Robust MPEG Video Watermarking Technologies

Jana Dittmann
GMD - German National Research Center for Information Technology, Institute (IPSI)
Dolivostraße 15,
D-64293 Darmstadt, Germany
+49-6151-869-845
dittmann@darmstadt.gmd.de

Mark Stabenau
GMD - German National Research Center for Information Technology, Institute (IPSI)
Dolivostraße 15,
D-64293 Darmstadt, Germany
+49-6151-869-845
stabenau@darmstadt.gmd.de

Ralf Steinmetz
Darmstadt University of Technology, Industrial Process and System Communications
Merckstr. 25,
D-64283 Darmstadt, Germany
+49-6151-166151
Ralf.Steinmetz@KOM.tu-darmstadt.de

1. ABSTRACT

The development of new multimedia services and environments requires new concepts both to support the new working process on distributed computers and to protect the multimedia data during the production and the distribution in digital marketplaces. This article addresses copyright protection as a major security demand in digital marketplaces. We propose and compare two watermarking techniques for MPEG video with the intention to show the advantages and the possible weakness in the schemes working in the frequency domain and in the spatial domain. To improve the view to the distortion of the watermarked frames we generate a 3D-difference view measuring the changes which were made during watermarking process and/or caused by several damaging attacks.

1.1 Keywords

Security and the media, digital watermarking for MPEG video, copyright protection

ACM Multimedia'98, Bristol, UK
© 1998 ACM 1-58113-036-8/98/0008 $5.00

2. Motivation

In the A4SM project, a new production environment currently under development at GMD-IPSI, a variety of security features are being developed to increase the users' acceptance. A detailed security risk model can be found in [4]. One major aspect is copyright protection. The video production process leads to a final product based on individual ideas, and the result is an unique intellectual creation. With the digital representation of the video, the procuders run the risk of suffering disadvantages like direct financial loss, legal problems and image loss. Problems include unauthorized taping, reading, manipulating or removing data. Designers, producers and publishers of video or multimedia material are therefore seeking technical solutions to the problems associated with copyright protection of multimedia data.

In this paper we propose and compare two watermarking techniques for MPEG video with the intention to show the advantages and the possible weakness in the schemes working in the frequency domain and in the spatial domain. The main concept of the approaches is to provide environments where digital videos can be signed by authors or producers as their intellectual property to ensure and prove ownership rights on the produced video material during its distribution. The most existing systems are mainly used for still images, [1], [2], [3], [10], or the copyright system SysCoP developed by [8]. The rarely existing video based technologies mostly lack robust continuous signing of all video frames. The following projects are related to digital video copyright protection, the Watermarking DataBlade™ from NEC (1996) or the digital watermarking of MPEG-2 Coded Video in the Bitstream Domain from the University of Erlangen, Germany,[7]. Our work focuses on the implementation and evaluation of robust watermarking technologies for MPEG video with the intention to embed watermarks in every encoded video frame. In the first section

we describe two existing watermarking techniques for images and our main intentions to adapt these algorithms to design a video based algorithm based on the existing schemes. We continue with the description of the experimental system, our improvements and tests results. Our tests mainly based on compression, format conversions and StirMark attacks, [8]. Finally, we assess our achievements so far, and provide an overview of further work.

3. Digital Watermarking

Digital watermarking is the enabling technology to prove of ownership on copyrighted material, detect the originator of illegally made copies, monitor the usage of the copyrighted multimedia data and analyse the spread spectrum of the data over networks and servers. Our goal is to design an algorithm which can be used for all these features and embeds every kind of coded information typically binary coded words.

Basically, watermarks, labels or codes embedded into multimedia data for enforcing a copyright must uniquely identify the data as property of the copyright holder, and must be difficult to be removed, even after various media transformation processes. Thus the goal of a label is to always remain present in the data. Today the existing labelling techniques have different security problems regarding robustness and visual artefacts, [2], [8]. In order to prevent any copyright forgery, misuse or violation, the key to the copyright labelling technique is to provide security and robustness of the embedded label against a variety of threats which include:

- Detecting embedding locations by comparing differently labelled versions of the same original material.
- Finding and altering the embedded label through visual or statistical analysis.
- "The IBM-attack": Instead of introducing a new watermark with an own algorithm and claiming the authorship, a counterfeit original of a watermarked picture is produced by removing a watermark, thus claiming that the original of the real owner contains the watermark which we removed.
- Damaging or removing the embedded label using common multimedia processing.

Mainly we want to address the last point because MPEG compression itself performs multimedia processing like lossy compression or scaling. All necessary transformations on the frames can lead to a distortion of the embedded information and the label cannot be retrieved by the owner. The two presented techniques are adapted versions of algorithms by [8] and [6] for still images previously published. Our work seeks to address MPEG format specifications to provide robust digital watermarking mechanisms for distributed video production systems.

3.1 Requirements for MPEG Video Watermarking

In this paper we address the video part, MPEG standard 1993, [11]. The MPEG compression algorithms employ Discrete Cosine Transform (DCT) coding techniques on image blocks of 8x8, prediction and motion compensation. The resulting output stream contents a sequence of I-, P- and B-frames.

Following requirements are considered important for MPEG video watermarking:

- Robustness against high compression rates of the DCT compression, motion compensation and prediction (very important)
- Robustness against scaling (very important)
- Labelling of every single video frame (I-, P- and B-frames) to provide continuous watermarking and avoid attacks of cutting single frames
- Ensuring correct decoding of the frame sequences without visual artefacts (remark: changes of an I-frame influences the following coding of B- and P-frames)
- Runtime, performance for streaming video or stored video (because today our environment do not need streaming video watermarking we do not regard performance in our first implementations)

Before we are describing the experimental system we describe the two basic algorithm, their advantages and disadvantages and our new implementation strategy:

4. The Zhao Koch Algorithm

The Zhao-Koch algorithm, [8], embeds the copyright label in the frequency domain. Originally, the luminance information Y in the spatial domain is discrete cosine transformed (DCT) into the frequency domain and then quantized. The algorithm pseudo randomly chooses three coefficients from the quantized DCT encoded block and manipulates them to store a single bit information of the copyright label (like binary coded name or address of the owner) using a secret key. For embedding the 1 or the 0 bit Zhao and Koch define different patterns with High, Middle and Low as manipulation rule, see [8]. If storing a bit of information requires a significant change in the coefficients of a block, then the coefficients are manipulated to form an invalid pattern to tell the retrieval there is no information embedded in that block. Generally the invalid pattern requires less of changes in the coefficients than encoding a 0 or 1.

During extraction process, the same coefficients are pseudo randomly selected using the secret key and the relationship between the coefficients are analysed. Depending on the relationship a 0 or 1 is extracted.

The algorithm does not need the original image for retrieval. An advantage is, that the watermark information is embedded in the compressed domain and can be easily

applied to MPEG compressed video with minimal operations.

Despite these advantages the algorithm has a few shortcomings: every block is modified and artefacts are common especially in smooth blocks or in sharp edges. The algorithm is not robust against scaling or rotation because the image dimension is used to generate a appropriate pseudo random sequence. **Our goal** is to evaluate the behaviour with MPEG compression and the coding in P- and B-frames. Our improvements address the visual distortion mainly to keep the high quality of the video and to prevent selective attacks on the watermark using efficient error correcting codes. We use smooth block and edge recognition schemes to avoid artefacts.

5. The Fridrich-Algorithm

The method is based on overlaying a pattern with its power concentrated mostly in low frequencies. The pattern is created using a pseudo random number generator and a cellular automaton with voting rules. The robustness of the method has been shown by Fridrich, [6]. The method also overcomes a possible weakness of the method of [2] which can be attacked by using the fact that areas of the image which are almost uniform or have an almost constant brightness gradient may show a portion of the watermark pattern. To overcome this weakness, Fridrich uses a watermarking method based on pattern overlaying. Since the pattern will be formed in a sensitive way based on the watermark sequence, even if the watermark pattern shows in uniform areas, it is not possible to mount an attack. The watermark bit sequence is used for initialising a pseudo-random generator to create a random black and white initial pattern of the same size as the image. A cellular automaton with voting rules is applied till a convergence to a fixed point is obtained. The voting rule coalesces random patches into connected areas. The pattern is further filtered by a smoothing filter to move the main portion of the power to low frequencies. The gray levels of the final pattern are scaled to a small range and the pattern is finally added to the image. The watermarked image shows no visible degradation caused by the overlaid pattern, yet the pattern is embedded in a robust sense. It is possible to prove the presence of the pattern in images after filtering, JPEG compression with as low as 5% quality factor, cropping, resampling, blurring, down-sampling, and noise adding. The watermark also appears to be resistant with respect to the collusion attack (averaging several watermarked images to remove the watermark).

The first main disadvantage is, that the retrieval process requires the original, un-watermarked image. For videos this is not acceptable because we would need the whole video to prove the watermark. Our algorithm fixes this shortcoming using plain statistical techniques to retrieve the label without the origin.

The second main disadvantage is that the watermarking algorithm embeds only one information: the pattern created using a pseudo-random number generator and a cellular automaton with voting rules. There is no detailed information about the author or producer embedded. The retrieval process provides only true or false if the pattern was retrieved successfully. Our goal is to extend the algorithm in a way that we can embed code words for detailed information like author name or address.

6. Experimental System - MPEG Watermarking

The experimental systems adapt the strength of the embedded watermark to the HVS-properties using two parameters instead of uniformly modulate the luminance values: smoothness and edge character of the block. The edge characteristics are mainly based on the analysis of DCT values so that we can mostly detect vertical and horizontal edges. Smoothness and edge character are the two main parameters. These are concerned in both algorithms and the expected visibility of the watermark can be calculated (as described later), resulting in a value which can be interpreted as the capability of the block to incorporate the watermark without visual distortions. Before the watermarking starts, the MPEG video is traversed with the decoding and single frames (frame data) are produced. The information to be embedded (label data) is encrypted with a secret user key and then embedded into the image data with the same user key used as seed for a pseudo random number generation. The general embedding scheme of both implementations is shown in the next figure:

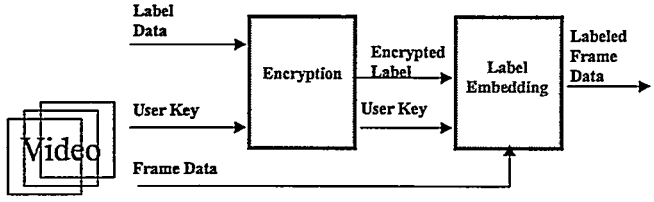

Figure 1: General Embedding Scheme

Before the watermarking starts, the MPEG video is traversed with the decoding and single frames (frame data) are produced. The information to be embedded (label data) is encrypted with a secret user key and then embedded into the image data with the same user key used as seed for a pseudo random number generation. The retrieval is performed with the inverse steps shown in the following figure:

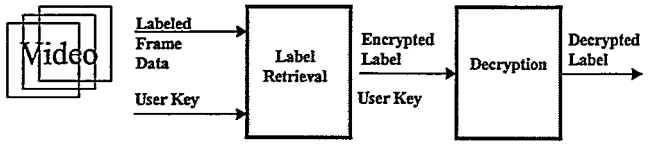

Figure 2: General Retrieval Scheme

Before the retrieval starts, the video is traversed again with the decoder and single frames are produced. In the next chapter we describe the detailed embedding and retrieval steps separated into the two used algorithms.

6.1 Approach I in the DCT Domain

6.1.1 Embedding method

The embedding of the label data is performed in three steps. Originally Zhao and Koch have used only two steps, the first and the third one. We have integrated a second step to improve the visual quality of the watermarked frame and integrate an error correcting code.

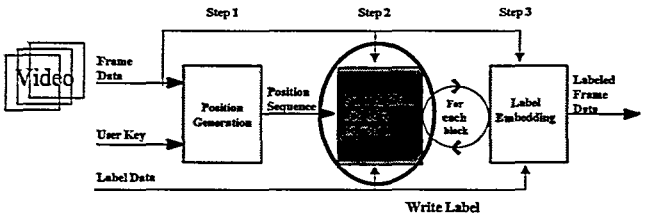

Figure 3: Improved Embedding Scheme

In the first step a position sequence is generated from the user key as a seed with a secure random number generator. This is necessary to hide the watermark in the frame. In the order of the generated position sequence every block is now discrete cosine transformed. The second step consists of the smooth block and edge detection as mentioned earlier. Although sophisticated techniques were developed to check for HVS-characteristics the calculation of the smoothness and the edge character of the block is kept quite simple. This is due to the fact that in future versions this calculation has to be done in the MPEG-stream domain and hopefully in real-time. The parameter *smooth* is simple the number of DCT-coefficients which are not zero after quantization with the quantization matrix Q_m, seen in the following matrix. Thus, high values of smooth indicate many frequency components and therefore a great visual tolerance against additional distortions through the watermark.

low	16	11	10	16	24	40	51	61	
	12	12	14	19	26	58	60	55	
	14	13	16	24	40	57	69	56	
	14	17	22	29	51	87	80	62	
	18	22	37	56	68	109	103	77	
	24	35	55	64	81	104	113	92	
	49	64	78	87	103	121	120	101	
	72	92	95	98	112	100	103	99	High

Unfortunately blocks with edge characteristics often have a lot of frequency components, too. Thus a second parameter *edge* is introduced additionally to reduce artefacts.

The parameter *edge* is calculated as simple as *smooth*: *edge* is the sum of the absolute values of the DCT-coefficients 1, 2, 8, 9, 10, 16, 17 as marked in Q_m, which represents the lower DCT frequencies. High values in these components indicate that the block could have edge characteristics. To determine the level of tolerance against distortions through the watermark caused by each of the two parameters a linear combination is made: *Level = smoothscale*smooth + edgescale*edge + offset*

The parameter *offset* is needed for a base strength of the watermark. The linear combination can now be imagined as a watermark strength indicated by offset and slight variations in strength in dependence of the block characteristics weighted with the parameters *smoothscale* and *edgescale.*

smoothscale = -10, edgescale= 0.27 and *offset=50* were evaluated through experiments.

Because *Level* can have negative values, *Level* is restricted to values between 0 and 50:

If Level>50 Level=50

If Level<0 Level = 0

So far the level-estimation is independent from the used watermarking algorithm.

To determine the strength of the watermark in dependence of *Level* an additional quantization-factor Q_f is used in the Zhao-Koch algorithm. Every change to DCT-values are made on the originally DCT-value quantized with Q_m/Q_f. Therefore if a change is made to a quantized DCT-value with Q_f=1 this lead to a 4 times higher change than with Q_f=4. Q_f is calculated from *Level* through a table-look-up, because of the not necessarily linear correlation and the small range of Q_f:

Q_f	1	1	2	3	4	4
Level/10	0	1	2	3	4	>4

Before embedding can start an error correcting code is created in the following way: every watermarking information letter is coded into 5 bits first. The resulting bitstream is then (31, 6, 15)-BCH encoded. To ensure improved redundancy every 31 bit word is inserted repeatedly. The parameter bit redundancy determines the amount of redundancy.

In the third step the watermark information with the error corrections and redundancy is embedded as described in the Zhao-Koch algorithm. From each quantized DCT-block three locations in the medium frequencies with absolute values Y1, Y2 and Y3 are chosen, where the bit should be inserted. To encode the bit the three values were changed to one of the following patterns:

Bit	1	1	1	1	0	0	0	0
Y1	H	H	M	M	L	L	M	M
Y2	H	M	H	M	L	M	L	M
Y3	L	L	L	L	H	H	H	H

If the changes to embed the Bit are too big, so that visual distortions are common, Y1, Y2 and Y3 are changed to an invalid pattern (H,L,M; L,H,M ; M,M,M). After the

changes are made the block is requantized and inverse discrete cosine transformed.

6.1.2 Retrieval Method

The retrieval is performed in the same way of the Zhao and Koch algorithm. We decode the single frame of the video and perform the inverse steps of the embedding: first the position generation and the retrieval of the embedded data.

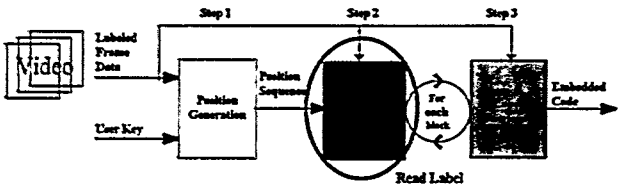

Figure 4: Improved Retrieval Scheme

The first two steps are exactly as in the embedding process. Step 2 is not essential but the information about the strength of the watermark in each block is helpful while using the described error correcting and redundancy code.

In the third step the same three locations from every block must be examined like the ones used in the embedding process. Then the patterns could be checked and the watermark bit could be read out.

6.1.3 Experimental Results

We have tested our adapted implementation of the Zhao-Koch algorithm with different MPEG-Videos. One example should demonstrate the capabilities and the shortcomings of the algorithm. The results of the first 15 frames of the video are shown. The table contains the error rates after MPEG-reencoding, format conversion to QuickTime and after the watermark removing program StirMark is used [9]. StirMark combines various attacks. It simulates distortions caused by a printing and rescanning process. Furthermore it introduces some minor geometric distortions like stretching, shearing, rotations and shifting. It is reported that StirMark is very effective against most even commercial watermarking techniques. However, the distortions introduced by StirMark are unrecognisable.

The chosen video museum.mpg is about a virtual museum. A camera leads from the entrance from the museum through several rooms. The first 30 frames show a short zoom from the entrance of the museum. For a better view the HTML version of the paper can be found in: www.darmstadt.gmd.de/mobile/watermarking with the images and video sources used in this paper.

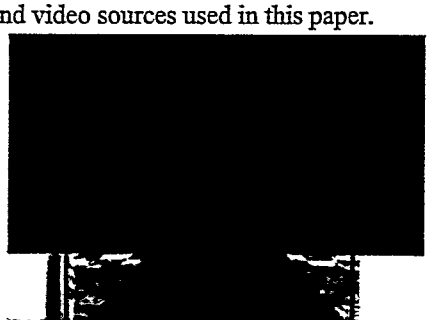

Figure 5: Original museum (first frame)

Video characteristics:		Museum.mpg
No. of 8x8 blocks		1320
Compression I-Frame		3.42%
Compression P-Frame	2.65%	
Compression B-Frame	1.22%	
IPB-order		IBBPBB

Compression-rates are calculated from uncompressed 24 bit images and the algorithm uses the following parameters: smoothscale = -10, edgescale = 0.43 and offset = 40, watermark strength: 2.0 and bit redundancy 4. Conversion table from Level to Quantization factor Q_f:

Q_f	1	1	2	3	4	4
Level/10	0	1	2	3	4	>4

These parameters leads to a watermark strength which cause slight artefacts in uniform regions. To improve the view to the distortion chapter 6.2.3 offers a 3D-difference view measuring the changes. In summary the strength of the algorithm can be assumed as high. The visual quality improvements can be evaluated at www.darmstadt.gmd.de/mobile/media/watermarking in the HTML version of this paper. The original Zhao algorithm Koch with the default settings produces several artefacts in the smooth regions and throughout brighter frames.

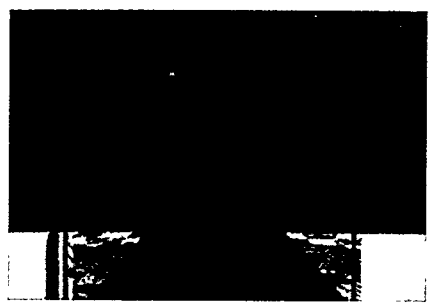

Figure 6: Watermarked museum (first frame)

Our robust test results are dispayed in the next table. We embedded 60 Bits of watermarking information. We performed MPEG-encoding, Quicktime transformation and StirMark-Attack and got following results with the improved Zhao-Koch-algorithm. The table measures the bit errors after the performed transformations with the error correcting code. The numbers show the amount of bit-errors occurred in the first 13 frames after high MPEG compression, QuickTime conversion and Stirmark attack.

Frame No.	0	1	2	3	4	5	6
F-Type	I	B	B	P	B	B	I
MPEG BCH	0	0	0	0	0	0	0
QuickTime BCH	0	1	9	1	0	0	0
StirMark BCH	22	29	29	16	17	19	24

Frame No.	7	8	9	10	11	12	13
F-Type	B	B	P	B	B	I	B
MPEG BCH	0	0	0	0	0	0	0
QuickTime BCH	0	0	0	1	0	0	5
StirMark BCH	22						

Table 1: Absolut Errors

The following bit error rates can be measured all together of our experiments (about 10 video streams):

museum.mpg:
MPEG:
I-Frames <1% P-Frames 1-2% B-Frames 5%
QuickTime:
I-Frame 1-2% P-Frames 5% B-Frames 7%
StirMark: 32%

Table 2: Error Rates

Regarding the error rate table we want to discuss our results: the first apparent thing is that the algorithm shows very good results in MPEG compression and Quicktime conversions. The error rates of the watermark information in I-Frame is excellent. B-Frames have still some problems. StirMark removes the watermark up to 30 percent, without the error correcting code the watermark was destroyed completely. Stirmark destroys the watermark in the original Zhao-Koch completely. Visual artefacts can be avoided.

6.2 Approach II in the Spatial Domain

The proposed watermarking technique of Fridrich is modified in the following way to overcome the two main shortcomings: retrieval without the origin and embedding of binary coded labels. To ensure the last requirement instead of one overlaying pattern over the whole frame we add a 8x8 pattern over every 8x8 Block of the frame. To embed binary code words we define additional modification rules of the overlaying 8x8 pattern described in the following embedding strategy. Furthermore we use statistical properties to find the label in the retrieval procedure without the original frame.

6.2.1 Embedding method

First of all a position sequence like the one used in the Zhao-Koch algorithm is generated to determine the blocks we want to modify. The next figure illustrates the embedding steps. For each block a user key dependent pattern is made in the following manner: We start by creating a 8x8 pseudo random pattern with the user key as a seed, step 1. To eliminate the high frequencies in this pattern a cellular automaton with simple voting rules is used. Every position in the 8x8 random pattern is tested on the number of '1' in the eight co-sited positions. If the number exceeds five the actual position is set to '1' too, if the number is less than 3 it is set to '0', see the marked rectangle for an example. By applying these rules several times on the whole 8x8 block we obtain a pattern M with less high frequencies, steps 2 - 4. Now a correlation between the pattern and the luminance block has to be inserted according to the bit we want to embed, step 5. If we want to embed a '1' we add a value k, which is calculated in the smooth/edge block estimation routine via a table look up from *Level*, in each luminance block position where the corresponding position in the pattern M is '1' and we subtract the value k if the corresponding position is '0'. If we want to embed a '0' we do it vice versa.

Due to the fact that we use much smaller patterns (8x8) than Fridrich (one pattern for the whole frame), we can embed much more information. The disadvantage of this technique is that we have to calculate with high bit errors in the detection process. We can overcome this shortcoming by applying an error-correcting code (we use a (31, 6, 15)-BCH code) and an additional redundancy code on the watermark information before we start the embedding process.

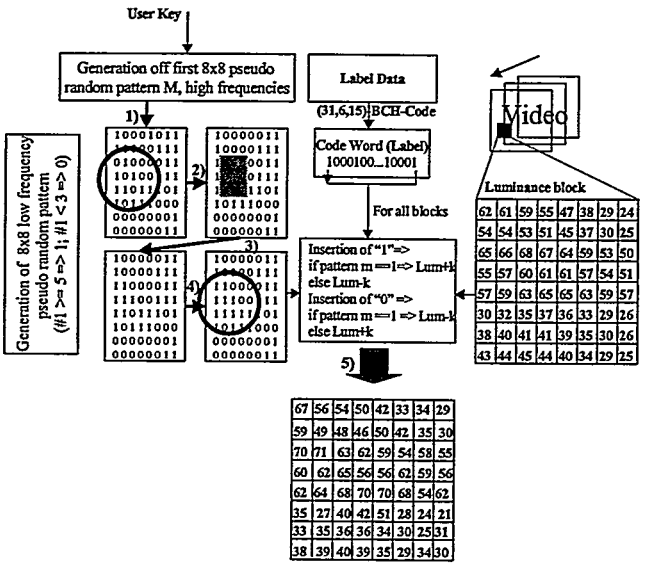

Figure 7: Embedding Process

6.2.2 Retrieval Method

In the retrieval process seen in the next figure the same 8x8 patterns M have to be generated as in the embedding process, step 1 - 4. To test the correlation between the luminance block and the pattern M the average luminance value *av1* (sum1 div #1) of positions with a corresponding '1' and the average luminance value *av0* (sum0 div #0) with a corresponding '0' in the pattern M is produced. If the luminance block and the pattern M would be uncorrelated the difference of both values should be near zero. But due to the embedding process one of these values should be significantly higher (around 2*k) than the other. Thus we estimate an embedded bit '1' if *av1>av0*. Otherwise we estimate an embedded bit '0'. With this statistical analysis we avoid using the original frames. If all bits are retrieved the watermark information is decoded with the same (31,6,15) BCH-Code and the additional redundancy code. The retrieval process is shown in the following diagram:

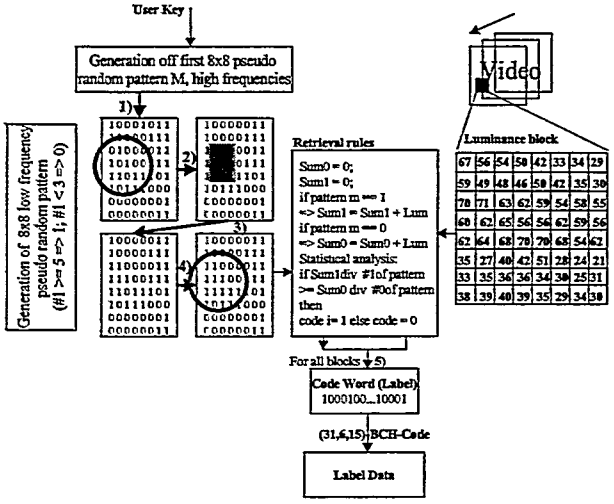

Figure 8: Retrieval Process

6.2.3 Experimental Results

We have tested the improved Fridrich-algorithm with the following parameters: smoothscale = -12, edgescale = 0.27 and offset = 50, watermarking strength 1.1 and bit redundancy 4. Conversion table from Level to k (see description of embedding method):

k	12	12	6	4	3	3
Level/10	0	1	2	3	4	>4

We chose the same video sequences of the museum.mpg. The first watermarked frame can be seen in the following picture. It is hard to evaluate the distortions. A better view can be seen in the 3D-difference view in chapter 6.2.3 or in the HTML version of this paper at www.darmstadt.gmd.de/mobile/media/watermarking. Obviously visual artefacts could be avoided

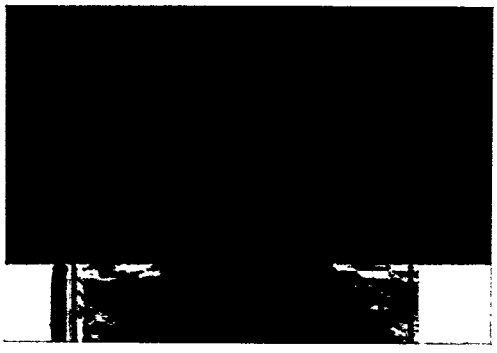

Figure 9: Watermarked museum (first frame)

We embedded 60 Bits of copyrigth information. We performed the same transformations: MPEG-encoding, Quicktime transformation and StirMark-Attack and got following results with the improved Fridrich-algorithm. The table measures the amount of bit errors after the performed transformations with BCH code and the additional redundancy code.

Frame No.	0	1	2	3	4	5	6
F-Type	I	B	B	P	B	B	I
MPEG BCH	0	0	7	6	0	4	0
QuickTime BCH	0	12	16	11	23	15	5
StirMark BCH	19	27	24	23	26	27	24

Frame No.	7	8	9	10	11	12	13
F-Type	B	B	P	B	B	I	B
MPEG BCH	0	2	0	2	8	0	5
QuickTime BCH	15	18	4	26	33	7	18
StirMark BCH	23						

Table 3: Absolute errors

The following error rates can be measured all together of our experiments (about 10 video streams):

museum.mpg:
MPEG:
I-Frames 1-2% P-Frames 3% B-Frames 7%
QuickTime:
I-Frame 8% P-Frames 15% B-Frames 35%
StirMark: 31%

Table 4: Error rates

Regarding the error rate table we want to discuss our results:

If only I- Frames would be watermarked the error rate after MPEG-encoding are also promising. B- and P-frames are not sufficient watermarked. Compared to our adapted Zhao-Koch implementations the algorithm is less successfully with the used error correction. If the watermark should be robust against QuickTime conversion though, the strength of the watermark must be increased by changing the value k or the parameters of the smooth block and edge detection part. As with the Zhao-Koch algorithm the watermark has error rates up to 30 percent after the StirMark attack. But the advantage of the algorithm is, if we embed only 10 information bits with a higher redundancy we get low error rates about 5 percent. Additionally the algorithm is more flexible to handle StirMark attacks more efficiently and robustly.

Now we want to discuss the visual artefacts in detail. To get a more descriptive view on the distortions introduced in the different steps, we measure the differences of the changes to the original frame. The idea is to transform the difference into a 3D scene, [5]. The absolute difference between the original frame and the watermarked frame, the watermarked areas and the intensity of the watermark, measured by the height of the 3D relief can be seen. Based

on these information the quality loss can be measured. It can be seen, if relevant image objects are watermarked and the robustness can be evaluated, regarding the intensity, and different algorithms can be compared. We have created the following 3D-scenes:

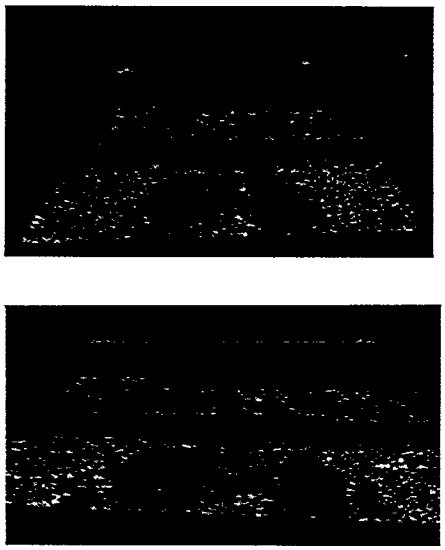

Figure 10: a) Watermarked, Fridrich, b) Watermarked, Zhao-Koch

In the above pictures you see the differences between the original frames and the watermarked versions. The different strengths of the watermark in dependence to the smooth and edge characteristics of the picture can be seen very good. Apparently both algorithms introduce similar changes. This is because of the fact, that both use the same smooth and edge detection algorithm and both introduce changes in the medium frequencies.

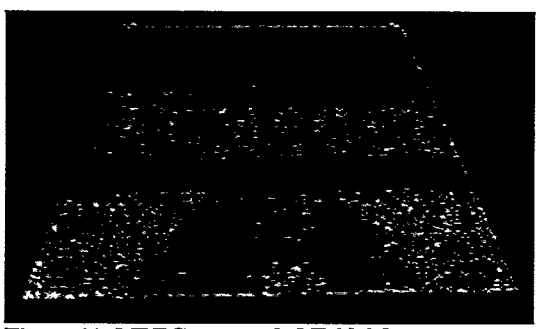

Figure 11: MPEG- reencoded Fridrich

The figure 11 describe the difference of the watermarked-re-encoded frames to the original ones. The difference to the upper two pictures are quite small but leads already to the observed error rates of the Fridrichs approach.

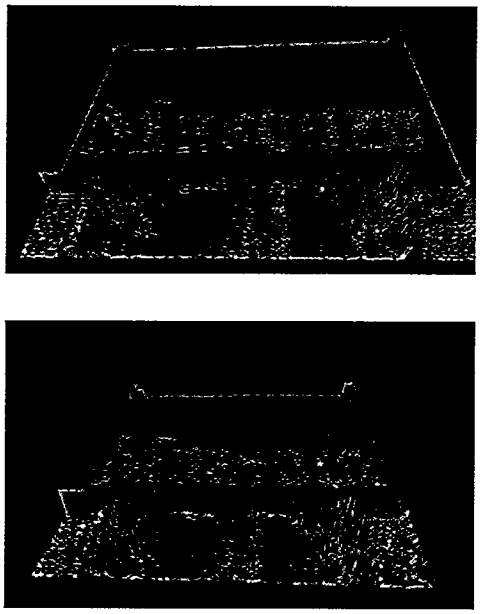

Figure 12: a) StirMark, Fridrich, b) StirMark, Zhao-Koch

The 3D-scenes of figure 12 show the StirMark distortions. Although the distortions seems to be very high they are invisible to the observer when only looking at the "Stir-Marked" frame. This is due to the fact that the biggest distortions are introduced through slight geometric transformations which are difficult to detect without the original frame. Nevertheless they are not invisible to the watermark detection algorithms as can be seen in the appropriate error rates.

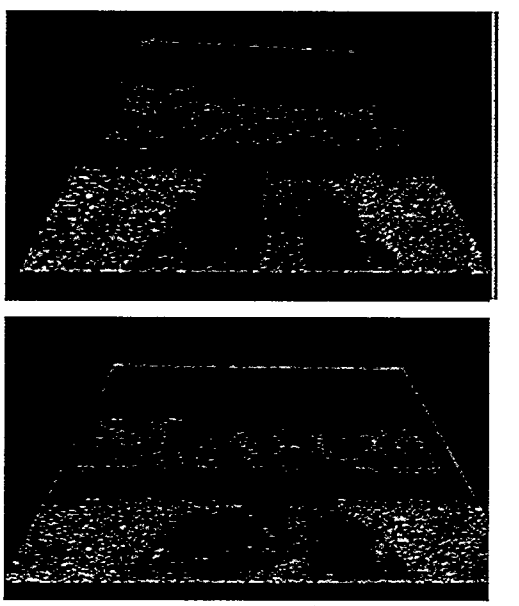

Figure 13: a) QuickTime conversion, Fridrich, b) QuickTime conversion, Zhao-Koch

The last two pictures off figure 13 show the changes to the watermarked frames due to the QuickTime conversion. The distortions of the Zhao-Koch are lower and verify the measured error rates.

6.3 Problems in the Experimental Systems

Is there a way to deal with attacks of the kind StirMark uses? The Fridrich-algorithm offers a nice possibility to handle geometric distortions and clipping. Before checking for correlations the pattern can be geometric transformed in the way the attack is expected. Then the correlation is tested. If the correlation increases the transformation was in the right way and another transformation could be applied. Although this process could be very time consuming it doesn't necessarily need to be applied to each frame, because the attacker would have to apply the same time consuming process. Against clipping the whole pattern consisting of all 8x8 patterns could be shifted over the clipped frame. Again the correlation would be tested and a high value would indicate that the correct part of the pattern matches the clipped frame.

7. Applicability for Object Watermarking

Object Watermarking is one major demand in the MPEG-4 standard to label separate objects of the video or video planes. Both algorithms were analysed if they are useable for inserting and retrieval of labels into regions of video instead of labelling the whole video frames. Our studies are based on traditional MPEG-1 and MPEG-2 videos with an edge detection algorithm based on the Canny algorithm, [5].

The main problem is to retrieve the correct watermarked regions of the whole frame without the knowledge of the position of the objects. Our first approach assumes a minimal region of 64x64 pixel blocks which is watermarked.

The watermarking with the second approach in the spatial domain is simple: a multiple 64x64 pixel pattern which depends on the user key is overlayed across the region which should be watermarked. The retrieval searches for a correlation of the 64x64 patterns in the actual video frame. If the correlation threshold is found an object can be identified. Our experiments have shown that the watermarking strength must be very high to differ from similar primary correlation in other regions of the video frame which were not watermarked. Therefore this practice causes substantial artefacts in the watermarked regions and can be found very easy for an attacker. For tests we have watermarked three regions in the background. The results can be seen in the HTMLversion.

The first approach in the frequency domain provides better results. We embedded an binary sequence of alternating 0 and 1 in 8x8 blocks. The retrieval searches for this alternating sequence in every 64x64 block. The amount of matches of the 0-1 sequence is measured. The visual distortions are less then with the approach II and can be evaluated in the HTML version.

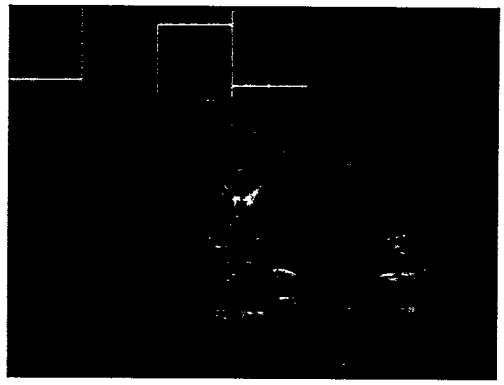

Figure 14: Frame with three watermarked regions approach I

8. Conclusions

In this paper we have discussed MPEG video watermarking techniques, their possibilities and disadvantages to ensure copyright protection. Our robustness tests are mainly based on compression, format conversions and geometrical transformations. We pointed out that our adapted Zhao Koch approach with the error correcting code is appropriate for MPEG video and the visual quality of the watermarked frames could be improved. Format conversions are handled with very low error rates. Especially there are high error rates after StirMark attacks. The attacks are very difficult to handle. The Fridrich approach could also be adapted and improved successfully to embed binary code words and perform a retrieval without the original. The robustness tests are satisfying, but there are still some problems with B- and P-frames after MPEG compression and format conversions. The advantage of the Fridrichs algorithm is to handle StirMark attacks. Therefore our future work is focused on improvements of the Fridrich algorithm to withstand StirMark attacks more efficient. In parallel we ensure multiple watermarking, object watermarking and improve the runtime behaviour by embedding the information into compressed video. The watermarking pattern of the second approach will be first DCT transformed before it is added to the DCT coefficients directly. Afterwards a drift compensation must be performed.

The 3D scenes are very useful to evaluate the visual artefacts and the distortions after several attacks. Our goal is to integrate the watermarking techniques in our distributed video production and distribution environment as an enabling technology for electronic commerce and for digital market places to ensure copyrights.

9. References

[1] Benham D., Memon N., Yeo B.-L., Yeung M.: Fast Watermarking of DCT-based Compressed Images, CISST '97 International Conference

[2] Cox, I.J., Miller, M.L.: A review of watermarking and the importance of perceptual modeling, Proc. Of Electronic Imaging'97, February 1997

[3] Digimarc: Watermarking Technology, PictureMarc™ 1996, http://www.digimarc.com/wt_page.html

[4] Dittmann, J., Steinmetz, A.: Konzeption von Sicherheitsmechanismen für das Projekt DiVidEd, GMD-Studie '97

[5] Dittmann, J., Steinmetz, A., Nack, F., Steinmetz, R.: Interactive Watermarking Environments, to appear in IEEE Multimedia 1998, Austin Texas

[6] Fridrich, J. :Methods for data hidung, Center for Intelligent Systems & Department of Systems Science and Industrial Engineering, SUNY Binghamton, Methods for Data Hiding", working paper (1997)

[7] Hartung, F., Girod, B.: Copyright Protection in Video Delivery Networks by Watermarking of Pre-Compressed Video, in: S. Fdida, M. Morganti (eds.), "Multimedia [7] Applications, Services and Techniques - ECMAST '97", Springer Lecture Notes in Computer Science, Vol. 1242, pp.423-436, Springer, Heidelberg, 1997

[8] Koch, E. and Zhao, J.: Towards Robust and Hidden Image Copyright Labelling, Proc. of 1995 IEEE Workshop on Nonlinear Signal and Image Processing (Neos Marmaras, Greece, Junu 20-22, 1995)

[9] Kuhn, M.G.: Stirmark, available at http://www.cl.cam.ac.uk/mgk25/strirmark/, Security Group, Computer Lab, Cambridge University, UK (email: mkuhn@acm.org)

[10] Kutter, M., Jordan,F. and Bossen,F.: Digital Signature of Colour Images using Amplitude Modulation, Signal Processing Laboratory, EPFL, Switzerland, 1995

[11] MPEG Internationaler Standard ISO/IEC 11172: Information Technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s, Part1: Systems, Part2: Video, Part3: Audio, 1993