NEW INTRA-VIDEO COLLUSION ATTACK USING MOSAICING

Gwenaël DOERR and Jean-Luc DUGELAY

Department of Multimedia Communications
Eurécom Institute, Sophia-Antipolis, FRANCE
http://www.eurecom.fr/∼image

ABSTRACT

Recent efforts for watermarking digital video extend the results obtained for still image watermarking. As a result, most of the proposed algorithms rely on a frame-by-frame approach. Such an adaptation leads to unreliable algorithms in terms of security. The goal of this article is to stress the problem of collusion when digital watermarked data is distributed at large scale and especially intra-video collusion in the context of video. Three simple collusion attacks are described before being evaluated on two alternative video watermarking algorithms based on the spread spectrum technique. Finally, some experimental results are presented confirming the danger of intra-video collusion and some perspectives are discussed.

1. INTRODUCTION

Digital watermarking is being researched for ten years now and is mostly related with copyright protection issues. It is often regarded as a second line of defense once digital data has been left in clear after decryption. This technology basically embeds in digital data a robust and invisible watermark, which encodes the rights associated with those data. The watermark is inherently tied to the content and survives D/A conversion: it fills the *analog gap* created by the decryption of digital data. If digital watermarking has been extensively studied for still images at the beginning, watermarking other types of digital multimedia data is currently being investigated and video data is one of those *new objects* of interest.

There is indeed an increasing need for copyright protection with digital video data. Content owners are reluctant to disseminate their high valued videos, which might be perfectly copied and rapidly distributed at large scale. On the other hand, digital watermarking does not seem to be mature enough in order to offer a reliable solution in the context of video. For example, it was mentioned in the copy control architecture of the Digital Versatile Disk (DVD) in 1996. However, no standard has been defined yet and it is not implemented to date. Recent meetings even stated that watermarking may not be implemented in DVD after all.

Video watermarking has inheritated from the results obtained for still images [4]. Some algorithms exploit the specificities of a compression standard [9]. However, most of the time, watermarking digital video content is regarded as watermarking a sequence of still images. The drawback of such a straightforward adaptation is that it does not consider the very specific nature of video content and this results in weak algorithms in terms of *security*. In Section 2, the collusion issue in the context of video is pointed out. Three different video collusion attacks are then described in Section 3. Eventually, some experimental results are presented in Section 4 and some tracks for future work are discussed in Section 5.

Contact author: jld@eurecom.fr

2. COLLUSION AND WATERMARKING

When high valued digital data is distributed at large scale, a set of malicious users may collude to obtain unprotected data. In the context of digital watermarking, colluders are likely to merge their knowledge, e.g. different watermarked data, in order to produce illegal content, i.e. unwatermarked data. Two different scenarii for successful collusion have already been isolated [13].

Collusion type I: The *same watermark* is embedded into *different data*. The colluders can estimate[1] the watermark from each watermarked data and obtain a refined estimation by linear or non-linear combination, e.g. the average or the median, of the individual estimations. Unwatermarked content can then be obtained by subtracting this estimation of the watermark from the watermarked data.

Collusion type II: *Different watermarks* are embedded into different copies of the *same data*. The colluders only have to make a combination, e.g. the average, of the different watermarked data to produce unwatermarked content[2].

Collusion is a very crucial issue in the context of digital video watermarking. There are indeed twice more opportunities to design a collusion than with still images.

Inter-videos collusion: Several users own a watermarked version of a video and gather together in order to produce unwatermarked video content. In a copyright protection environment, the same watermark is embedded in different videos and collusion type I is possible. Alternatively, in a fingerprinting application, the watermark will be different for each user and collusion type II can be considered. Inter-videos collusion requires different watermarked videos in order to produce unwatermarked content.

Intra-video collusion: This is a video-specific opportunity for collusion. This comes from the fact that watermarking video often comes down to watermarking series of still images. If the same watermark is inserted in each frame, collusion type I strategy can be enforced since moving scenes provide different images. On the other hand, if alternative watermarks are embedded in each frame,

[1] A simple watermark estimation consists in computing the difference between the watermarked data and a low-pass filtered version of it

[2] Indeed, averaging different watermarks generally converges toward zero.

collusion type II becomes a danger in static scenes since they produce similar images. As a result a watermarked video *alone* permits to remove the watermark from the video stream.

The danger of collusion is not always critical depending on the targeted application. For example, in a broadcast monitoring context, it is useless to remove a watermark from a video commercial. The advertiser will indeed detect that the commercial has not been broadcasted and sue you in court. Nevertheless, there are many upcoming applications (Pay-Per-View or Video-On-Demand) where collusion has to be addressed, since it may open ways for forgery and later on result in a drastic loss of royalties. In the remainder of this article, we will focus on intra-video collusion. Concerning inter-video collusion, it should be possible to apply results from still images [1, 3, 14].

3. INTRA VIDEO COLLUSION ATTACKS

Despite the recent efforts for designing efficient benchmarking tools [2], video watermarking algorithms are not evaluated in a hostile environment yet. In other terms, the verification process simply checks if the watermark survives attacks without any underlying malicious intelligence, e.g. lossy compression, transcoding, aspect ratio conversion, frame by frame attack... However it is sometimes necessary to evaluate the resistance of the watermark against hostile intelligence, especially if high valued video contents are distributed at large scale.

3.1. Frame temporal filtering (FTF)

Watermarks are generally located mostly in high frequencies and one of the simplest way of removing them is to low-pass filter the watermarked data. Spatial filtering has been investigated extensively and most algorithms for still images are resilient against it. In the context of video, since neighbor video frames are very similar, temporal filtering can be used to estimate the video frames before watermarking. This can be written:

$$\tilde{F}_k = \mathcal{F}(E_k), \quad E_k = \{F_i, 0 \leq |i-k| \leq w/2\} \qquad (1)$$

where F_i is the original video frame at position i, w is the size of the temporal window, $\mathcal{F}(.)$ is the used temporal low-pass filter and $\tilde{F}_k$ is the k^{th} attacked video frame.

The embedded watermark should be temporally in high frequencies, i.e. watermarks present in the temporal window should be uncorrelated, so that temporal filtering succeeds in stirring out the watermark signal. Such a situation occurs for example when the encoded payload [5] or the carried timestamp [10] changes. Moreover, for visibility reasons, such a filtering can only been performed when the video frames of the temporal window are highly similar. As a result, frame temporal filtering is pertinent when dealing with a static scene, which has uncorrelated watermarks embedded in each frame.

3.2. Watermark estimation-remodulation (WER)

Instead of estimating the signal before watermarking, an alternative approach consists in estimating the embedded watermark first and remodulating it later [15]. Since watermarks are generally in high frequencies, a rough estimation is given by the difference between the watermarked signal and its low-pass version. When dealing with video data, several individual estimates, obtained from different frames, can be combined to further refine the watermark estimation as follows:

$$\hat{W} = sign\left(\frac{1}{|\mathcal{S}|}\sum_{k\in\mathcal{S}} T_t(F_k - \underline{F_k})\right) \qquad (2)$$

where F_k is the k^{th} video frame, $\underline{F_k}$ its low-pass filtered version, $\mathcal{S}$ is a set of randomly chosen frames and $|\mathcal{S}|$ the cardinal of this set. An additional thresholding operation $T_t(.)$ is performed to remove non pertinently estimated high valued samples, e.g. around edges. Most detectors are based on correlation. The best way to confuse them is consequently to reduce the correlation between the estimated watermark $\hat{W}$ and each attacked video frames $\tilde{F}_k$ down to zero thanks to the following equation:

$$\tilde{F}_k = F_k - \frac{\langle F_k, \hat{W}\rangle}{\langle \hat{W}, \hat{W}\rangle}.\hat{W} \qquad (3)$$

where $\langle , \rangle$ is the inner product. Additionally, the estimated watermark $\hat{W}$ can be perceptually shaped for visibility reasons before remodulation.

In order to refine the estimation in Equation 2, the individual estimates need to be correlated i.e. the same watermark should be embedded in each video frame like in [7]. Furthermore, if the scene is static, the individual estimates will be roughly the same and no significant refinement is to be expected. In other terms, watermark estimation remodulation is pertinent when dealing with a moving scene which has the same watermark embedded in each frame.

3.3. Frame temporal filtering after registration (FTFR)

The last trend in dewatermarking relies on replacing each part of the watermarked signal with one or a combination of other parts from the same signal. For example, similar blocks of an image, carrying different parts of the watermark, can be swapped to confuse the detector [11]. In a video context, neighbor frames are highly similar and one can try to estimate each video frame from its neighbors. This was done previously with a simple temporal low-pass filtering in Equation 1. However, in order to cope with large temporal window and moving scenes, it could be useful to register each frame with a reference frame before filtering, which can be written as follows:

$$\tilde{F}_k = \mathcal{F}(E_k), \quad E_k = \{F_i^{(k)}, 0 \leq |i-k| \leq w/2\} \qquad (4)$$

where the $F_i^{(k)}$ are the original video frames after registration with the k^{th} frame. Each video frame is a projection of a three-dimensional scene and neighbor frames can be seen as different projections of almost the same scene. Frame registration brings all those projections onto the same reference frame so that all the projections of a given 3D point from the scene overlap. This allows temporal filtering with large windows without introducing much visual distortion.

Obviously, frame temporal filtering is a specific case of Equation 4 when the registration function is the identity i.e. $F_i^{(k)} = F_i$. As a result, frame temporal filtering after registration will have similar performances when dealing with a watermarking algorithm, which inserts different watermarks in the frames of a static video scene.

4. EXPERIMENTAL RESULTS

The efficiency of the FTF and WER attacks has been previously demonstrated in [13]. The remaining part of the article will consequently be devoted to the evaluation of the FTFR attack. A possible implementation based on video mosaicing is proposed and is tested against two alternative watermarking algorithms.

4.1. Watermarking algorithms

When dealing with video watermarking in a frame by frame manner, two major alternative strategies can be enforced. The same watermark can be inserted in each video frame as in [7]. Alternatively, a different watermark can be embedded in each video frame as in [5, 10]. This situation has consequently motivated the use of two simple watermarking algorithms in the spatial domain based on the spread spectrum technique.

Algorithm 1 (WM1): The same watermark is embedded in each video frame F_k according to the following equation:

$$\check{F}_k = F_k + \alpha_k.W = F_k + \alpha_k.W(K) \tag{5}$$

where W is a normally distributed watermark with unit variance and pseudo-randomly generated from a secret key K. The adaptive embedding strength α_k is chosen in such a way that the linear correlation between the watermarked frame $\check{F}_k$ and the watermark W is equal to a given target value t_{lc}. As a result, α_k is given by:

$$\alpha_k = t_{lc} - \frac{1}{N}\langle F_k, W\rangle \tag{6}$$

where N is the number of pixels in a video frame.

Algorithm 2 (WM2): A different watermark is embedded in each video frame F_k as follows:

$$\check{F}_k = F_k + \alpha_k.W_k = F_k + \alpha_k.W(K+k) \tag{7}$$

where the watermark W_k is now generated from a frame dependent seed $K + k$ and the adaptive strength α_k is given by Equation 6 where W_k has been substituted to W.

Both algorithms rely on linear correlation for detection. Each video frame F is correlated with the assumed embedded watermark W_F and the result is compared to a threshold T to assert the presence or absence of the watermark. The overall detection process can consequently be written:

$$lc(F, W_F) = \tfrac{1}{N}\langle F, W_F\rangle \tag{8}$$
$$\begin{cases} W_F & \text{is absent if} \quad lc(F, W_F) < T \\ W_F & \text{is present if} \quad lc(F, W_F) > T \end{cases}$$

where $lc(F, W_F)$ is the linear correlation between the frame F and the potentially embedded watermark W_F. Since the embedding process ensures that this linear correlation value should be t_{lc}, the threshold T can be reasonably set to $t_{lc}/2$.

In relation with the definition of the attacks in Section 3, one can easily predict the performances of those two algorithms. The individual estimates of the watermark repeatedly inserted by WM1 can be gathered for a finer estimation and the WER attack will be successful. On the other hand, the frame dependent watermarks inserted by WM2 will be washed out by a FTF attack.

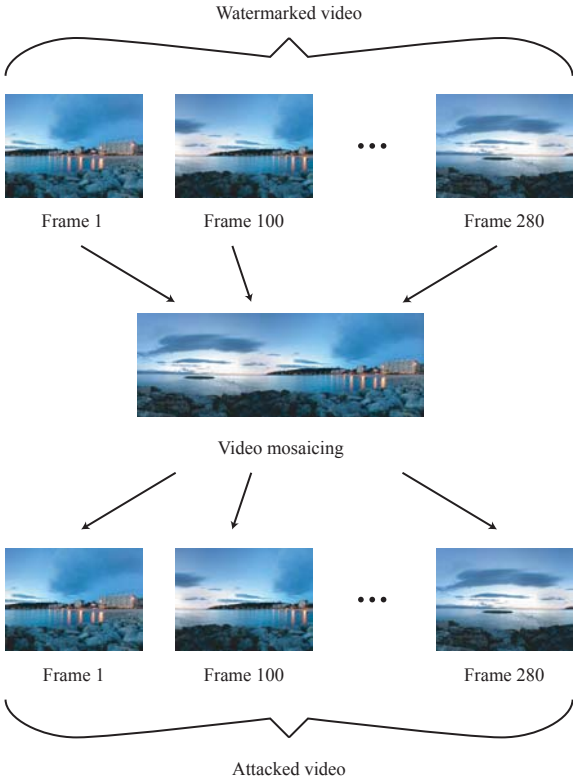

Fig. 1. FTFR implementation using video mosaicing.

4.2. Video mosaicing

When an infinite temporal window is considered in Equation 4, frame temporal filtering after registration can be seen as video mosaicing: all the video frames are brought onto the same reference frame before being averaged. Video mosaicing is an open issue for research [6] and the point of this article is not to present a method of doing it. Synthetic videos have consequently been generated from large panorama images for experiments. Starting with a panorama image, a frame size $w \times h$ and a set of displacements $\mathcal{D}$, a synthetic video is generated. For the moment, only translations have been used in the experiments. This corresponds to a tracking sideways of the camera in front of a far static background. Moreover, non-integer displacements can be used but it means that interpolations have to be performed at different moment of the process. Since interpolation can be considered as spatial low-pass filtering, using such non-integer displacements induce a first degradation of the watermark. As a result, only integer displacements have been considered so that the reported results are only due to the FTFR attack.

A typical experiment is shown in Figure 1 where the whole process of the FTFR attack is depicted. A synthetic video, consisting of 280 frames of size 512×384, has been generated from a panoramic view of the Cap d'Antibes. In this specific case, the displacements $\mathcal{D}$ are only 2 pixels per frame horizontal translations from right to left. This video is subsequently watermarked using the two previously presented algorithms (WM1 and WM2) with a targeted correlation value t_{lc} equal to 3. For each one of the watermarked videos, a detection is performed immediately af-

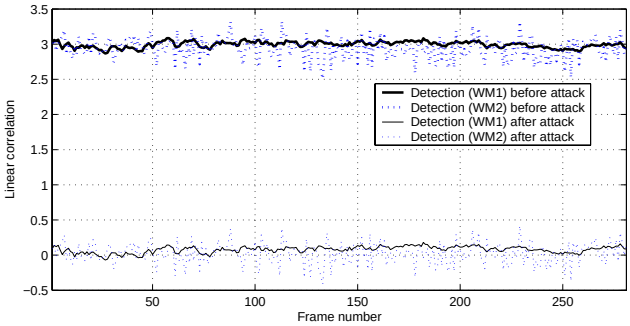

Fig. 2. Effect of the FTFR attack on the detection results.

ter embedding. The Figure 2 shows that a watermark is detected in each video frame with both methods. Moreover, the correlation score is close to the expected one t_{lc}. From each one of the watermarked videos, a mosaic is built with the video frames and the same set of known displacements $\mathcal{D}$ than during the generation of the synthetic video. The resulting mosaics of size 1072×384 are then split again into video frames to obtain the attacked videos. A watermark detection is then performed and the results are gathered in Figure 2. For both methods, the correlation score has dropped down almost 0, i.e. far below the detection threshold $T = 1,5$. No video frame is detected as containing a watermark anymore.

5. CONCLUSION AND FUTURE WORK

Video watermarking has been considered as a simple extension of image watermarking for a long time, by enforcing a frame-by-frame strategy. However, such an adaptation has led to weak video watermarking schemes, in particular when considering collusion. Both systematic strategies *always insert the same watermark* and *always insert a different watermark* have been proven to be bad as shown in Table 1. For the moment, our implementation of the FTFR attack only handles a very academic case: global translation in a large panoramic image. Our implementation will consequently be slightly modified so that real camera motion can be handled (pan and tilt), as well as moving objects with their own motion. Future work will also explore other implementations of this FTFR attack using optical flows. Some skeptical people might consider that this attack is too intensive in terms of computations to be realistic. Nevertheless, such video mosaics or *sprite panoramas* are also used for efficient compression of the background in the upcoming video standard MPEG-4 [8]. As a result, MPEG-4 compression will have a similar impact on the watermark than our implementation of the FTFR attack.

	WM1	**WM2**
Static scene	×	FTF / FTFR
Moving scene	WER / FTFR	FTFR

Table 1. Pertinence of collusion attacks

There has been a *game* between *watermarkers* and *hackers* for a long time. Both of them participate to the progress of the domain. For example, recent advances regarding robustness have been triggered by the Stirmark attack [12]. Here, the FTFR succeeds in removing the watermark because the embedded watermarks are not spatially synchronized after frame registration. As a result, temporal low-pass filtering removes the watermark signal. The basic idea is then to add some kind of *informed watermarking* to be immune to this attack. All the projections of a given 3D point should carry the same watermark sample along the video. In other terms, one should find a way to simulate a perfect *self-watermarked world*, where each object of the filmed stage carries its own watermark. As a result, when the object is projected, the watermark follows and the video is watermarked.

6. REFERENCES

[1] D. Boneh and J. Shaw, "Collusion-Secure Fingerprinting for Digital Data", in *IEEE Transactions on Information Theory*, 44(5):1897-1905, 1998.

[2] Certimark: http://www.certimark.org

[3] J. Dittmann, A. Behr, M. Stabenau, P. Schmitt, J. Schwenk and J. Ueberberg, "Combining Digital Watermarks and Collusion Secure Fingerprints for Digital Images", in *Proceedings of SPIE 3657, Security and Watermarking of Multimedia Content*, pp. 171-182, 1999.

[4] G. Doërr and J.-L. Dugelay, "A Guide Tour of Video Watermarking", in *Signal Processing: Image Communication*, 18(4):263-282, 2003.

[5] F. Hartung and B. Girod, "Watermarking of Uncompressed and Compressed Video", in *Signal Processing*, 66(3):283-301, 1998.

[6] M. Irani, P. Anandan, J. Bergen, R. Kumar and S. Hsu, "Mosaic Representations of Video Sequences and Their Applications", in *Signal Processing: Image Communication, Special Issue on Image and Video Semantics: Processing, Analysis, and Application*, 8(4):327-351, 1996

[7] T. Kalker, G. Depovere, J. Haitsma and M. Maes, "A Video Watermarking System for Broadcast Monitoring", in *Proceedings of SPIE 3657, Security and Watermarking of Multimedia Content*, pp. 103-112, 1999.

[8] R. Koenen, "MPEG-4 Overview", in *ISO/IEC JTC1/SC29/WG11 N4668*, 2002.

[9] G. Langelaar, R. Lagendijk and J. Biemond, "Real-Time Labelling of MPEG-2 Compressed Video", in *Journal of Visual Communication and Image Representation*, 9(4):256-270, 1998.

[10] B. Mobasseri, M. Sieffert and R. Simard, "Content Authentication and Tamper Detection in Digital Video", in *Proceedings of the IEEE International Conference on Image Processing*, 1:458-461, 2000.

[11] C. Rey, G. Doërr, J.-L. Dugelay and G. K. Csurka, "Toward Generic Image Dewatermarking?", in *Proceedings of the IEEE International Conference on Image Processing*, III:633-636, 2002.

[12] Stirmark: http://www.cl.cam.ac.uk/∼fapp2/watermarking/stirmark

[13] K. Su, D. Kundur and D. Hatzinakos, "A Novel Approach to Collusion-Resistant Video Watermarking", in *Proceedings of SPIE 4675, Security and Watermarking of Multimedia Content IV*, pp. 491-502, 2002.

[14] W. Trappe, M. Wu and K. Ray Liu, "Collusion-Resistant Fingerprinting for Multimedia", in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 4:3309-3312, 2002.

[15] S. Voloshynovskiy, S. Pereira, A. Herrigel, N. Baumgartner and T. Pun, "Generalized Watermarking Attack Based on Watermark Estimation and Perceptual Remodulation", in *Proceedings of SPIE 3971, Security and Watermarking of Multimedia Content II*, pp. 358-370, 2000.