ be diagonal and N_i and T_{ii} be relatively prime with $N_i > 2T_{ii}$. Once the appropriate interleaving pattern is calculated, we can pair the recovery packets, i.e. individual descriptions of a block, with the corresponding carrier blocks.

3.1.2. Integrity Verification and Authentication Information

Authentication information in the watermark payload has two functions: checking the integrity of pixel values and verifying the location of the block. We will use the idea of encoding block information in the digital signature as in (2). That is, the authentication information, the digital signature of the block, is formed; first by hashing the most significant bits of the block and the recovery packet(R_{Packet}), then by XORing it with the localization information (L_{Packet}) and encrypting the result.

$$S = \mathcal{E}(\mathcal{H}(MSB||R_{Packet}) \oplus L_{Packet}, Key); \tag{8}$$

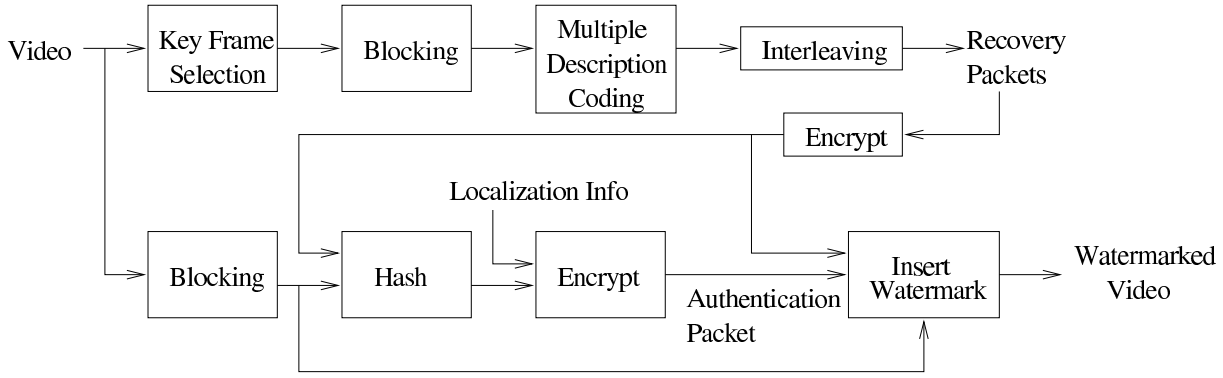

Figure 1: Watermark embedding process

During watermark verification localization information (L_{Packet}) is obtained by:

$$L_{Packet} = Decrypt(S, Key) \oplus Hash(MSB||R_{Packet}) \tag{9}$$

In order to prevent any cut& paste attacks, localization information contained in a block should uniquely identify the block. Thus, L_{Packet} consists of following fields: A unique *video ID* that differentiates between different sequences, *Frame Number*, *Block Horizontal and Vertical Position* that differentiates blocks in a particular sequence. In addition to a unique ID, we may incorporate some parameters of the sequence for further tamper resistance against spatial and temporal cropping attacks. In particular we shall append: *Frame Size*, and the *Total Number of Frames*.

3.1.3. Watermark Insertion by LSB Modification

An elementary method of data hiding in digital images and video is the least significant bit(LSB) modification. In LSB modification, the payload information is embedded in the content by modifying the LSB of pixel values or transform coefficients. Despite its simplicity and high data hiding capacity, this method is not robust to any of the signal processing operations encountered in practice, i.e. embedded information is lost upon processing of the content. Nonetheless, in fragile authentication watermarks where any processing of the content should, by definition, invalidate the watermark, LSB modification is commonly employed.[10] Moreover, this method can be applied in compressed domain such as Motion JPEG or MPEG with partial decoding. In Motion JPEG, each frame is compressed using JPEG still image compression algorithm. JPEG consists of three basic steps: block DCT transform, quantization, and entropy coding, Huffman coding in particular. In JPEG domain, LSB modification can be applied to non-zero DCT coefficients by multiples of quantization step sizes. As modifications are amplified by the quantization step size, visibility considerations severely limit the data hiding capacity in JPEG. MPEG is a similar standard that incorporates motion compensation. MPEG uses JPEG like DCT-based encoding for intra-frame coding and inter-frame residual coding. A modification of the scheme proposed by Hartung et al.[19] can be utilized for data hiding for MPEG compressed video.

Any of the preceding methods can be utilized for inserting the watermark payload, depending on the compression type of the host. Depending on the resulting watermark channel capacity, block sizes and frequency of the key frames should be adjusted accordingly.

3.2. Watermark Verification

During verification, first the watermark payload is extracted from the host. This is followed by extraction of the localization information using (9). If the pixel values within the block have not been tampered with, the extracted information will match the embedded version. Otherwise, the extracted fields will be arbitrary.

Once the localization information is extracted from multiple blocks, a majority voting scheme is employed to determine the dominant *video ID, frame size* and *total number of frames* in the sequence. Likewise, in each

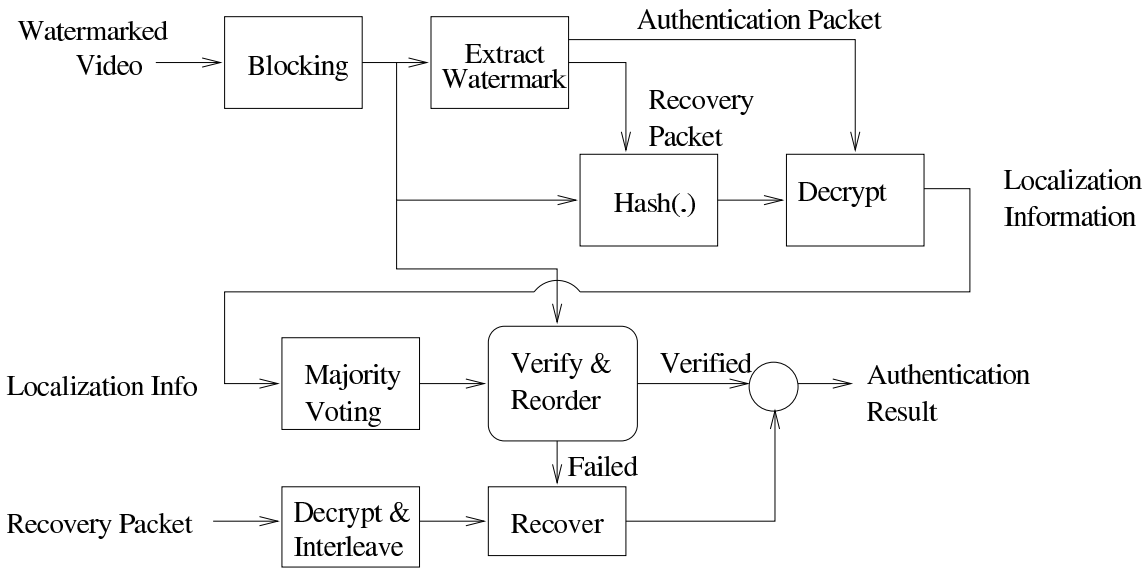

Figure 2: Watermark verification process

frame, a voting (by constituent blocks) determines the *frame number*. Note that if there are no manipulations, the voting results will be unanimous. By accepting the voting results as the original values, we validate each block by checking the *video ID, frame size* and *total number of frames*. Tampered regions or unauthentic frames that do not match localization information are marked. Furthermore, we verify the position of the blocks in the video sequence and re-order them, if necessary. This step recovers from any temporal attacks, e.g. re-indexing to change sequence of events.

The blocks marked as tampered or the missing frames have to be reconstructed from the embedded copy of the content. Therefore, the recovery packets corresponding to those blocks are decrypted and decoded. Note that if an authentic recovery packet is found; its position, thus the location of the corresponding block, is known. So the interleaving step can be easily reversed. As a further improvement, when a block and all of its descriptions are missing, we simply copy the block from the previous frame as an error concealment strategy.

4. EXPERIMENTAL RESULTS

4.1. Implementation Details

In order to demonstrate the effectiveness of the proposed solution, we realized a simple implementation of the scheme. This implementation operates on uncompressed YUV 4:2:0 video format, with simple key frame selection and a multiple description coding scheme. A list of implementation specifications are given below:

Block size:	16x16
Data hiding algorithm:	Pixel LSB modification
Hiding capacity per block:	384 bits (256, 64, 64 for Y, U, V respectively)
Authentication Packet:	128 bits
Recovery Parameters:	
Recovery Packet:	256 bits
Key frames:	Every other frame (2:1 sampling)
Compression scheme:	Fixed length DCT based (2 bpp)
Recovery packets per block:	2 (1 bpp each)
MDC scheme:	DCT coefficient grouping (even and odd coefficients)
Interleaving parameters (m, N):	7, 18; 7, 22; 13, 100 (vertical; horizontal; temporal)

Details of the authentication packet is as follows:

Authentication Packet:	128 bits
Hash Algorithm:	MD5 (128 bits)
Encryption Algorithm:	AES (128 bits)
Video ID:	32 bits
Frame No:	32 bits
Number of Frames:	32 bits
Frame Width:	8 bits (in multiples of block width)
Frame Height:	8 bits (in multiples of block height)
Block Position (V):	8 bits (in multiples of block width)
Block Position (H):	8 bits (in multiples of block height)

Multiple Description Coding

We utilize a JPEG like block compression algorithm proposed by Fridrich et al.[5] We compress only the luminance component of the frame, since it provides the necessary reconstruction quality in most applications. In particular, for each 8×8 block of a frame, we: *i)* perform DCT (discrete cosine transformation); *ii)* quantize DCT coefficients by a JPEG quantization table; *iii)* encode each coefficient using a fixed bit allocation table.

The JPEG quantization matrix and bit allocation tables used are as follows:

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix} \quad L = \begin{bmatrix} 7 & 7 & 6 & 5 & 4 & 3 & 2 & 0 \\ 7 & 6 & 5 & 5 & 4 & 2 & 0 & 0 \\ 6 & 5 & 5 & 4 & 3 & 0 & 0 & 0 \\ 5 & 5 & 4 & 3 & 0 & 0 & 0 & 0 \\ 4 & 4 & 3 & 0 & 0 & 0 & 0 & 0 \\ 3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Among various multiple-description coding algorithms, we employ a simple example.[16] In this scheme, the two descriptions of an image block consists of DCT coefficients with even and odd indices, only exception being the DC coefficient which is included in both of the descriptions. This scheme together with fixed length DCT coding provide a simple yet effective representation.

4.2. Experimental Results

4.2.1. Self-Recovery Quality

We have tested the self-recovery of our method using the first frame of the news sequence. Frame size is 352×288 pixels and only Y component is compressed with roughly 2 bpp. Despite being less efficient than most conventional image compression algorithms, the proposed scheme is a low complexity solution with a fixed rate guarantee for small image patches (16×16 blocks). PSNR when both of the descriptions are present is 33.1dB. This is about 2 dB lower than JPEG compression with quality factor 50. When only one description can be recovered, the resulting PSNR's are 22.9 dB and 23.8 dB. Thus, we expect the quality of a recovered block to be between 23dB and 33dB in terms of PSNR.

4.2.2. Temporal Attacks

In this section, the performance of our algorithm under temporal manipulations is highlighted. A common non-malicious manipulation which the video sequences are subject to is frame rate conversion by frame dropping or insertion. For instance, a 30 frames/sec video can be converted to 25 frames/sec by dropping every sixth frame. In this case, the algorithm detects missing frames from the extracted frame numbers and can either reconstruct the missing frames or simply indicate the change (see Fig. 5(a)). Moreover, this attack may be combined with any other spatial or temporal attack, and will not impair the functionality of the watermark.

A major malicious temporal manipulation is the re-indexing attack, where the sequence of events is altered by shuffling groups of frames as seen in Fig. 5(b). Once more the extracted frame numbers of each frame enables the re-ordering of frames, and thus the recovery of the original video sequence.

Figure 3: Self-recovery results: (a) Original frame, (b) Reconstructed frame with both descriptions

Figure 4: Multiple description coding results: (a) First description only, (b) Second description only

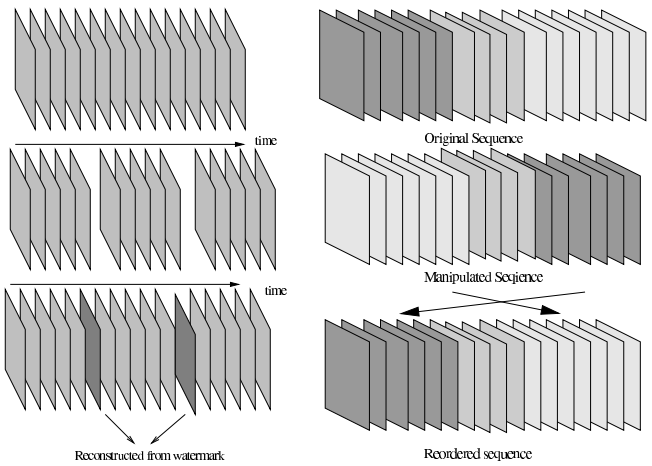

Figure 5: Illustration of temporal attacks and recovery: (a) Frame rate change, (b) Re-indexing attack

Figure 6: Spatial attack results: (a) Manipulated frame, (b) Detection result, (c) Recovered frame

4.2.3. Spatial Tamper Localization and Recovery

In addition to temporal tamper detection, our algorithm can also detect manipulations of pixel values within a given frame. First, tampered regions are localized and marked using embedded authentication packets. For instance, in Fig. 6(a) one of the dancers is removed from the scene. Although authentic blocks from the same frame are used during this manipulation, the algorithm successfully determines the manipulated regions using localization information (Fig. 6(b)). Moreover, all of the recovery packets for the tampered blocks are found and the original scene is restored, as seen in Fig. 6(c).

The example illustrates the tamper localization accuracy of our algorithm, where regions are determined down to a 16×16 resolution. This resolution is limited by the capacity of the watermarking channel and the required recovery quality. For instance, the accuracy of the scheme can be improved by utilizing additional LSB planes for watermark embedding, trading embedding distortion with localization accuracy. On the contrary, resolution of the scheme operating on compressed video streams will be much lower, due to the limited watermarking channel capacity.

5. CONCLUSION AND DISCUSSIONS

In this paper, we presented a video authentication watermark with self recovery capabilities. The scheme allows authentication of video data with spatial and temporal tamper-localization and also self-recovery of a reduced quality original from a tampered version that has been subjected to spatial and/or temporal (re-sequencing) manipulations. The presented experimental results illustrate the effectiveness of the scheme in localizing spatial/temporal manipulations and recovering the original video data from the embedded self-recovery information.

Though the scheme was illustrated in the uncompressed domain, it may be extended to compressed domain by modifying the watermark embedding process and adjusting algorithm parameters. The experimental implementation presented here utilized simple compression and multi-description coding schemes in order to keep the complexity to a minimum. The performance of the method can be improved through the use of more advanced building blocks. For instance, variable length coding algorithms with truncation can be employed instead of the rather inefficient fixed length coding scheme.

A major disadvantage of the proposed solution is the multiple passes required to diffuse the recovery information throughout the sequence. Even if the diffusion is limited in a certain group of frames, the algorithm has substantial delay and buffering requirements, making it unsuitable for real-time applications. However, when watermark embedding and verification (not necessarily re-ordering) need to be performed in real-time, the scheme can be stripped down to a simple authentication algorithm without self-recovery.

The proposed watermarking scheme enables a trade-off between various performance criteria. In particular, tamper localization resolution, security, recovery quality and embedding distortion of the system can be traded-off with each other in order to facilitate the particular requirements of a given application. Given a watermarking

algorithm, the embedding distortion increases with the increasing capacity of the watermarking channel. For instance, during watermarking by LSB modulation, we may utilize additional bit-planes. This increases both the capacity of the watermark channel and the distortion caused by the watermark embedding process. Likewise, we may operate on smaller pixel blocks, which effectively increases the tamper localization resolution of the system. If the watermarking channel capacity is fixed, this implies use of smaller recovery packets and/or shorter digital signatures. Thus, the improvement comes in the expense of the security and/or the recovery quality of the system. Here, we assumed that longer digital signatures provide better security against cryptanalytic attacks. A similar analysis can be performed on the experimental instantiation the algorithm with respect to earlier authentication watermarks without self-recovery properties. In essence, the self-recovery property of the system comes in the expense of the localization resolution, but not the security of the system, which depends on well-known cryptographic algorithms. In conclusion, the proposed scheme is a secure and flexible solution to the fragile video authentication problem.

REFERENCES

1. A. Menezes, P. van Oorchot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Florida, USA, 1997.
2. R. L. G.C. Langelaar, I. Setyawan, "Watermarking digital image and video data," *IEEE Signal Processing Magazine* **17**, pp. 20–46, September 2000.
3. F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE* **87**, pp. 1079–1107, July 1999.
4. M. Swanson, M. Kobayashi, and A. Tewfik, "Multimedia data-embedding and watermarking technologies," *Proceedings of the IEEE* **86**, pp. 1064–1087, June 1998.
5. J. Fridrich and M. Goljan, "Images with self-correcting capabilities," in *Proceedings of IEEE International Conference on Image Processing*, (Kobe, Japan), Oct 1999.
6. J. Fridrich, "Security of fragile authentication watermarks with localization," *Proceedings of SPIE Security and Watermarking of Multimedia Contents IV* **4675**, Jan 2002.
7. B. Mobasseri and A. Evans, "Content-dependent video authentication by self-watermarking in color space," *Proceedings of SPIE Security and Watermarking of Multimedia Contents III* , Jan 2001.
8. G. Friedman, "The trustworthy digital camera: Restoring credibility to the photographic image," *IEEE Trans. on Consumer Electronics* **39**, pp. 905–910, November 1993.
9. M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *Proceedings of IEEE International Conference on Image Processing*, pp. 680–683, (Santa Barbara, CA, USA), Oct 1997.
10. P. Wong, "A public key watermark for image verification and authentication," in *Proceedings of IEEE International Conference on Image Processing*, pp. 425–429, (Chicago, USA), October 4-7, 1998.
11. J. Fridrich, M. Goljan, and A. Baldoza, "New fragile authentication watermark for images," in *Proceedings of IEEE International Conference on Image Processing*, (Vancouver, Canada), September 10-13, 2000.
12. M. Celik, G. Sharma, E. Saber, and A. Tekalp, "A hierarchical image authentication watermark with improved localization and security," in *Proceedings of IEEE International Conference on Image Processing*, (Thessaloniki, Greece), Oct 2001.
13. M. Holliman and N. Memon, "Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes," *IEEE Transactions on Image Processing* **9**, pp. 432–441, March 2000.
14. J. Dittman and et al., "Combined video and audio watermarking: embedding content information in multimedia data," *Proceedings of SPIE Security and Watermarking of Multimedia Contents II* **3971**, Jan 2000.
15. A. Ferman, A. Tekalp, and R. Mehrota, "Effective content representation for video," in *Proceedings of IEEE International Conference on Image Processing*, (Chicago, IL, USA), Oct 1998.
16. Y. Wang, M. Orchard, and A. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proceedings of IEEE Signal Processing Society Workshop on Multimedia Signal Processing*, (Princeton, NJ, USA), Jun 1997.
17. V. Vaishampayan and S. John, "Interframe balanced multiple description video compression."

18. V. Goyal, "Multiple description coding: compression meets the network," *IEEE Signal Processing Magazine* **18**, pp. 74–93, Sept 2001.
19. F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Processing* **66**, pp. 283–301, May 1998.