# Video Watermarking Scheme resistant to MPEG Compression

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

## Abstract

*In this article an extension of the watermarking scheme, based on key sequence, proposed by Lin et al. is presented. This new scheme is designed taking into account the temporal modifications of the MPEG-2, MPEG-4 part 2 and MPEG4 Part 10 compressions. Besides the redundancy control of Lin is modified and the characteristics extraction process is improved. In this way, a better performance, up to 40% in the BCR (bit correct ratio) measure is gained. Experimental results show that the proposed scheme achieves a better performance against MPEG video compressions than Lin's scheme.*

## 1. Introduction

Currently, the use of video watermarking is focused on protection issues where just a few bits are enough for copyright protection of content, nevertheless the use of schemes that embed more information opens the door for the use of the watermarking in emerging rich media applications like indexing, subtitles, hypervideo, interactive video, etc. [1]. The protection of videos is achieved usually by hiding a single watermark in a full video [2]. The watermarking schemes for media protection embed little information to ensure a greater probability of detection. Increasing the capacity of information embedding 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 (temporal synchronization is the process of identifying the correspondence between the temporal coordinates of the watermarked signal and the ones for watermark). Furthermore, it is important to mention that the video will be compressed, which causes the watermarking schemes to be robust to video compression attacks.

The video compression process removes temporal redundancy by using a motion compensation unit and a rate control unit. These two internal tasks are the main cause of temporal desynchronization on compressed videos. The temporal synchronization is crucial for successfully detecting watermarks. If the detector cannot be synchronized with its input video, an embedded watermark cannot be detected even though it is present in the video. The video compression schemes attack the watermark hidden in each frame and desynchronize in time the detector due to the mechanism used to reduce the redundancy in the video. Thus, if two or more frames hide a single watermark (regardless of the reasons), then a temporal synchronization process is necessary to resist the video compression attack.

In order to achieve temporal synchronization and resistance to video compression, some watermarking schemes, like [3, 4], hide information by changing slightly the magnitudes of the motion vectors of an entire scene. To perform this, motion vectors are classified in different groups, then a bit of information is embedded in each one. In this way, a bit is hidden within a group of vectors with similar characteristics. Note that when using this internal task of MPEG compression [5, 6, 7] (the generation of motion vectors), these schemes provide robustness to this attack. This could be seen also as a disadvantage, because if the video is re-encoded, the watermark can be lost. Another scheme hides information using other inner process carried out into the compression process like the use of the DCT coefficients in the MPEG-4 compression [8]. Here, spread spectrum is used to hide the same watermark in a subset of DCT blocks per video scene. This scheme depends on the internal task performed by the encoder. Lin's scheme hides informa-

978-1-4244-4480-9/09/$25.00 ©2009 IEEE

tion on the raw video, which makes it independent of the encoder. However, it is not resistant to compression attacks. The proposed scheme consists on improving Lin's scheme to make it robust to MPEG-2, MPEG-4 Part 2 and MPEG-4 part 10 compression attacks. It is important to mention that this scheme present the best resistance against the more common temporal synchronization attacks.

## 2. Scheme proposed by Lin *et al.*

The Lin's scheme [9, 10, 11] performs temporal synchronization by manipulating the embedding keys used for watermarking each frame. In this way, the watermark, that is embedded in each frame, carries along temporal information. The key is changed every $\beta$ frames using the information of the current frame and the previous key, thereby a new key time-dependent of the current and the previous frame is generated. To perform key generation a finite state machine (or FSM) is used ( figure 1). The key generation is achieved in the following way:

1. The FSM is restarted (state 0) and the current embedding key $K_i$ is set to $K_e$ (the master key), then the embedding process generates a watermarked frame $Y(i)$.

2. The characteristics X, Y and Z are obtained from $Y(i)$ and they are used to define the next state in the FSM.

3. The next embedding key $K_{i+1}$ (used to watermark the frame $X(i+1)$) is computed using the state equations shown in the figure 1.

4. The next frame is obtained and the embedding key $K_{i+1}$ is used to watermark it. The steps 2, 3, 4 are carried out, using this new frame, until all frames are generated and watermarked.

There is also a mechanism for generating a period redundancy $\alpha$ (indicates the maximum number of frames that will be watermarked until the use of $K_e$) and a repeat redundancy $\beta$ (indicates the amount of frames watermarked using the same key).

Figure 2 shows an example of a key sequence with $\alpha = 8$ y $\beta = 2$. The same color indicates that the same key was used to hide the watermark. It can be noticed that every two frames the key is changed and every eight frames the key is restarted. Moreover, the keys used in the frames 2 and 3 are not the same as those used in the frames 10 and 11. During a period $\alpha$ the only key that remains unchanged is the master key $K_e$. The frames watermarked with the master key are

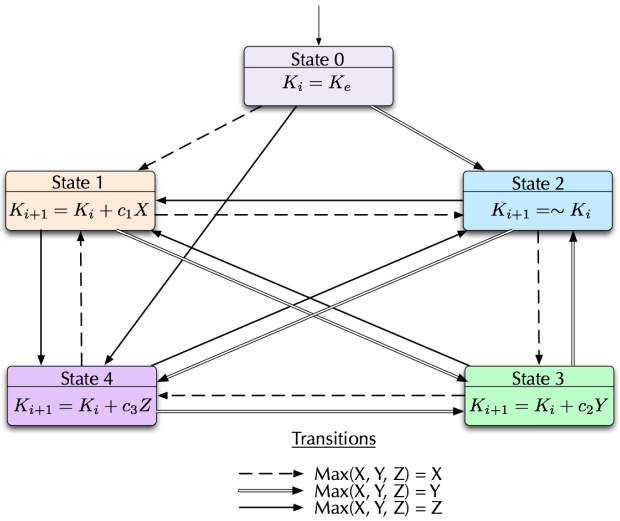


**Figure 1. Finite state machine used to generate new keys**

called resynchronization frames, which are used as a reference when the detector loses synchronization. The period given by $\alpha$ controls the degree of overall redundancy and determines how many frames the key generator has restarted. The repetition $\beta$ controls the degree of local synchronization. A high degree of repetition $\beta$ increases the redundancy and decrements the generation of new keys, which is an advantage to resist temporal attacks such as frame dropping, frame transposing and frame averaging. Moreover, if the degree of repetition is small there is little or no redundancy, which means that when a single watermark is not detected the synchronization period is completely lost.

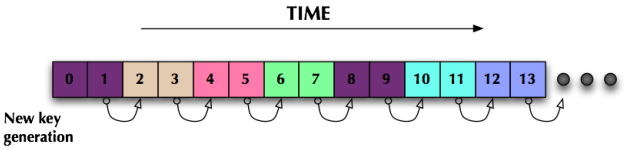


**Figure 2. Key sequence with $\alpha = 8$ y $\beta = 2$**

In the watermark detection process, the main task is carried out by the finite state machine and by a queue that keeps track of frames not synchronized. A null queue is generated first, and then the key $K_e$ is used for continuously extracting the watermark from the frames. Once this watermark is correctly detected, the predicted state-keys (using the same FSM as in the embedding, figure 1) is pushed into the queue. Then watermark extraction of each frame is repeated with

the keys $K_e$ and those reserved in the queue. When a watermark is detected, the current state-key will be moved to the front of queue if it does not belong to the initial set. Then the predicted state-keys of current state-key is pushed into the queue. It is important to mention that key generation in the embedding and in the extracting processes depends on the use of the FSM. It is important to mention that when the magnitudes of X, Y and Z (from a frame) are close during the embedding process and $Y(i)$ is attacked, the FSM transition (in the extracting process) can be different from the expected one, causing the temporal desynchronization.

## 3. MPEG video compression

In order to gain robustness in Lin's scheme to MPEG-2, MPEG-4 part 2 and MPEG-4 part 10 compression, three improvements are proposed. These MPEG standards include three kinds of frames: 1) intra picture frames (I-frames); 2) forward-predicted frames (P-frames); 3) bidirectional-predicted frames (B-frames). A video stream or recording will always start with an I-frame and will typically contain regular I-frames throughout the stream. These regular I-frames are crucial for the random access of recorded MPEG-4 files, such as with rewind and seek operations during playback. The disadvantage of I-frames is that they tend to compress much less than P-frames or B-frames [12].

I-frames are coded without reference to other frames, they are coded like an image. P-frame applies motion prediction by referencing an I-frame or P-frame in front of it, motion vector points to the block in the referenced frame. B-frame applies motion prediction, referencing a frame in front of it and/or a frame behind it. Each of the two referenced frames may be I-frame or P-frame. Macro block (MB) in video stream is represented as a $16 \times 16$ sample area. Each MB contains six $8 \times 8$ blocks, four for luminance and two for chrominance. A block of an I-frame contains six $8 \times 8$ blocks, four for luminance and two for chrominance. A block of an I-frame contains simply values of luminance or chrominance of its own. A block of a P-frame or B-frame contains the difference between the values of itself and the referenced block. This process is called motion compensation. Each frame is divided into MBs. Coding process of each block includes DCT, quantization, run length encoding and entropy coding in that order. The resulting video stream consists of entropy codes, motion vectors and control information about the structure of video and characteristics of coding. The general frame sequence of MPEG videos (MPEG-2,MPEG-4 part2 and MPEG-4 part 10) can be seen in figure 3.

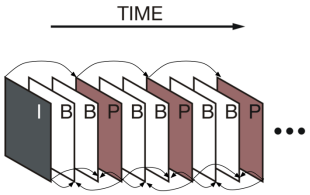


**Figure 3. General MPEG frame sequence**

## 4. Modified scheme

The tree proposed improvements to Lin's scheme are:

1. To perform an adaptable $\beta$ repetition.
2. To modify the characteristic extractor in order to consider $\beta$ frames.
3. To modify the temporal redundancy in order to generate redundancy in I-frames.

In the described scheme thus far, a fixed $\beta$ repetition is utilized. However, this can be counterproductive due to MPEG-2, MPEG-4 part 2 and MPEG-4 part 10 compressions strongly attack temporal redundancy through motion compensation. If a frame sequence belongs to an scene with little movement, this compensation destroys part of these frames. If frame destruction happens and the magnitude of $\beta$ is small, the watermark hidden in that repetition will be lost, which causes temporal desynchronization (the following keys will be lost due to they are related to the previous keys). The proposed improvement is to hide the same watermark in a number of frames given by the mutual information between them. So if similar frames the mutual information is high and disimilar frames the mutual information is low, in this way a measure of similarity in time can be utilized. The metric of mutual information is given by

$$I(X,Y) = \sum_x \sum_y p(x,y) \log \left( \frac{p(x,y)}{p(x)p(y)} \right) \qquad (1)$$

where $p(x,y)$ is the probability mass function *or pmf* of the random variables $X$ and $Y$ (i. e. frame $X$ and frame $Y$), $p(x)$ and $p(y)$ represents the marginal pmf of X and Y, respectively.

To determine the amount of frames watermarked with the same key, a threshold $U$ is used, in this case

$U = 1$. Futhermore, a limit in the number of consecutive frames to be watermarked is established to 20. An example of this improvement is shown in figure 4, the values of $\beta$ are 4, 2, 4, 3, 2 and 2. These values indicates that the first four frames are similar, the next two frames are similar too, the next four frames are similar again and so on. The dissimilarity between the consecutive frames determines the end of each $\beta$ frames.

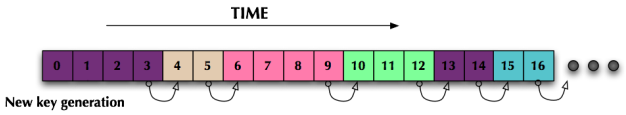


**Figure 4. Keys generated using a repetition $\beta$ adaptive**

The second improvement consists on modifying the characteristics extractor used in the Lin's scheme. The extractor obtains the characteristics from the watermarked frame (the average of frame set is being used so far) and uses them to decide the route through FSM transitions. It is important to notice that the extractor invokes the last watermarked frame of each $\beta$ repetition, which generates a dependence between the frame and the generated key. In this manner, if the last frame of repetition is eliminated during detection, the characteristics of another frame will be taken into acount to generate the new key; which causes that the new key is not the expected one. The proposed improvement consists on taking all the frames watermarked with the same key to extract their characteristics. An image that represents all the frames within $\beta$ repetition is obtained by averaging the low frequencies of the watermarked frames (utilizing a Gaussian filter with $\sigma = 4$). By adding the magnitudes of the feature points (using the Harris Detector [13]) to the X, Y and Z characteristics (obtained with Lin's original method), Lin's scheme is improved due that the new X, Y and Z characteristics are no longer close and this prevents the erroneous transition in the FSM.

In this way, the proposed extractor is not dependent of a previous frame, thus the resistance against frame dropping is improved. Figure 5 shows this behavior.

Finally, the third improvement consists on modifying the temporal redundancy control to hide a watermark in each posible I-frame(see figure 6). It is worth to mention that MPEG compression encodes some frames internally as if they were images (I-Frame), some others using motion estimation taking I-frames as basis (P-Frame) and finally some others using motion estimation taking P-frames as basis (B-Frame). In

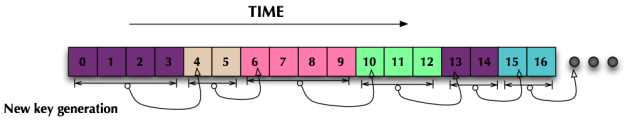


**Figure 5. Keys generated using a characteristics extractor that use a set of $\beta$ frames**

this manner, it pretends to hide the watermark on each posible I-frame to avoid the aggresive attack of motion estimation. Also, I-frames are generally created by the codificator when a shift of scene is detected. In this way, the aim is to generate redundancy of keys utilizing the shift of scene (the end of each scene is obtained using the metric of mutual information from equation 1). In this manner there is redundancy of scene $\gamma$. The idea behind this new redundancy is to guarantee detection of at least one bit for each I-frame. With these modifications, the scheme is able to support MPEG-2, MPEG-4 part 2 and MPEG-4 part 10 compressions.

It is important to mention that the period redundancy $\alpha$ (unknown at the begining of the embedding process) depends on the amount of information to be embedded and on the size of $\beta$ (adaptable). For example, if five bits are embedded and the values of $\beta$ are: 3, 4, 3, 5 and 4, $\alpha = 19$. Finally, the proposed scheme uses two $\alpha$ values, one for embed ding the information in a sequential way (like Lin's scheme) and the other for embedding the information at the begining of each scene (thirth improvement). Finally, key generation on the proposed scheme, in both embedding and extracting processes, is performed as that of Lin's scheme.

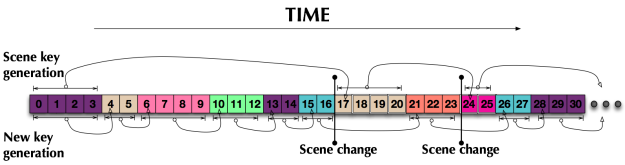


**Figure 6. Key generation redundancy per scene**

## 5. Experimental results

In order to evaluate the resistance against compression attacks, the proposed improvements described in section 4 were performed. Seven videos of $352 \times 288$ pixels, with a range of 870 to 2101 frames at 25 frames per second (fps) were used. The experiments carried

out consist on testing these videos with different *bitrates* and detecting the hidden watermark, measuring the bit correct ratio (or BCR). The utilized parameters, for Lin's scheme are $\alpha = 150$ and $\beta = 10$, in other words, 15 bits were hidden and the size of the queue is 10 entries.

Figure 7 shows the performance of both schemes, Lin's and the proposed one, against compression attacks. The performance of the proposed scheme overcomes Lin's scheme in all but the case of encrypted video at 400 Kbps bitrate where they perform similarly. It can be noticed that both schemes are affected by MPEG-4 part 10 compression, since this achieves a better reduction of temporal and spatial redundancy mechanism (bitrate control). However, the detection of the watermark using the proposed scheme is up to 40% better that Lin's scheme against the MPEG-4 part 10 compression at 200 Kbps. The bitrate influences the impact of temporal and spacial attacks during compression. The higher the bitrate the less agressive the spatial and temporal attacks, and the lower the bitrate the less severe the spatial and temporal attacks. The bitrate also affects the watermark detection, as shown in figure 7, where the detection percentage improves as the bitrate increases. It is worth to mention that Lin's scheme as well as the proposed scheme utilize a basic spread spectrum technique to embed the watermark as indicated by equation 2.

$$y_i = x_i + sw_i \tag{2}$$

Where $y_i$ is the watermarked $i$ frame, $x_i$ is the original $i$ frame, $s$ is a robustness factor and $w_i$ is a watermark. The watermark is a random variable with normal distribution $N(0, 1)$. Detection is performed by using a low pass filter to reduce the effect of the original image (as suggested in [14]) followed by correlation with the watermark signal. The correlation is carried out as shown in equation 3.

$$\begin{gathered} c = E\{\mathbf{y} \cdot \mathbf{w}\} \\ \text{if } c \approx s, \text{ the watermark exist} \\ \text{else } c \approx 0, \text{ the watermark does not exist} \end{gathered} \tag{3}$$

## 6. Conclusions

A modified state transformation scheme of key sequence for video watermarking robust to MPEG-2, MPEG-4 part 2 and MPEG-4 part 10 is proposed. The proposed modification of redundancy mechanism in embedding process increases the resistance to the main widely used video compression schemes. Experimental results indicate that the proposed scheme has a better resistance to MPEG compression than Lin's scheme.

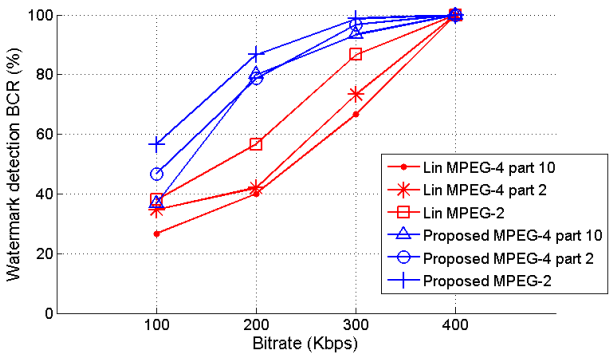


**Figure 7. Performance against MPEG-2, MPEG-4 part 2 y MPEG-4 part 10 compressions**

As future work, an analysis of the influence of the watermark embedding in the bitrate control mechanism can be performed. In addition, embedding multiple watermarks in a frame to increase the capacity scheme can be performed.

**Acknowledgments:** 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] 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, Sep. 2007.

[2] L. E. Coria-Mendoza, *Low-Complexity methods for image and video watermarking*. PhD thesis, The University of British Columbia, Vancouver, Canada, Mar. 2008.

[3] 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, Oct. 2007.

[4] 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, Nov. 2008.

[5] Moving Pictures Expert Group, *ISO/IEC 14496-2*.

[6] Moving Pictures Expert Group, *ISO/IEC 14496-10*.

[7] Moving Pictures Expert Group, *ISO 13818*.

[8] 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.

[9] E. T. Lin and E. J. Delp, "Temporal synchronization in video watermarking," in *Proceedings of the SPIE International Conference on Security and Watermarking of Multimedia Contents IV*, vol. 4675, (San Jose, California, U.S.A.), pp. 20–25, 2002.

[10] E. T. Lin and E. J. Delp, "Temporal synchronization in video watermarking," *IEEE Transactions on Signal Processing*, vol. 52, pp. 3007–3022, Oct. 2004.

[11] E. J. Delp and E. T. Lin, *Optical and Digital Techniques for Information Security*, vol. 1, ch. Watermarking Streaming Video: The Temporal Synchronization Problem, pp. 135–153. Springer New York, 2005.

[12] M. Smart and B. Keepence, "Understanding mpeg-4 video," White paper IC-COD-REP012, IndigoVision, Jun 2008.

[13] C. Harris and M. Stephens, "A combined corner and edge detection," in *Proceedings of The Fourth Alvey Vision Conference*, pp. 147–151, 1988.

[14] T. Kalker, G. Depovere, J. Haitsma, and M. Maes, "A video watermarking system for broadcast monitoring," in *In Proc. SPIE Security and Watermarking of Multimedia Contents I*, vol. 3657, (San Jose, California, U.S.A.), pp. 103–112, Jan. 1999.