Applying interest operators in semi-fragile video watermarking

Stefan Thiemert[a], Hichem Sahbi[b], Martin Steinebach[a]
[a]Fraunhofer IPSI, Darmstadt, Germany
[b]INRIA, Rocquencourt, France

ABSTRACT

In this article we present a semi-fragile watermarking scheme for authenticating intra-coded frames in compressed digital videos. The scheme provides the detection of content-changing manipulations while being moderately robust against content-preserving manipulations. We describe a watermarking method based on invariant features referred to as interest points. The features are extracted using the Moravec-Operator. Out of the interest points we generate a binary feature mask, which is embedded robustly as watermark into the video. In the verification process we compare the detected watermark with the interest points from the video to be verified. We present test results evaluating the robustness against content-preserving manipulations and the fragility in terms of content-changing manipulations. Beside the discussion of the results we propose a procedure to provide security of the scheme against forgery attacks.

Keywords: video authentication, integrity protection, interest points, content-fragile watermarking

1. INTRODUCTION

Today there are numerous application domains for watermarking-based authentication, like juristic, military and medical scenarios. A common example is video authentication for surveillance-cameras, which is important when the captured video streams are used at court. For authentication watermarking, fragile and semi-fragile approaches are two strategies. Fragile watermarking allows us to detect any slight manipulation on a material and can be compared to cryptographic hash functions and digital signatures [1]. To reduce the huge amount of data already watermarked videos have to be re-encoded into MPEG-1/2 streams of lower bit rate. Such compression, considered as content-preserving manipulation [2], makes a video unauthentic using a fragile watermarking scheme. Therefore, the latter is not suitable for authentication. Semi-fragile watermarking will be more appropriate in order to differentiate between content-preserving and content-changing manipulations. More generally, we mean by content-preserving manipulations those, which are applied in post-production processes such as compression or format changes. Content-changing manipulations remove, insert or replace objects in a single frame or a sequence of frames.

We describe in this paper a semi-fragile watermarking scheme for MPEG-1/2 compressed videos. With the scheme we can localize the manipulated positions. As the predicted frames of MPEG-1/2 (P- and B-frames) [3] store the differences to the intra-coded frames (I-frames), we concentrate only on the authentication of I-frames. Manipulations on I-frames will result in modifications of the predicted frames.

The paper is organized as following: In section 2 we give a short review on existing semi-fragile watermarking schemes for compressed video data. In section 3 we present our framework for content-fragile watermarking followed by the experimental results in section 4. In section 5 we discuss security issues and we present a possible solution. The paper finishes with a conclusion and future issues.

2. RELATED WORK

Several approaches for semi-fragile watermarking schemes for compressed images [5] and videos [4] were proposed in the past. In this work we address semi-fragile watermarking using our concept introduced in [6]. Our basic idea is to detect features from video frames, which are invariant to content-preserving manipulations but fragile to content-changing ones. We generate a feature vector and we embed it in a robust way into the I-frames of MPEG-1/2 compressed videos.

A concept for feature detection was proposed by Lin and Chang [2]. They use a relationship between two coefficients in a JPEG image, which is invariant to JPEG compression with a pre-determined lowest quality factor. This relationship is

used to generate a semi-fragile feature vector. Together with information to recover the original material, the feature vector is embedded by changing the least significant bits (LSB). In [6] we apply the concept for generating a feature vector using an energy relationship between block groups of 8x8 pixels in a given MPEG-1/2 I-frame. The embedding procedure is done with the Differential Energy Watermarking (DEW) scheme [8].

While the previously introduced approaches concentrate on semantic relationships inside JPEG images and MPEG-1/2 I-frames our approach detects manipulations on the content itself. The general concept is similar to an approach proposed by Dittmann et al. [9]. They generate a feature vector with an edge detection algorithm. As in [6] and [2] the robustly embedded feature vector is used to check the authenticity of the material and to localize possible manipulated positions.

In [13] we introduced a scheme using object detection to detect manipulations on parts of an image, defined as regions of interest (ROI). The protected objects are faces, detected by the approach in [14]. The scheme is able to detect manipulations on the content, e.g. moving faces. In contrast to the scheme proposed in this work the face detection system must be trained to detect the objects.

In [10] Yin and Yu introduced a hybrid watermarking scheme for identifying manipulations in the spatial domain as well as in the temporal domain. As a first indication for a manipulation in the spatial domain they use a highly fragile watermark, denoted as M_F. If M_F has been broken they try to detect a robust watermark M_R, which is robust to transcoding processes. Embedded control data including the frame order and frame position in a group of pictures (GOP) help to identify manipulations in the temporal domain. The information of the current GOP is embedded into the next GOP. In order to detect manipulations in the temporal domain we use a similar approach, being presented in the next section.

The feature vector extraction process is based on interest-operators. These are methods, which detect interest points in images. Interest points should be able to be identified even in image sequences with motion. Interest-operators are commonly used for feature extraction in the field of image matching and analysis. Further application fields are robotic systems and driving assistant systems. Methods for identifying interest points were addressed by Moravec [11] and Förstner and Gülch [12].

3. OUR METHODOLOGY

We focus in this work on a semi-fragile watermarking scheme for digital videos based on invariant features referred to as prominent blocks. The prominence of a block is computed using the algorithm of Moravec [11] for detecting interest points. The algorithm is totally un-supervised and does not require any a priori knowledge. The used feature is robust against content-preserving manipulations and sensitive to content-changing manipulations. Furthermore, the scheme provides security against forgery attacks. The approach proceeds as following: first, we generate a binary feature mask by detecting the most prominent blocks from a given I-frame. Then, we embed the binary feature mask robustly into an adjacent I-frame. At the detection stage, we detect and compare the watermark and the most prominent blocks of the current I-frame, in order to verify the authenticity and to find locations of possible malicious manipulations.

3.1. Binary feature mask generation

According to [11] we use the luminance values of an I-frame in order to compute the prominence of a pixel. Define $(r,c) \in N_0 \times N_0$ as a location in a grey level frame and $V(r,c,\theta) \in R$ the response of the Moravec filter at (r,c) with orientation θ. In order to compute $V(r,c,\theta)$ we estimate the mean values of the squared intensity differences in vertical, horizontal and two diagonal directions. $V(r,c) \in R$, defined as prominence of the pixel at (r,c), results from the minimum of the four responses. Using the minimum, instead of averaging, makes the Moravec filter reacting only on significant grey value differences. Therefore this filter is less sensitive to noise and content-preserving manipulations.

The watermark sequence is considered as a binary feature mask where an entry is set to 1 or 0 according to the following steps:

- First we use a smoothing filter on an I-frame to reduce noise.
- We subdivide an I-frame into block groups. The subdivision enables localization of possible manipulations inside a group.
- The prominence value, denoted by $V(b_i)$ for each block b_i, is defined as the mean of the prominence values for each pixel inside the block. As the middle and high frequencies will be used later for embedding the

watermark, these frequencies are not used for computing the interest values. This ensures that the binary feature mask will not be influenced by the embedded watermark.

- We declare a block b_i as prominent, if the prominence value $V(b_i)$ is among the k highest prominence values in the underlying group. If b_i is a prominent block and b_j is the block with the highest prominence of all non-prominent blocks, then the following condition has to be fulfilled:

$$V(b_i) - V(b_j) > t \quad (1)$$

where $t \in R$ is a threshold. If the condition is true, we set the value for b_i in the binary feature mask to 1, otherwise to 0. For all non-prominent blocks we set the value in the binary feature mask to 0. We use equation (1) in order to avoid changes in the binary feature mask after content-preserving manipulations. If the difference between $V(b_i)$ and $V(b_j)$ is too small, such a manipulation could marginally change the order of the prominence values in the group. This could result in the detection of an unauthentic group even a content-preserving manipulation was applied.

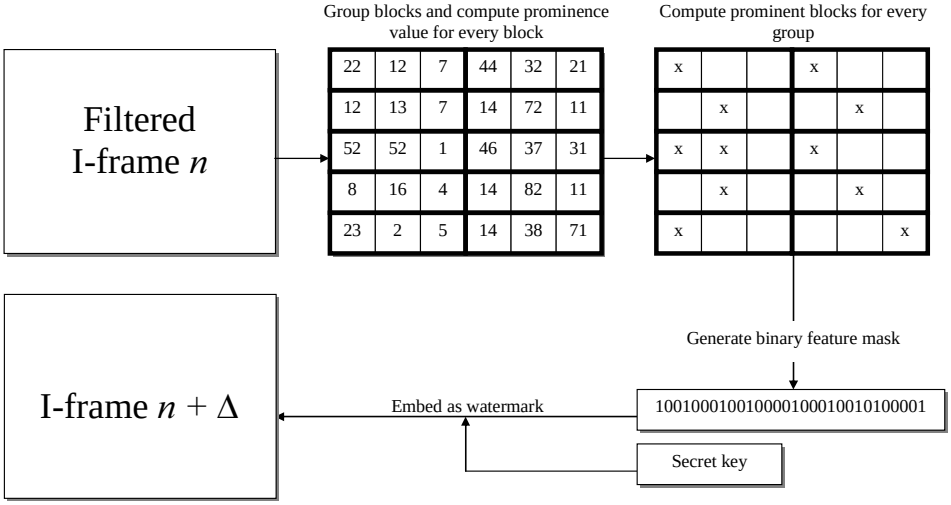

Figure 1: Binary feature mask generation and embedding process

3.2. Embedding

Using a robust watermarking scheme we embed the binary feature mask into the underlying middle and high frequency DCT coefficients of one of the adjacent I-frames. In practice we use the Differential Energy Watermarking (DEW) scheme of Langelaar et al. [8] for compressed videos. It is robust against compression and provides good transparency and high capacity. Using this scheme we embed a binary value of the mask by enforcing an energy relationship of the DCT values between the upper and the lower part on a given block group. This relationship is achieved by deleting the high and middle frequency DCT coefficients in the blocks of the upper (resp. the lower) part of the group if the binary value is 1 (resp. 0). For security reasons the watermark positions are pseudo-randomly selected, controlled by a secret key.

Other robust video watermarks can also be used for embedding the binary feature mask. The watermark has to be robust against compression while providing a high capacity. Moreover the embedding process of the watermark should not modify the DCT values used for computing the binary feature mask. A possible alternative could be the scheme, we proposed in [15].

Notice that the binary feature mask of the current I-frame is not embedded into the same frame. Any content-changing manipulation would influence the watermark and therefore localization will be impossible. Our solution is to embed the binary feature mask of the current I-frame into a neighbouring frame.

Figure 1 shows the feature generation and embedding process. After applying a smoothing filter we group the blocks of I-frame n and compute for every block its prominence value. In this example one group contains 3 blocks. For every group we compute the most prominent block. In the example k is set to 1. As can be seen, one of the groups contains two blocks with the same prominence value. The group contains no block, which is most prominent. Hence for every

block in this group we set the value in the binary feature mask to 0. The generated binary feature mask is embedded with a secret key into one of the neighbored I-frames.

3.3. Watermark detection and integrity verification

Figure 2 shows the watermark detection and integrity verification process. First we apply a smoothing filter on I-frame *n* for reducing noise, caused by e.g. compression. We group the blocks of I-frame *n* and compute the prominence value for each block. We apply the same rules as in the embedding process (continuing the example of figure 1) to generate the binary feature mask.

From the neighbouring I-frame we detect the watermark position using the secret key. Without the secret key from the embedding process we are not able to find the correct watermarking positions. According to [8] we estimate the energy relationship of the DCT values between the upper and the lower part on a given block group. If the lower part has a higher energy than the upper part we detect 1 in that group, otherwise 0.

We compare the detected watermark from the neighbouring frame with the binary feature mask of I-frame *n*. The frame is authentic only if the vectors are similar. If a block group in I-frame *n* is subject to content-changing manipulations, such as adding or deleting objects, then we expect that the prominence order will change significantly in that group. This manipulation can be detected and also localized by finding differences between the watermark and the binary feature mask.

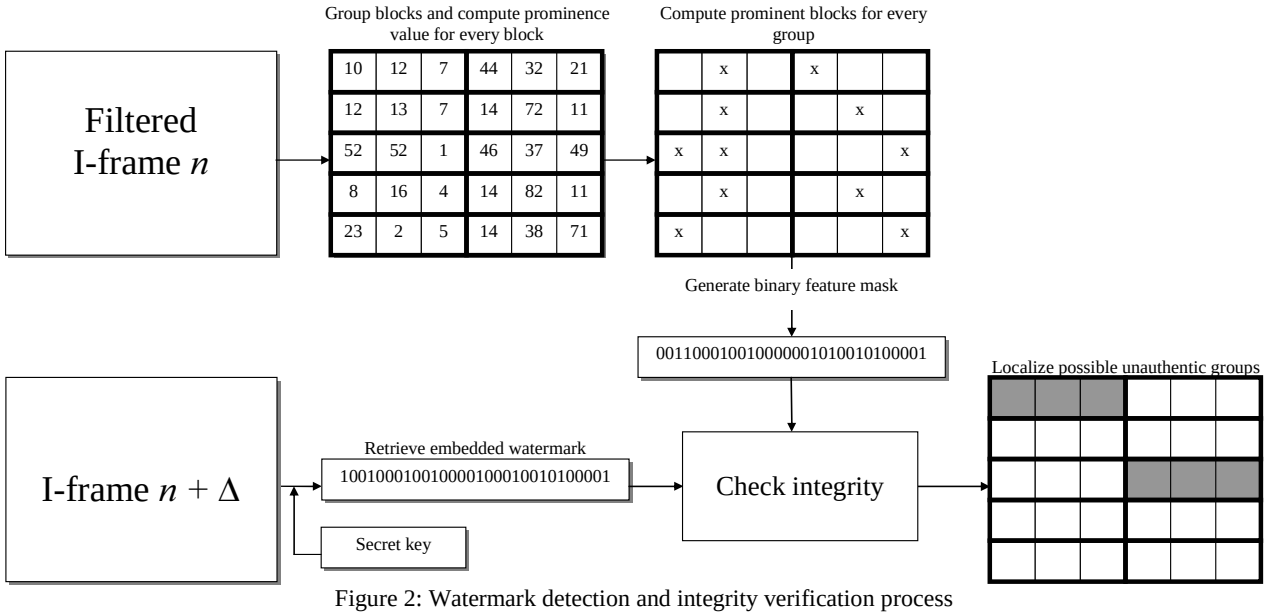

Figure 2: Watermark detection and integrity verification process

3.4. Detecting manipulations in frame order

Beside the detection of malicious manipulations inside frames it is important to detect cuts of scenes and the removal, reordering and replacement of single frames. A solution for this problem is to divide the frames in authentication groups. A similar mechanism we introduced in [7], which can be applied to a group of frames. Figure 3 shows an example of the framework. The group contains a fixed number of adjacent I-frames (here 3). Each frame contains the binary feature mask of its neighbouring I-frame and a group code *x* as watermark. The group will be used for identifying the members of an authentication group. For security reasons *x* should be inserted on different positions into the authentication message of each frame. For instance I-frame *n* may contain the binary feature mask of I-frame *n+1* and group code *x* as watermark. We explain the functionality of the authentication group with two examples.

Example 1: remove a frame

The removal of a frame can be detected by the group code. If I-frame *n+1* has been removed only two members of the authentication group contain group code *x*. Due to the fact that I-frame *n+2* can verify the integrity of I-frame *n* we can detect the removal of I-frame *n+1*.

Example 2: replace a frame

If I-frame *n+1* has been replaced by another frame the integrity of I-frame *n+2* can not be verified. I-frame *n* would be verified as authentic with the watermark embedded in *n+2*. This would be an indication for the authenticity of *n+2*. If the inserted frame is not similar to *n+1* the manipulation can be detected easily. If the inserted frame is similar to *n+1* the manipulation can be detected by the missing watermark containing group code *x*.

Beside the removal of a single frame it is possible that a complete group has been removed. For that case we can use group code *x*. For simplicity *x* can be a combination of only a few bits, which changes from one group to another.

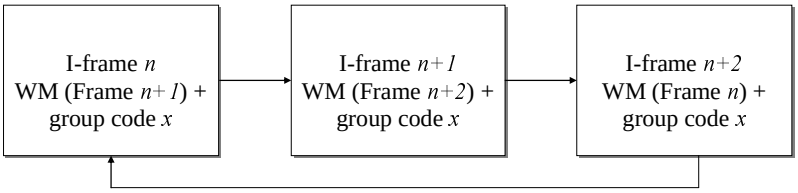

Figure 3: Example of an authentication group

4. EVALUATION

In our preliminary experiments we studied two aspects in terms of the binary feature mask: robustness against content-preserving manipulations and fragility against content-changing ones. We used a set of 100 images, divided into 10 different categories, like e.g. 3D images and images with uniform or textured areas. Each image was converted into a separate MPEG-1 video of resolution 352x288 pixels with a bit rate of 1.152.000 bit/s. The resulting MPEG videos consisted of a single I-frame representing the original image. We used up to 6 low frequency AC and DC values for each 8x8 DCT block in order to compute low-pass filtered luminance values of the individual pixels of the I-frame. The images were smoothed using different Gaussian and mean value filters of size 3x3, 5x5, 7x7 and 9x9 pixels. Such filters are often used for noise removal after compression.

At this stage we apply the Moravec filter at each location in a neighbourhood of 3x3 pixels. Then we estimate the prominence value for each block of 32x32 pixels. The block size is chosen to balance the embedding capacity of the robust watermark and the precision of detecting manipulations. The value of each block in the binary feature mask is set to 1 if the block is the most prominent inside the group of three blocks. The threshold *t* introduced in (1) varies between 0.0 and 1.0. After compression the resulting binary feature mask contains 66 entries.

4.1. Robustness

When evaluating the robustness of the binary feature mask we applied several content preserving manipulations on the material:

- Manipulation 1: re-encoding
- Manipulation 2: compression down to 75% of original bit rate
- Manipulation 3: compression down to 50% of original bit rate
- Manipulation 4: scaling down to 50% of the original resolution
- Manipulation 5: scaling up to 200% of the original resolution
- Manipulation 6: additive Gaussian noise with variance 5% of the colour values
- Manipulation 7: additive Gaussian noise with variance 10% of the colour values

We analysed whether the binary feature mask in the manipulated video is identical to the original one. For each video we estimated the error bit rate, which is the ratio of different bits between the binary feature mask of the original and manipulated video in relation to the total number of bits. Our experimental results (see figure 4) show that the average

of the error bit rate differs between 0.45% (Manipulation 1) and 2.84% (Manipulation 7). The robustness of the feature depends on the applied manipulation. Manipulations 4, 6 and 7 decrease the visual quality. This has a significant influence on the robustness of the binary feature mask. Because these attacks decrease the intensity of edges (Manipulation 4) or produce additional significant grey values (Manipulation 6 and 7) the prominence values of different blocks change. A shifting in the order of the prominence values after such a manipulation makes a group unauthentic.

We achieved the best results for robustness against content-preserving manipulations with the following parameter setting:

- Filter: combination of Gaussian and mean value filter with a resolution of 9x9 pixels
- Compute low-pass filtered luminance values only with the DC value
- Threshold *t* is set to 1.0

The setting of filter, threshold and coefficients decreases the intensity of edges. Hence the manipulations 4, 6 and 7 have not such a strong influence on the robustness of the binary feature mask.

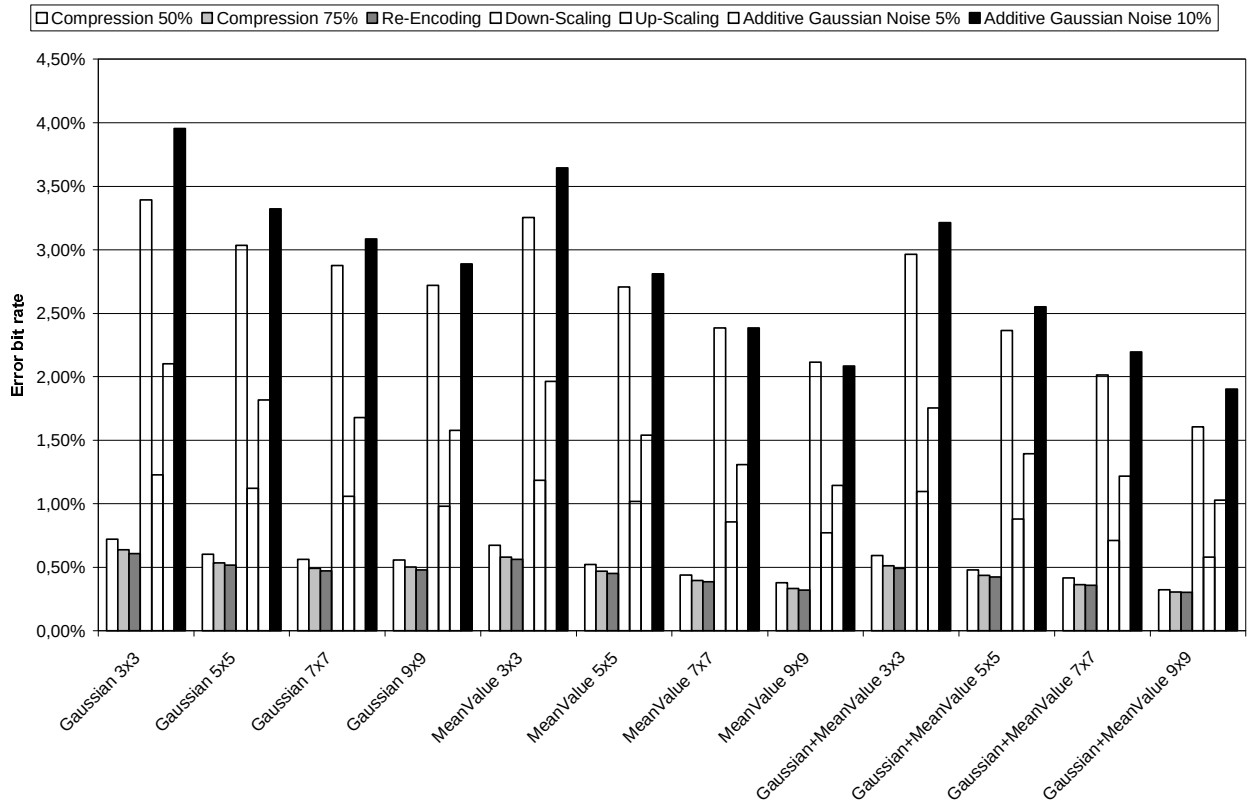

Figure 4: Robustness of binary feature mask against content-preserving manipulations

4.2. Sensitivity

We focus in this section on the sensitivity in terms of content-changing manipulations. On each of the 100 videos we applied the following most common manipulations:

- Manipulation 1: insert a logo
- Manipulation 2: move an object
- Manipulation 3: remove an object
- Manipulation 4: rotate an object (rotation angle differs between 5° and 180°)
- Manipulation 5: replace an object by another one

The results are shown in figure 5. As can be seen the rates of correctly detected manipulations differ from 29.43% up to 73.66% depending on the kind of manipulation. The binary feature mask was most sensitive to manipulations 1 and 5. The sharpness and brightness of the inserted logo in manipulation 1 and of the object, which replaced another one, in manipulation 5 is different to the rest of the I-frame or significantly different to the removed object. The binary feature mask was less sensitive to manipulation 4. The reason for the loss of sensitivity can be found in the block size of 32x32 pixels. The rotation of a small object inside a block only marginally influences the prominence value of the block.

The best sensitivity in terms of content-changing manipulations was achieved with the following parameter setting:

- Filter: combination of Gaussian and mean value filter with a resolution of 3x3 pixels
- Compute low-pass filtered luminance values with 6 low frequency AC values and the DC value
- Threshold *t* is set to 0.5

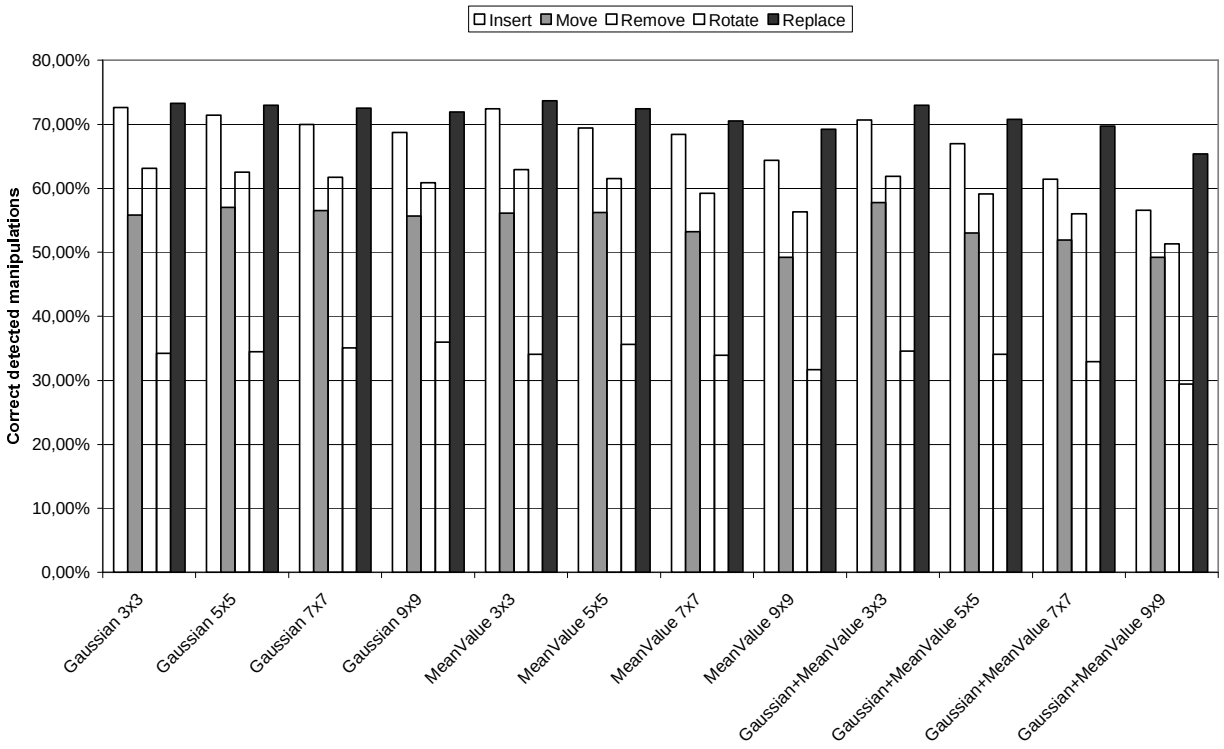

Figure 5: Sensitivity of binary feature mask compared to content-changing manipulations

4.3. Robustness of watermarking scheme

Finally we combined the robust watermark introduced in section 3.2 with the binary feature mask. In order to generate the binary feature mask, we used the following parameters:

- Filter: mean value filter with a resolution of 5x5 pixels
- Compute low-pass filtered luminance values with 6 low frequency AC values and the DC value
- Threshold *t* is set to 0.4

The parameter setting was chosen because it provides a good sensitivity in terms of malicious manipulations while being moderately robust against content-preserving manipulations. We used a group size of 16x48 pixels and 32x16 pixels to embed one bit of the binary feature mask into the I-frame. The group size was chosen because of the capacity of 66 bit being embedded in one frame. Table 1 shows the results for the robustness evaluation compared to compression and re-encoding. The error bit rate is the ratio of different bits between the originally embedded vector and the retrieved watermark in relation to all embedded bits. Our preliminary results show that the DEW scheme is only moderately robust against these content-preserving manipulations. The reason for the high error bit rates can be found in the high capacity we require. The scheme enforces an energy relationship into the groups to embed one bit. To be robust

the energy difference between the upper and the lower part of the group has to be high enough. Especially in videos with large plain areas the energy difference is too low. Hence the DEW scheme is less robust in these videos. As a challenge for the future we will investigate the evaluation of other schemes for embedding the binary feature mask into the videos, e.g. the scheme proposed in [15]. Further optimisation is focussed on applying error correction codes to the robust watermark.

	Re-encoding	Compression 75%	Compression 50%
Group size 16x48 pixel	13,05%	11,00%	27,25%
Group size 32x16 pixel	13,50%	14,08%	32,05%

Table 1: Error bit rates of DEW scheme compared to content-preserving manipulations

5. SECURITY

For the application of our watermarking scheme in military and juristic scenarios it is necessary to analyse the security of the scheme. In this case we have to concentrate on the difficulty to create forgeries, which will not be detected by the watermarking scheme. The scheme provides security in two ways:

1. Randomness for the creation of the binary feature mask

Different levels of security and randomness can be considered when we build the binary feature mask. Our solution consists of selecting arbitrarily groups of blocks using a secret key. We use triangulation as a simple method to select arbitrarily constellations of blocks spanned over the whole I-frame. Figure 6 gives an example. First, we set a particular triangle in the upper-left corner of the I-frame. Then the whole space of the I-frame will be covered with triangles whose dimensions are randomly selected using a secret key. At the end of the covering process all the blocks crossing a given triangle will belong to the same group. These random triangular groups will prevent an attacker from generating forged blocks, which might lead to a forged binary feature mask.

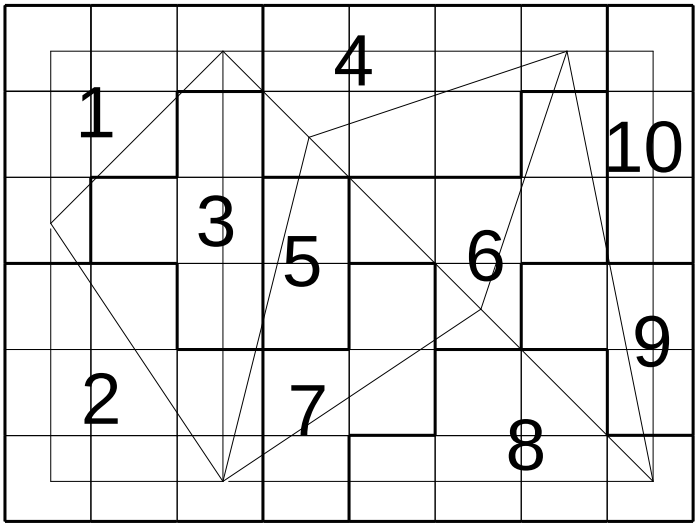

Figure 6: Example for grouping blocks using triangulation

2. Using the low frequency AC coefficients

Using only the low frequency AC coefficients and the DC coefficients for generating the binary feature mask makes it difficult for an attacker to generate a forged image. He has to apply manipulations, which do not affect the middle and high frequencies. Otherwise he would affect the embedded watermark, which will be detected during the integrity verification. It has to be analysed in which way it is possible to create a new object, which does not affect the binary feature mask and the embedded watermark. At the same time the forgery should not contain visual artefacts.

6. CONCLUSION

We have introduced in this work a semi-fragile watermarking scheme for protecting I-frames in MPEG-1/2 videos. The scheme uses the Moravec operator in order to build a semi-fragile binary feature mask, which is embedded robustly in its adjacent I-frame. Preliminary results show the robustness to content-preserving manipulations and fragility to content-changing manipulations. Security can be introduced by using arbitrarily groups of blocks depending on a secret key. Using only low frequency coefficients for generating the binary feature mask provides additional security against forgeries.

As a future work we will focus on increasing the robustness and the fragility of the binary feature mask depending on the manipulation. We will also consider other possible content-preserving manipulations such as increasing the contrast, for instance by histogram equalization. Moreover we have to evaluate whether the watermarking scheme can detect manipulations, which change object colours. Modifying the colour of an object can influence its meaning, e.g. a striped flag. A further challenge is the robustness of the watermarking scheme, which embeds the binary feature mask.

ACKNOWLEDGMENTS

The information in this document is provided as is, and no guarantee or warranty is given or implied that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability. The work described in this paper has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

REFERENCES

[1] B. Schneier: *Applied Cryptography*, 2nd Edition, John Wiley & Sons, ISBN 0-471-11709-9, 1996

[2] C. Y. Lin, S. F. Chang: *Semi-fragile watermarking for authenticating JPEG visual content*, Proceedings of SPIE Vol. 3971, Security and Watermarking of Multimedia Contents II, ISBN 0-8194-3589-9, 2000

[3] A. Puri, T. Chen: *Multimedia Systems Standards and Networks*, Marcel Dekker, Inc., ISBN 0-824-79303-X, 2000

[4] C. Y. Lin, S. F. Chang: *Issues and Solutions for Authenticating MPEG Video*, Proceedings of SPIE Vol. 3657, Security and Watermarking of Multimedia Contents, ISBN 0-8194-3128-1, 1999

[5] K. Maeno, Q. Sun, S. F. Chang, M. Suto: *New semi-fragile image authentication watermarking techniques using random bias and non-uniform quantization*, Proceedings of SPIE Vol. 4675, Security and Watermarking of Multimedia Contents IV, ISBN 0-8194-4415-4, 2002

[6] Y. Dai, S. Thiemert, M. Steinebach: *Feature-based watermarking scheme for MPEG-I/II video authentication*, Proceedings of SPIE Vol. 5306, Security, Steganography, and Watermarking of Multimedia Contents VI, ISBN 0-8194-5209-2, 2004

[7] A. Lang, S. Thiemert, E. Hauer, H. Liu, F. A. P. Petitcolas: *Authentication of MPEG-4 data: risks and solutions*, Proceedings of SPIE Vol. 5020, Security and Watermarking of Multimedia Contents V, ISBN 0-8194-4820-6, 2003

[8] G.C. Langelaar, R.L. Lagendijk, J. Biemond: *Watermarking by DCT coefficient removal: A statistical approach to optimal parameter settings*, Proceedings of SPIE Vol. 3657, Security and Watermarking of Multimedia Contents, ISBN 0-8194-3128-1, 1999

[9] J. Dittmann, S. Fischer, I. Rimac, M. Steinebach, R. Steinmetz: *Combined video and audio watermarking - Embedding content information in multimedia data*, Proceedings of SPIE Vol. 3971, Security and Watermarking of Multimedia Contents II, ISBN 0-8194-3589-9, 2000

[10] P. Yin, H. H. Yu: *A semi-fragile watermarking system for MPEG video authentication*, IEEE International Conference on Acoustics Speech and Signal Processing, Vol. 4, ISSN 0749-8411, 2002

[11] H. P. Moravec: *Towards Automatic Visual Obstacle Avoidance*, Morgan Kaufmann Publishers, 5th Int. Joint Conference on Artificial Intelligence, Vol. 2, 1977.

[12] W. Förstner, E. Gülch: *A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features*, ISPRS Intercommission Workshop, Interlaken, 1987

[13] H. Liu, H. Sahbi, L. Croce-Ferri, M. Steinebach: *Authentication using automatic detected ROIs*, 5th International Workshop on Image Analysis for Multimedia Interactive Services, ISBN 972-98115-7-1, 2004

[14] H. Sahbi, D. Geman, N. Boujemaa: *Face Detection Using Coarse-to-fine Support Vector Classifiers*, Proceedings of the IEEE International Conference on Image Processing, ISBN 0-7803-7622, 2002

[15] S. Thiemert, T. Vogel, J. Dittmann, M. Steinebach: *A high-capacity block based video watermark*, Proceedings of the 30th IEEE EUROMICRO Conference, ISBN 0-7695-2199-1, 2004