Using entropy for image and video authentication watermarks

Stefan Thiemert[a], Hichem Sahbi[b], Martin Steinebach[a]

[a]Fraunhofer IPSI, Darmstadt, Germany;
[b]Cambridge University, Cambridge, United Kingdom

ABSTRACT

There are several scenarios where the integrity of digital images and videos has to be verified. Examples can be found in videos captured by surveillance cameras. In this paper we propose a semi-fragile watermarking scheme, which can be applied on still images as well as on digital videos. We concentrate on the protection of I-frames in compressed MPEG-1/2 videos. We use the entropy of the probability distribution of gray level values in block groups to generate a binary feature mask, which is embedded robustly into an adjacent I-frame. The approach can distinguish between content-preserving and content-changing manipulations. Positions of content-changing manipulations can be localized. We provide experimental results to analyze the effectiveness of the scheme. In the evaluation part we concentrate on the robustness against content-preserving and the sensitivity to content-changing manipulations.

Keywords: video authentication, image authentication, integrity protection, entropy, semi-fragile watermarking

1. INTRODUCTION

In several scenarios it is necessary to check the integrity of digital images and videos, for instance videos captured by surveillance cameras. The latter might be used as evidence in court hearings. Solutions can be found in cryptography as well as in digital watermarking.[1–3] Cryptographic approaches, e.g. digital signatures and hash functions, are secure. However the document will be not authenticated if the format has been changed or lossy compression has been applied. That means that hash functions and digital signatures only verify the binary integrity of documents. For authentication watermarking, fragile, semi-fragile and content-fragile approaches are known strategies. Fragile watermarks are destroyed by slight manipulations on an image or a video cover and therefore can be compared to cryptographic hash functions and digital signatures.[4,5] Semi-fragile watermarking schemes can distinguish between content-preserving and content-changing manipulations. Content-preserving manipulations are applied in post-production processes such as compression or format conversion. Content-changing manipulations remove, insert or replace objects in a single frame or in a sequence.[6] Here the goal is to manipulate the basic message of a video without being perceived by the observer. Both kinds of manipulations are also known as unintentional and intentional attacks.

In this paper we propose a semi-fragile watermarking scheme, which can be applied on still images as well as on videos. We demonstrate the concept on intra-coded frames (I-frames) in compressed MPEG-1/2 videos. From an I-frame we extract a feature, describing the frame content, and generate a binary feature mask. The mask is embedded robustly as digital watermark into the video. With the binary feature mask we can distinguish between content-preserving and content-changing manipulations. The location of content-changing manipulations can be found by comparing the mask extracted from the manipulated I-frame and the mask detected from the watermark. Manipulations in a frame sequence can be detected by additional information embedded as watermark. We provide experimental results to analyze the effectiveness of the scheme.

2. RELATED WORK

Several watermarking schemes are proposed in the literature, which extract visual features[4,7,8] and embed them robustly into the underlying images. For instance [9] use the interest-operator of Moravec to find the most prominent blocks whose location define a binary feature mask. This mask is robustly embedded into the I-frames of videos. Another semi-fragile scheme based on edge features has been proposed in [10] where the

Security, Steganography, and Watermarking of Multimedia Contents VIII, edited by Edward J. Delp III, Ping Wah Wong,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6072, 607218, © 2006 SPIE-IS&T · 0277-786X/06/$15

feature vectors, embedded in JPEG images and I-frames, are used in order to check the authenticity and to localize the content-changing manipulation.

In this paper we use a feature, based on the entropy of the local distribution of gray levels. In [11] Schneider and Chang introduced a content-based watermarking scheme using the intensity histograms of small blocks. Each block is assigned to its mean histogram value, which is embedded as a watermark into the image. The authors show that the approach is able to localize content-changing manipulations and that it is robust against content-preserving manipulations.

The entropy, we use as a feature in this scheme, has been originally defined by C. E. Shannon[12] as the bandwidth of a transmission channel. Later the entropy became a measure for the content of information in information theory. It is often used for estimating the minimal number of bits for compressed information. We use the entropy because of its robustness against content-preserving manipulations, like noise and blurring effects. We assume that the entropy changes significantly after content-changing manipulations.

3. OUR METHODOLOGY

We detect semi-fragile features from a given I-frame and we embed it robustly as a watermark into an adjacent I-frame. We estimate the entropy[12] of local gray level distributions taken from small blocks. Our approach works in three steps:

1. **Feature detection:** given an I-frame I_k, we extract a binary feature mask $M(I_k)$ which consists of the set of locations for which the entropy of the local gray level distribution is estimated.
2. **Embedding:** embed the resulting binary feature mask $M(I_k)$ as watermark into an adjacent I-frame I_{k+1}.
3. **Detection and Comparison:** detect the watermark containing the original binary feature mask $M(I_k)$ from I-frame I_{k+1}. Possible manipulations are localized by comparing the binary feature mask extracted from the current and possible manipulated I-frame $M'(I_k)$ and $M(I_k)$.

3.1. Feature detection

Define a finite set of data $\mathcal{D} = \{x_1, ..., x_N\}$ i.i.d generated according to a probability distribution $P = \{p_1, ..., p_N\}$. Shannon's entropy[12] of P is given by:

$$H(P) = -\sum_{j=1}^{N} p_j \cdot log_2\ p_j \tag{1}$$

From the source-channel point of view[13] $H(P)$ stands for the average size of the code necessary to transmit data from $\mathcal{D}$ when these are generated by P. In our experiments $\mathcal{D}$ is the set of gray level values taken from a reference block (in practice N=256).

Given a block $B(x, y)$, where (x, y) is its location in an image of $m \times n$ pixels. Let $H(B(x, y))$, be the entropy of the gray level distribution in $B(x, y)$. We derive a slight variation of (1) as following:

$$H(B(x,y)) = -\sum_{j=0}^{255} p_j \cdot log_2\ p_j \tag{2}$$

Here $\{p_j\}$ is the probability distribution of the gray level values in $B(x, y)$.

Following these steps, we generate the binary feature mask $M(I_k)$ of the frame I_k:

1. While decoding I_k apply a low-pass filter on the frequency values of luminance DCT blocks. We need to delete the f highest frequencies because of their usage in the watermark embedding process. If we would use those frequencies for generating the binary feature mask the watermark embedding process would influence the result of the feature detection. We do not need to decode the blue and red colour channel, because we only detect the feature from luminance values. This decreases the complexity of the feature extraction process.

2. Apply a grey value blur filter for noise reduction. In practice we apply filters used in common image processing tools, e.g. Gaussian filter and Despeckle filter.

3. Because content-preserving manipulations, like noise and lossy compression, might influence the entropy feature reduce the number of luminance values according to a quantization factor QF to produce frame $Q(I_k)$. After the quantization we have QF different luminance values in $Q(I_k)$.

4. Subdivide $Q(I_k)$ into blocks $B(x, y)$ of size $a \times b$. In practice we choose a and b being equal.

5. For each block $B(x, y)$ estimate the entropy $H(B(x, y))$ according to equation (2).

6. Depending on the capacity of the underlying robust watermark divide the frame into block groups and compute the resulting entropy total or entropy average.

7. Generate the binary feature mask $M(I_k)$ by converting the group entropies into binary numbers.

3.2. Feature embedding

For embedding the binary feature mask $M(I_k)$ we use a robust watermarking scheme. The scheme should embed the data into the d middle and high frequencies of luminance DCT blocks not used for generating $M(I_k)$. A possible scheme has been proposed in [14]. It embeds the information by enforcing an intensity relationship between blocks of 8×8 pixels. For security reasons the blocks are coupled into groups using a secret key. For embedding a 1 into a block group the majority of blocks must have a higher intensity than the mean intensity of the group. In case of embedding a 0 the relationship has to be vice versa. We enforce the relationship by modifying the luminance DCT values of the blocks. Beside the intensity relationship in the block group we embed the bit by enforcing a relationship between the DCT coefficients in each block representing the embedded bit. Other robust watermarking schemes, satisfying the requirements mentioned above can also be used for embedding $M(I_k)$, e.g. the scheme proposed in [15].

The binary feature mask $M(I_k)$ of frame I_k will not be embedded into the frame itself but into the adjacent I-frame I_{k+1}. Ohterwise any content-changing manipulation on I_k would influence the watermark. Hence localizing the manipulated positions would be impossible. Figure 1 shows the feature detection and embedding process.

3.3. Watermark detection and verification

In the verification process we detect the original binary feature mask $M(I_k)$ from I-frame I_{k+1}, e.g. by using the watermarking scheme described in [14]. We couple blocks into groups using the same secret key as in the embedding process. A bit in a block group will be detected by analyzing the relationship between the intensities in the block group as well as the relationship between the DCT coefficients in each block. The user can decide which relationship might be more robust after post-production operations like re-encoding, compression and format conversion. From I-frame I_k being analyzed, we generate a new binary feature mask $M'(I_k)$ according to the steps in 3.1. By comparing $M(I_k)$ and $M'(I_k)$ we can measure the authenticity of the frame. If a block group has been modified by a content-changing manipulation we detect a difference between the masks. We expect that content-changing manipulations significantly influence the entropy in a block group. Figure 2 shows the watermark detection and verification process.

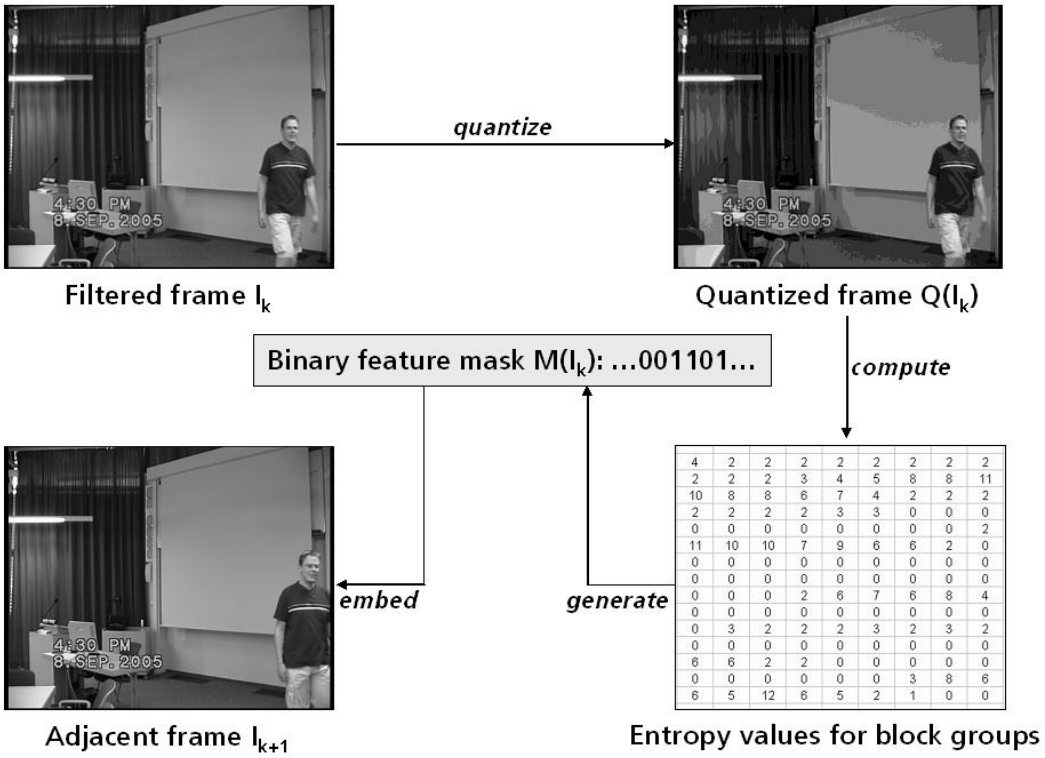

Figure 1. Example for feature detection and embedding process. We applied a low-pass filter (removing the $f = 54$ highest frequency coefficients in an 8×8 DCT block) and a Gaussian blur filter on frame I_k. We quantized the luminance values with $QF = 25$. The entropy values for blocks of 8×8 pixels were divided into groups of 2×3 blocks. The group entropy is the entropy average of all blocks in the group. The resulting vector is embedded into an adjacent frame I_{k+1}.

3.4. Security

Security means that an attacker should not be able to apply content-changing manipulations without the possibility being detected. Hence we have to introduce randomness, based on a secret key. The robust watermark, introduced in [14] uses a secret key to estimate the positions in which the watermark will be embedded. Therefore manipulations on the robust watermark itself will be detectable.

To be secure an attacker must not be able to generate a forged I-frame with the same feature vector. As mentioned in 3.1 we remove the f highest frequencies of a DCT block during the generation of $M(I_k)$. Manipulations, which will not be detected, must be applied without changing those frequency coefficients, which is difficult. Simply copying the middle and high frequency DCT coefficients to the forged I-frame may cause visual distortions. Furthermore we use arbitrarily shaped block groups, which is a concept we introduced in [9]. Based on a secret key we compute a triangulation, which leads to random arbitrarily shaped groups.

4. EXPERIMENTAL RESULTS

In our experiments we concentrate on the robustness against content-preserving manipulations and on the sensitivity to content-changing manipulations. With the experiments we want to find a trade-off between both requirements, which compete against each other.

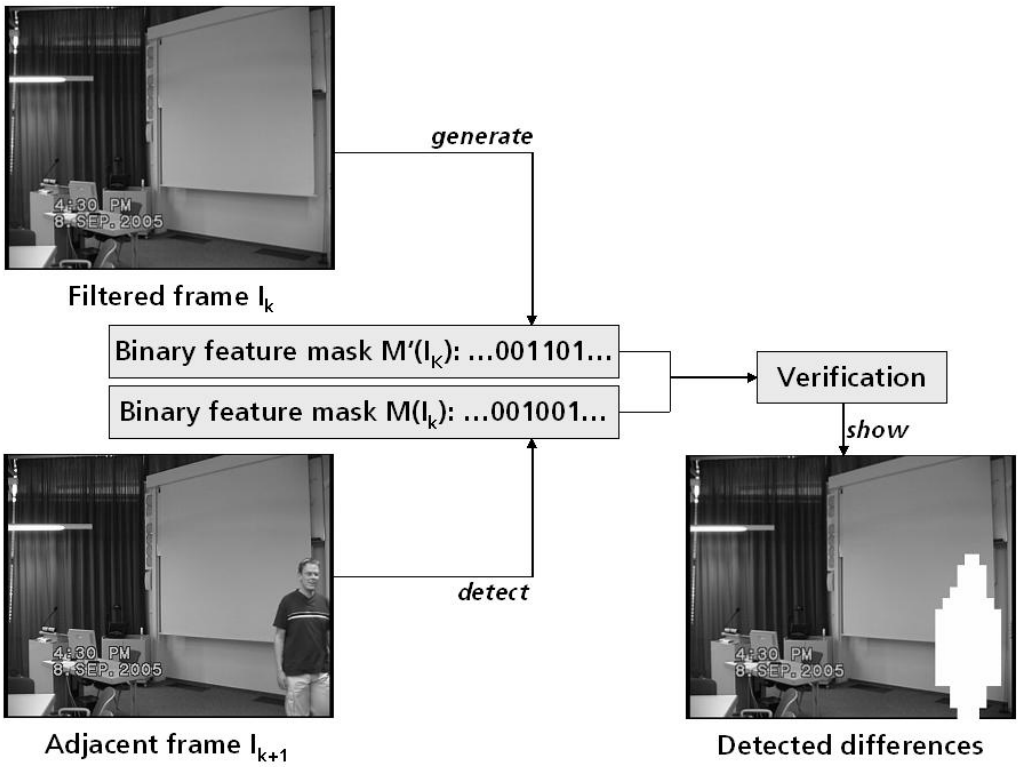

Figure 2. Example for watermark detection and verification process. The parameters used for producing the binary feature masks $M(I_k)$ and $M'(I_k)$ are the same as in figure 1.

4.1. Robustness

For robustness evaluation we use still images, converted into MPEG-1 videos. The resolution is fixed at 352×288 pixels with a bit rate of 1.125 kBit/s. We used the following parameter settings for evaluation:

- Resolution: 1×3 blocks of 8×8 pixels (average of entropy values) [$RES = 1$], 2×3 blocks of 8×8 pixels (total of entropy values) [$RES = 2$], 1 block of 16×16 pixels [$RES = 3$], 1 block of 32×32 pixels [$RES = 4$]
- QF: 10, 25, 50
- DCT coefficients (block of 8×8 pixels in Zig-Zag order): 0-4, 0-9, 0-19, 0-31
- Blur filter: Despeckle, Gaussian

We applied re-encoding and lossy compression to various bit rates (down to 50% of original bit rate) to a set of 100 videos. In the verification process we analyzed the correlation between the binary feature masks of the original and the manipulated frames. The experimental results can be found in the tables 1 and 2. We ordered the resulting bit error rates according to resolution index RES and quantization factor QF. The results show that the entropy feature is partially robust against the content-preserving manipulations. We observed that the robustness mainly depends on the resolution RES and the quantization factor QF. The kind of blur filter and the number of DCT coefficients have no wide influence on the robustness. We achieved the best results for a block size of 32×32 pixels and a quantization factor $QF = 50$. Here the mean error bit rate (estimated over 100

videos) differs between 0.98% for re-encoding and 1.86% for compression down to 50% of the original bit rate. Based on the experimental results we can conclude that big block groups and a high quantization factor make the feature more robust against content-preserving manipulations.

4.2. Sensitivity

In this section we focus on the sensitivity of the entropy feature to content-changing manipulations, i.e. manipulations on objects. Single objects are manipulated when they are moved, removed, pasted, substituted or rotated. We applied those manipulations on the set of still images as well as on a set of videos.

First we computed the error bit rates of the binary feature mask after a content-changing manipulation. We compared the results with the error bit rates after a re-encoding process. An example for moving and rotating objects compared to re-encoding are shown in figure 3. The error bit rates increased significantly after content-changing manipulations. For the parameter combination with the best robustness against content-preserving manipulations the error bit rate (estimated over 100 videos) increased from 0.98% to 1.99% after rotating objects. Hence a higher error bit rate is a first indicator for a content-changing manipulation.

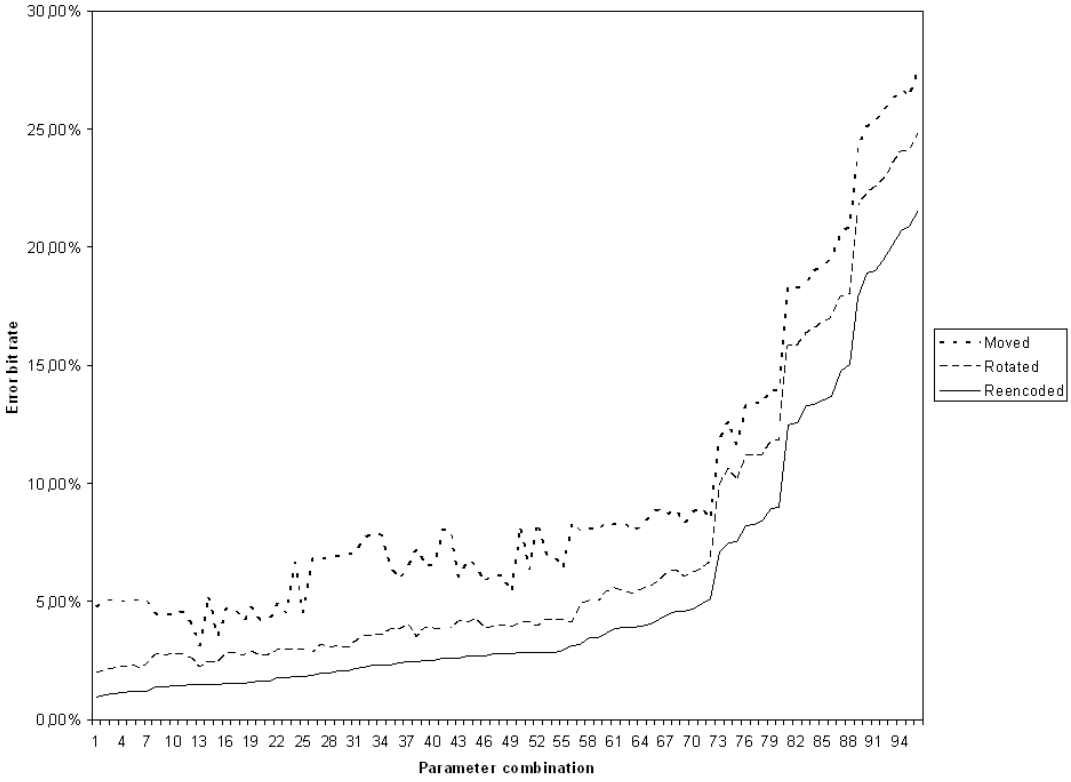

Figure 3. Error bit rates of the binary feature mask after moving and rotating objects compared to the error bit rate after re-encoding. The rates are computed for different parameter combinations.

The results for the set of still images are shown in the tables 3 and 4. We define False Acceptance Rate (FAR) and False Rejection Rate (FRR) according to the equations (3) and (4):

$$FAR = \frac{Number\ of\ not\ detected\ manipulated\ videos}{Number\ of\ manipulated\ videos} \tag{3}$$

$$FRR = \frac{Number\ of\ not\ manipulated\ blocks\ verified\ as\ not\ authentic}{Number\ of\ not\ manipulated\ blocks} \quad (4)$$

Here we define that a manipulation in a video has been detected if at least 50% of the manipulated blocks are verified as not authentic.

Applying the parameter settings we used in 4.1 we observed that the semi-fragile feature is very sensitive to pasting objects but moderatly sensitive to the rotation of objects. Obviously pasting, removing and moving objects changes the entropy in a block group significantly. Substitution will only be detected if the new object, pasted on the position of the old object contains another texture. Rotating an object does not change the probability distribution of the gray level values in a block group. Hence the entropy does not change significantly. Again the number of DCT coefficients and the kind of blur filter had no wide influence on the sensitivity.

Finally we generated from 6 public domain videos* the binary feature masks. For simplicity we applied content-changing manipulations on the uncompressed frames. In detail we removed, pasted and substituted objects. Afterwards we compared the orignal binary feature mask $M(I_k)$ with the new binary feature mask $M'(I_k)$. Figure 4 shows some examples. The binary feature mask reacted very sensitive to the applied manipulations. In each case of a removed and pasted object the manipulation has been localized correctly. The colour modification in the third case (4c) was not detected. Here the manipulation did not change the probability distribution of the gray level values. This problem could be solved by extending the scheme to colour values.

5. CONCLUSION AND FUTURE WORK

This paper introduces a semi-fragile watermarking scheme for the protection of I-frames in MPEG-1/2 videos. We use the entropy of the probability distribution of gray level values in block groups to generate a binary feature mask, which is embedded robustly into an adjacent I-frame. First results show that the scheme is robust to content-preserving manipulations and sensitive to content-changing manipulations. Problems where observed regarding the sensitivity to rotation.

An extended analysis of possible content-preserving and content-changing manipulations has to follow in further research activities. Further content-preserving manipulations can be sharpening and additive Gaussian noise.[6] The focus has to be set on the extension of the scheme to colour values so that manipulations like in figure 4c can be detected. Another challenge is the improvement of the robustness and sensitivity. Finally the approach has to be combined with a robust digital watermark for videos. Here we have to find a trade-off between the robustness of the watermark and the capacity requirements of the binary feature mask. Another aspect for video integrity is the protection of the frame order. This functionality has to be added to the scheme framework. An approach for the protection of the frame order has been proposed in [9].

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.

*ftp://ftp.tek.com/tv/test/streams/Element/index.html

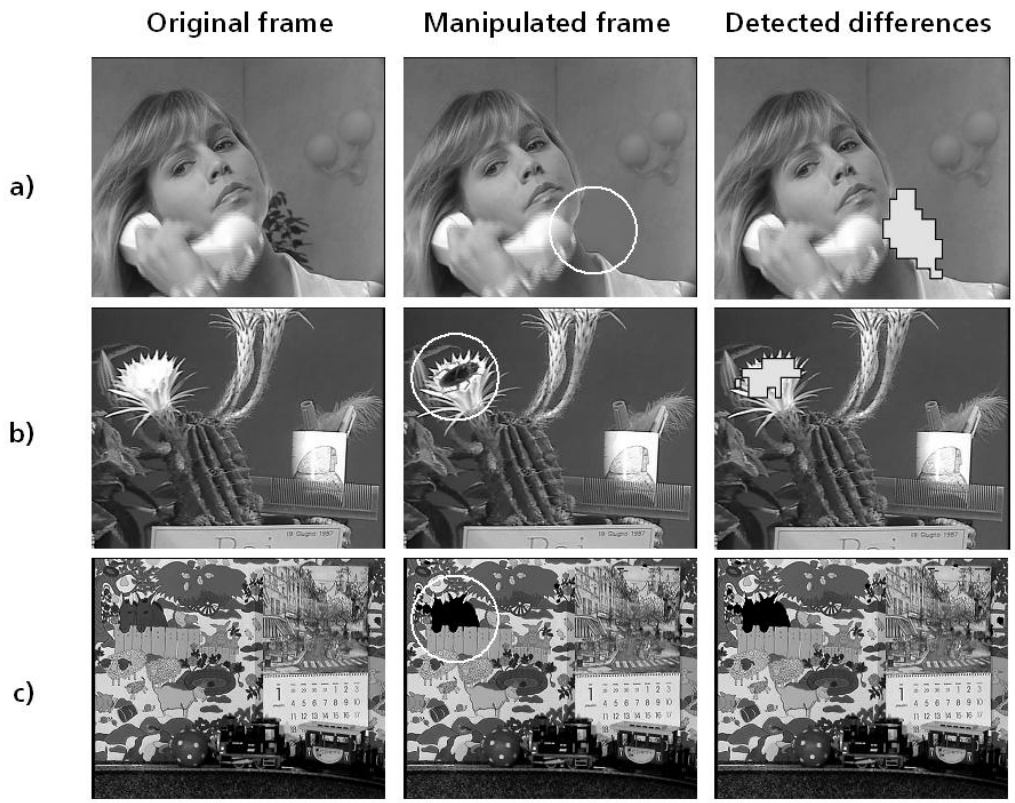

Figure 4. Examples for detecting content-changing manipulations. **a)** A plant in the background has been erased. **b)** An insect has been pasted on the cactus. **c)** The colour of the horses has been changed to a darker colour which is similar to the substitution of an object.

REFERENCES

1. B. Schneier, *Applied Cryptography, 2nd Edition*, John Wiley & Sons, 1996.
2. C.-Y. Lin and S.-F. Chang, "Issues and solutions for authenticating MPEG video," in *Electronic Imaging: Security and Watermarking of Multimedia Contents*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **3657**, pp. 54–65, 1999.
3. M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Video authentication with self-recovery," in *Electronic Imaging: Security and Watermarking of Multimedia Contents IV*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **4675**, pp. 531–541, 2002.
4. J. Fridrich, "Security of fragile authentication watermarks with localization," in *Electronic Imaging: Security and Watermarking of Multimedia Contents IV*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **4675**, pp. 691–700, 2002.
5. E. Lin and E. Delp, "A review of fragile image watermarks," in *Proceedings of the ACM Multimedia and Security Workshop*, pp. 25–29, 1999.
6. O. Ekici, B. Sankur, B. Coskun, U. Naci, and M. Akcay, "Comparative evaluation of semifragile watermarking algorithms," *Journal of Electronic Imaging* **13**, pp. 206–216, 2004.
7. Y. Dai, S. Thiemert, and M. Steinebach, "Feature-based watermarking scheme for MPEG-I/II video authentication," in *Electronic Imaging: Security, Steganography, and Watermarking of Multimedia Contents VI*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **5306**, pp. 325–335, 2004.

8. C.-Y. Lin and S.-F. Chang, "Semi-fragile watermarking for authenticating JPEG visual content," in *Electronic Imaging: Security and Watermarking of Multimedia Contents II*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **3971**, pp. 140–151, 2000.
9. S. Thiemert, H. Sahbi, and M. Steinebach, "Applying interest operators in semi-fragile video watermarking," in *Electronic Imaging: Security, Steganography, and Watermarking of Multimedia Contents VII*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **5681**, pp. 353–362, 2005.
10. J. Dittmann, S. Fischer, I. Rimac, M. Steinebach, and R. Steinmetz, "Combined video and audio watermarking - embedding content information in multimedia data," in *Electronic Imaging: Security and Watermarking of Multimedia Contents II*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **3971**, pp. 455–464, 2000.
11. M. Schneider and S.-F. Chang, "A robust content-based digital signature for image authentication," in *Proceedings of the 1996 IEEE International Conference on Image Processing*, 1996.
12. E. S. C and W. Weaver, *The Mathematical Theory of Communication*, University of Illinois Press, 1963.
13. T. Cover and J. Thomas, *Elements of Information Theory*, Wiley & Sons, 1991.
14. S. Thiemert, T. Vogel, J. Dittmann, and M. Steinebach, "A high-capacity block based video watermark," in *Proceedings of the 30th EUROMICRO Conference*, 2004.
15. T. Kalker, G. Depovere, J. Haitsma, and M. Maes, "A video watermarking system for broadcast monitoring," in *Electronic Imaging: Security Watermarking of Multimedia Contents*, E. J. Delp III and P. W. Wong, eds., *Proceedings of SPIE* **3657**, pp. 103–112, 1999.

Table 1. Error bit rates, ordered by resolution index RES. The values in brackets show the best values received for the resolution index.

RES	Reencoded	Compression 75%	Compression 50%
1	2.92% (1.48%)	4.41% (2.27%)	6.34% (2.96%)
2	13.84% (7.09%)	20.06% (10.51%)	27.70% (13.24%)
3	2.64% (1.38%)	3.74% (1.91%)	5.07% (2.35%)
4	1.95% (0.98%)	2.46% (1.33%)	3.31% (1.86%)

Table 2. Error bit rates, ordered by quantization factor QF. The values in brackets show the best values received for the quantization factor.

QF	Reencoded	Compression 75%	Compression 50%
10	7.67% (2.20%)	11.10% (2.90%)	15.39% (3.88%)
25	5.25% (1.85%)	7.41% (2.20%)	10.20% (2.78%)
50	3.10% (0.98%)	4.49% (1.33%)	6.22% (1.86%)

Table 3. FAR and FRR, ordered by resolution RES.

	Move	Remove	Paste	Subsitute	Rotate
RES	FAR (FRR)	FAR (FRR)	FAR (FRR)	FAR (FRR)	FAR (FRR)
1	63.32% (2.95%)	55.82% (3.11%)	20.15% (2.94%)	73.85% (2.93%)	92.22% (2.84%)
2	12.25% (14.00%)	8.58% (14.75%)	2.93% (13.81%)	14.11% (13.88%)	33.50%(13.56%)
3	50.08% (2.66%)	48.10% (2.76%)	15.94% (2.63%)	69.41% (2.64%)	92.48%(2.59%)
4	59.28% (1.96%)	53.58% (2.04%)	21.67% (1.95%)	78.40% (1.96%)	92.09%(0.89%)

Table 4. FAR and FRR, ordered by quantization factor QF.

	Move	Remove	Paste	Subsitute	Rotate
QF	FAR (FRR)	FAR (FRR)	FAR (FRR)	FAR (FRR)	FAR (FRR)
10	32.96% (7.70%)	27.59% (8.15%)	6.59% (7.73%)	51.55% (7.71%)	71.29%(7.30%)
25	45.36% (5.34%)	39.59% (5.56%)	14.35% (5.21%)	58.11% (5.25%)	77.58%(4.87%)
50	60.38% (3.13%)	57.38% (3.29%)	24.57% (3.06%)	67.16% (3.09%)	83.85%(2.74%)