Towards the Construction of a Benchmark for Video Watermarking Systems: Temporal Desynchronization Attacks

Pedro A. Hernandez-Avalos, Claudia Feregrino-Uribe, Rene Cumplido and Jose Juan Garcia-Hernandez
National Institute for Astrophysics, Optics and Electronics
Luis Enrique Erro No.1, Sta. Maria Tonantzintla, Puebla, Mexico C.P. 72840
Email: {phernandez,cferegrino,rcumplido,jjuan}@inaoep.mx

Abstract— **Although several digital watermarking schemes have emerged as a solution to traditional copyright protection technologies, only a few benchmark suites, specialized in still images and audio, have been presented to measure the robustness and performance of these schemes. The aim of this paper is to single out the particularities in the way of evaluating the performance of video watermarking systems to generate a video watermarking benchmark framework specialized in temporal desynchronization attacks. In this way, this paper presents the most important temporal desynchronization attacks and performance measures for video watermarking systems.**

I. Introduction

Today, the research in digital watermarking is growing rapidly, thus many watermarking schemes have been developed. The way to evaluate and compare them is converging to the utilization of benchmark suites to measure up its performance against various types of attacks [1]. The growing number of attacks has shown the importance of efficient and reliable benchmarking to improve the quality of existing watermarking methods. Image and audio watermarking evaluation approaches and benchmarking suites have been described in the literature, however they have neglected video watermarking.

Nowadays, the main benchmark suites are Stirmark [2], [3]: specialized in images an audio attacks; and Checkmark [4], Optimark [5] and WET (Watermark Evaluation Testbed) [6]: all of them specialized in images attacks. Although there are plans to add video attacks to the Checkmark benchmark (such as frame editing, frame reordering, aspect ratio changes, frame rate changes, collusion and averaging attacks as well as all applicable image processing attacks which can be applied to video frames) they do not include the minimum temporal attacks needed to evaluate video watermarking schemes. The video watermarking schemes are vulnerable to another kind of attacks due to the third dimension that it presents, the time.

Currently, the use of video watermarking is focused on protection issues where hiding a single watermark or hiding a few bits is enough for copyright protection of content, nevertheless the use of schemes that embed more information opens the door for the use of watermarking in emerging rich media applications like indexing, subtitles, hypervideo, interactive video, etc. [7]. Embedding little information ensures a greater probability of detection. Increasing the capacity of embedding information involves the use of more complex watermarking schemes, mainly to ensure the correct extraction of the watermark in the right order, this implies introducing a temporal synchronization phase. These new applications of video watermarking are another motivation for creating a benchmark of temporal attacks to evaluate and fairly compare video watermarking schemes.

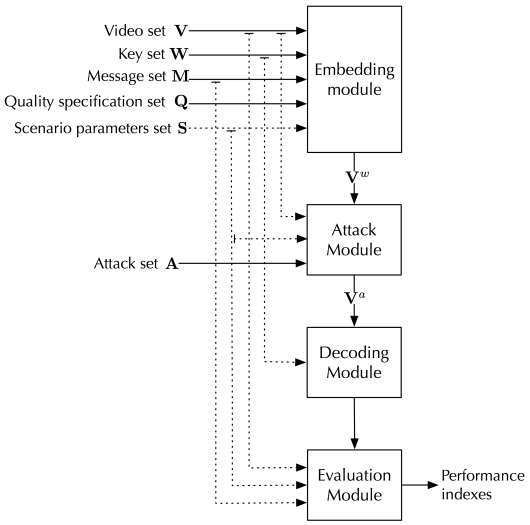

Fig. 1. Watermarking benchmark system.

II. Video Watermarking Benchmark

Like in Optimark, the inputs to the video benchmark system are the embedding and decoding algorithms, and the outputs are the performance indexes that illustrate the behaviour of the watermarking system against sev-

eral temporal attacks. Internally, the benchmark system has a fixed data set (video, keys, messages, etc.) to perform a fair comparison.

The general scheme of a video watermarking benchmark system is shown in figure 1. The benchmark system comprises the watermark embedding module, the attack module, the decoding module and the performance evaluation module. In the embedding module a message M_k is embedded into a video V_i using a key K_j. The watermarked video should satisfy the quality specification Q_l. This is repeated for all elements of **V**, **M**, **K** and **Q**. The output is a set $\mathbf{V}^w$ of watermarked videos. The attack module distorts the watermarked videos of $\mathbf{V}^w$ using all the attacks of the set **A** and a set $\mathbf{V}^a$ of attacked videos is generated. The decoding module performs the decoding and/or the extracting process into the attacked videos of $\mathbf{V}^a$. The output is an approximation to the embedded message. It is evaluated only if detection is positive. The performance evaluation module is used in order to obtain the performance scores of the watermarking scheme under test or its suitability for a certain application scenario (represented by set **S**).

The proposed video benchmark system *VidMark* focuses on evaluating the performance of the watermarking schemes against temporal attacks. At this moment only a set of temporal attacks can be used to evaluate the performance of video watermarking schemes. In the future, other attacks can be added such as collusion, additive and multiplicative noise, among others. Its important to highlight that the aim of this paper is to describe the bechmark framework and the temporal attacks (defined in the attack module) performed by VidMark. The VidMark can be found in [8].

III. Video watermarking attacks

Video sequences consist of a series of consecutive and time spaced still images called video frames. Thus, any watermarking scheme for still images can be used to watermark every frame in the video. The watermarking scheme designer needs to consider some aspects about this medium to implement a specific video watermarking scheme. First, the size of a video is bigger than in an image, thus, generally a video always will be codified to reduce the storage size. This implies that by default the watermarking scheme needs to be robust to video compression (the main compression standards are: MPEG-1, MPEG-2, MPEG-4 part 2 and part 10). Second, the big amount of information that needs to be presented to the user in a specific time increments the restrictions to consider real time applications like video streaming. And finally, the more important aspect is the presence of temporal attacks, which are not present in images e. g. frame dropping, frame transposition, frame inserting, among others.

The main robustness requirements of video watermarking schemes [9] are: video compression, signal enhancement, additive and multiplicative noise, A/D and D/A conversions, video format conversions, averaging and collusion attacks, temporal attacks, among others.

Most of these robustness requirements are spatial attacks i. e. attacks that modify each video frame separately like still images. However, the attacks that change the order or amount of video frames like video compression, A/D and D/A conversions, video format conversions and temporal attacks imply temporal modifications. The temporal modifications must be considered when the message needs to be hidden in two o more video frames, which implies a synchronization phase to extract the message in the correct order. Temporal synchronization is the process of identifying the correspondence between the temporal coordinates of the watermarked signal and the ones for the watermark.

A general diagram of video watermarking is shown in figure 2. The embedding process hides a message M into the original video **X** using, if necessary, an embedding key K_e. The result of this process is a watermarked video **Y**. This watermarked video **Y** can possibly be attacked, generating $\hat{\mathbf{Y}}$. The detection process extracts the message M' from the video $\hat{\mathbf{Y}}$ using, if necessary, the detection key K_d. The original video **X** is an ordered sequence

$$\mathbf{X} = \langle X(0), X(1), \ldots, X(t), \ldots, X(n-2), X(n-1) \rangle \quad (1)$$

$X(t)$ corresponds to the frame shown at time t and the cardinality of **X** is the total number of frames or $|\mathbf{X}| = n$. It is important to mention that, in this paper, $\langle . \rangle$ denotes an ordered sequence and $\{.\}$ denotes set membership. Thus, $\mathbf{X} = \langle X(0), X(1), X(2) \rangle$ is the ordered sequence of $X(0)$ followed by $X(1)$, followed by $X(2)$. $X = \{X(0), X(1), X(2)\}$ is a set with members $X(0), X(1)$ and X(2). In this way, the watermarked video **Y** is a sequence

$$\mathbf{Y} = \langle Y(0), Y(1), \ldots, Y(t), \ldots, Y(n-2), Y(n-1) \rangle \quad (2)$$

and the possibly attacked video $\hat{\mathbf{Y}}$ is a sequece

$$\hat{\mathbf{Y}} = \langle \hat{Y}(0), \hat{Y}(1), \hat{Y}(2), \ldots, \hat{Y}(t), \ldots \rangle \quad (3)$$

in this case the number of frames of $\hat{\mathbf{Y}}$ can be different to n due to temporal attacks. If $\hat{\mathbf{Y}}$ is identical to **Y** i. e. $\hat{\mathbf{Y}}(t) = \mathbf{Y}(t) \forall t$ then no attacks were performed.

A. Temporal Attacks

Due to the lack of a benchmark suite to evaluate video watermarking schemes against temporal desynchronization attacks, a set of temporal attacks is proposed. It does not differentiate between intentional and unintentional processing.

Frame dropping: Video frames can be deleted or simply lost during the delivery of frames in several scenarios like video streaming or video compression. This implies that $|\hat{Y}| < |Y|$ and $\hat{Y} \subset Y$. Finally, all the frames in $\hat{\mathbf{Y}}$ are ordered ascending in time, thus every pair of attacked frames $\hat{Y}(t-1)$ and $\hat{Y}(t) \in \hat{\mathbf{Y}}$ which correspond to the

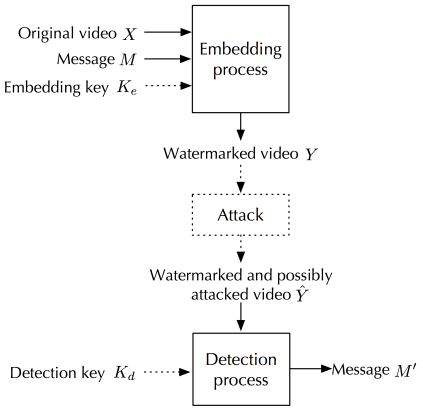

Fig. 2. General diagram of video watermarking.

watermarked frames $Y(t_a)$ and $Y(t_b) \in \mathbf{Y}$ respectively are ordered in time because $t-1 < t$ and $t_a < t_b$.

For example, let $\mathbf{Y} = \langle y_1, y_2, y_3, y_4, y_5 \rangle$ be a watermarked video, after a frame dropping attack the video generated is $\hat{\mathbf{Y}} = \langle y_2, y_3, y_5 \rangle$ (frames $Y(0) = y_1$ and $Y(3) = y_4$ were dropped). This implies that $|\hat{Y}| < |Y|$ and the time index of every pair of attacked frames in comparison with the time index of the same frames before the attack is ascending. Since the pairs of consecutive attacked frames are (y_2, y_3) and (y_3, y_5) and its time index before the attack are (1,2) and (2,4) respectively, thus the indexes operations $1 < 2$ and $2 < 4$ confirm the ascending order presented. Several video watermarking schemes that present results against frame dropping are [10]–[16], however the results they present are not standarized. In the proposed frame dropping attack, each video frame has the same probability of being dropped.

Frame inserting: This attack consists on inserting randomly unwatermarked frames (frames of the original or another video). After this attack, $|\hat{Y}| > |Y|$. Let X be the unwatermarked video that is utilized to insert its frames in the watermarked video, then the frame inserting attack satisfies the conditions: $Y \subset \hat{Y}$ and $\exists x \in X \wedge x \in \hat{Y}$.

The first condition ensures that the attacked video $\hat{Y}$ consists of all the frames of Y and the second one ensures that, additionally, some frames of $\hat{Y}$ are elements of X. An example of this attack is the next. Let $\mathbf{Y} = \langle y_1, y_2, y_3, y_4, y_5 \rangle$ be a watermarked video and $\mathbf{X} = \langle x_1, x_2, x_3, x_4 \rangle$ be an unwatermarked video. After frame inserting the attacked video would be $\hat{\mathbf{Y}} = \langle x_3, y_1, y_2, x_1, y_3, y_4, x_4, y_5 \rangle$. As it can be observed, the previous two conditions are satisfied. Some of the works that report this attack are [10]–[13], [15]. In the proposed frame inserting attack, the unwatermarked frame insertion position is randomly selected.

Frame transposing: This attack consists on interchanging two or more frames at the same time. There are two ways to perform this attack. 1)Simple transposition: A frame is selected randomly and a new position is given. This implies that more than one frame changes its position. 2)Simultaneous transposition: A frame is interchanged with any frame in the whole video. In this way, only two frames modify their positions.

In this attack $|\hat{Y}| = |Y|$ and $y \in \hat{Y} \wedge y \in Y$, but $\hat{\mathbf{Y}} \neq \mathbf{Y}$ because $\hat{\mathbf{Y}}$ and $\mathbf{Y}$ are not equally ordered.

For example, let $\mathbf{Y} = \langle y_1, y_2, y_3, y_4, y_5 \rangle$ be a watermarked video, after a simultaneous transposition attack the generated video is $\hat{\mathbf{Y}} = \langle y_1, y_5, y_3, y_4, y_2 \rangle$. In this example the frames interchanged were $Y(1) = y_2$ and $Y(4) = y_5$. Some of the works that report this attack are [13] and [16].

Frame decimation: The frame decimation occurs when x consecutive frames are dropped. Decimation with a factor of N_D retains the first frame out of every N_D consecutive frames. Another way of performing the frame decimation is by randomly selecting the frame to retain out of the N_D frames, thus the selected frame can be anyone. In this attack $|\hat{\mathbf{Y}}| = \lfloor |\mathbf{Y}|/N_D \rfloor$. A work that reports this attack is [13].

Frame averaging: This attack can be performed in two ways. 1)Sliding averaging: A window of fixed size is swept temporally across the video. At each step, a composite frame is constructed by computing the average pixel value over all the frames in the window. 2)Simple averaging: Like the simultaneous transposition, a frame is averaged with other frame in the whole video. This attack generates a video

$$\hat{\mathbf{Y}} = \langle \hat{Y}(0), \hat{Y}(1), \ldots, \hat{Y}(n-1) \rangle$$

where

$$\hat{Y}(t)_{ij} = \frac{1}{k} \sum_{x=0}^{k-1} \bar{Y}(x)_{ij}$$

$$\bar{\mathbf{Y}} = \langle \bar{Y}(0), \bar{Y}(1), \ldots, \bar{Y}(k-1) \rangle$$

here, $\bar{\mathbf{Y}}$ is the frame set selected to be averaged with $K > 0$, $0 \leqslant t \leqslant n-1$ and in all the cases $|\hat{Y}| = |Y|$. In [13] this attack is reported.

Frame rate conversion: In this work this attack is performed in two ways. 1)Duplication and dropping. The way to achieve the FPS (frame per second) required is by duplicating and deleting frames. 2)Frame blending: The only frame operation permitted is frame averaging to insert a frame or to substitute a frame. A work that reports this attack is [15].

Combined attack: Combination of previous attacks can be performed. This kind of attack generally decreases drastically the visual quality of the video.

MPEG compression: In most cases, it is impractical to store raw videos due to the huge information redundancy they present, so, video compression algorithms are used to reduce this redundancy. The MPEG compression supported by the proposed benchmark are: MPEG-2, MPEG-4 part 2 and MPEG-4 part 10. It is important

to mention that, although the MPEG compression is not considered a temporal attack, internally the MPEG compression performs temporal attacks like frame deleting to reach the bitrate required (the way to perform this depends on the implementation of the rate control unit). Moreover, in MPEG compression the most obstructive video attack appears, the motion prediction attack, that in some cases deletes most of the frame energy.

The attacks described above are considered the most significative temporal attacks in video watermarking schemes.

IV. Performance measures commonly used

In order to properly evaluate the performance of video watermarking schemes and to allow a fair comparison between different schemes, a benchmark suite must include a set of tests and a way of measuring the results of the tests using controlled conditions. In this way, the tests initial conditions are of high importance. Thus, attack parameters would be selected by analyzing the state of the art in video watermarking schemes that resist temporal attacks.

In watermarking, the tests are oriented to measure the requirements of an application. So, the robustness, the fidelity and the capacity are commonly measured.The term robustness describes the watermark resistance to these attacks and can be measured by the bit correct ratio (BCR), which is defined as the ratio of correctly extracted bits to the total number of embedded bits. The most important performance measure relates the watermark robustness to the attack and shows the overall behavior of the method towards attacks. Additionally, the false probability alarm and false rejection probability tests can be performed. Fidelity is the perceptual similarity between the original and the watermarked video. Fidelity is usually defined in terms of Signal to Noise Ratio (SNR) or Peak Signal to Noise Ratio. The capacity is the maximum amount of information that can be hidden in a medium, due to the robustness is the most important requirement used in watermarking, the capacity is usually not considered.

Unlike the image watermarking, the embedding and detection time is important to evaluate, since the video presents the time dimension.

V. Conclusion

A new framework of video watermarking benchmark specialized in temporal attacks, has been described in this paper. Due to the temporal attacks can cause loss of synchronization and this kind of attacks are exclusive for video, the main temporal attacks are detailed and different ways of performing them are described. As future work, the performance measures will be analyzed in depth in order to evaluate the fidelity and robustness, among others. In addition, the construction of a video data set can be performed. Finally, due the module approach of the benchmark framework more video attacks and performance measures can be added.

Acknowledgment

The authors acknowledge to CONACYT the support provided through the grant for PhD studies number 174444 and the project grant CB-2007-1-84668.

References

[1] M. Kutter and F. A. P. Petitcolas, "A fair benchmark for image watermarking systems," *Security and Watermarking of Multimedia Contents*, vol. 3657, pp. 226–239, Abril 1999.

[2] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Information Hiding, Second International Workshop, IH'98*, ser. Lecture Notes in Computer Science, D. Aucsmith, Ed., vol. 1525. Portland, Oregon, U. S. A.: Springer, Abril 1998, pp. 218–238.

[3] F. A. P. Petitcolas, "Watermarking schemes evaluation," *IEEE signal processing*, vol. 17, no. 5, pp. 58–64, Septiembre 2000.

[4] S. Pereira, S. Voloshynovskiy, M. Madueño, S. Marchand-Maillet, and T. Pun, "Second generation benchmarking and application oriented evaluation," in *In Information Hiding Workshop III*, Pittsburgh, PA, USA, Abril 2001.

[5] V. Solachidis, A. Tefas, N. Nikolaidis, S. Tsekeridou, A. Nikolaidis, and I. Pitas, "A benchmarking protocol for watermarking methods," in *2001 IEEE International Conference on Image Processing*, vol. 3, Thessaloniki, Greece, Octubre 2001, pp. 1023–1026.

[6] H. C. Kim, H. Ogunleye, O. Guitart, and E. J. Delp, "The watermark evaluation testbed (wet)," in *Security, Steganography, and Watermarking of Multimedia Contents*, ser. Proceedings of SPIE, E. J. Delp and P. W. Wong, Eds., vol. 5306. SPIE, 2004, pp. 236–247.

[7] C. O. Dumitru, S. Duta, M. Mitrea, and F. Preteux, "Gaussian hypothesis for video watermarking attacks: Drawbacks and limitations," *The International Conference on "Computer as a Tool" EUROCON, 2007*, pp. 849–855, Septiembre 2007.

[8] [Online]. Available: http://ccc.inaoep.mx/~cferegrino/vidmark/

[9] EURECOM, UNIGE, TUD, AUTH, UVIGO, NIM, INA, EPFL, PHILIPS, TCC, IGD, and UCL, "Common data processing and intentional attacks," Certimark, Tech. Rep. D 4.1, Oct. 2001. [Online]. Available: http://www.certimark.org/public/public_deliverable.html

[10] L. Kezheng, Y. Wei, and F. Bo, "Digital watermarking synchronization algorithm of mpeg-4 video," *Computer and Computational Sciences, International Multi-Symposiums on*, vol. 0, pp. 164–167, 2008.

[11] C. Chen, T. gang Gao, and L. zong Li, "A compressed video watermarking scheme with temporal synchronization," *Congress on Image and Signal Processing*, vol. 5, pp. 605–612, 2008.

[12] L. Kezheng, Y. Wei, and L. Pie, "Video watermarking temporal synchronization on motion vector," *3rd International Conference on Intelligent System and Knowledge Engineering. ISKE 2008.*, vol. 1, pp. 1105–1110, Noviembre 2008.

[13] E. T. Lin and E. J. Delp, "Temporal synchronization in video watermarking," *IEEE Transactions on Signal Processing*, vol. 52, no. 10, pp. 3007–3022, Octubre 2004.

[14] L. he Zhang, G. Xu, J. jun Zhou, and L. jie Shen, "A video watermarking scheme resistant to synchronization attacks based on statistics and shot segmentation," *Seventh International Conference on Intelligent Systems Design and Applications. ISDA 2007*, pp. 603–608, Octubre 2007.

[15] L. Yang and Z. Guo, "A robust video watermarking scheme via temporal segmentation and middle frequency component adaptive modification," in *Digital Watermarking*, ser. Lecture Notes in Computer Science, vol. 4283, 2006, pp. 150–161.

[16] J. Sun and J. Liu, "A temporal desynchronization resilient video watermarking scheme based on independent component analysis," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 1, Sept. 2005, pp. I–265–8.