JOURNAL OF VISUAL COMMUNICATION AND IMAGE REPRESENTATION
Vol. 9, No. 4, December, pp. 256–270, 1998
ARTICLE NO. VC980397

Real-Time Labeling of MPEG-2 Compressed Video

Gerrit C. Langelaar, Reginald L. Lagendijk, and Jan Biemond

Faculty of Information Technology and Systems (ITS), Information and Communication Theory Group, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands
E-mail: gerhard@it.et.tudelft.nl, lagendijk@it.et.tudelft.nl, biemond@it.et.tudelft.nl

Received December 3, 1997; accepted August 21, 1998

Digital mass recording devices for video data will enter the consumer market soon. Service providers are reluctant to offer services in digital form because of their fears for unrestricted duplication and dissemination. Therefore adequate copy protection systems should be developed, most likely based on labeling techniques. In this paper we propose two different techniques for the real-time labeling of digital video. Both methods embed the label information directly into an MPEG compressed video bitstream. The first method embeds the label by changing variable length codes in the bitstream. The second method discards some of the high frequency DCT-coefficients of the bitstream to embed the label. The first technique is computationally less expensive than the second one, however, the second one is more robust against attacks to remove the label. We show that the label can still be extracted after MPEG re-encoding at a lower bit-rate. © 1998 Academic Press

1. INTRODUCTION

Digital mass recording devices for video data will enter the consumer market soon. The development rate and success of such devices depend not only on technological advances, but also on the existence of adequate copy protection methods. The importance of copy protection systems is stressed by representatives of the consumer electronics and motion picture industries who have agreed to seek legislation concerning digital video recording devices that would protect both intellectual property and consumers' rights [1]. A recommendation is submitted to the US Congress, which allows copyright proprietors to prohibit copying from pay-per-view, video-on-demand programming and pre-recorded media.

To realize a copy protection system, protection signals must be added to the video data in real-time by the service providers and/or the recording devices to indicate if the data may be copied or not. By embedding the protection signals directly in the video data instead of storing them separately, the protection remains intact across different user platforms, interfaces (e.g., VCR/TV-set, multimedia PC) and different data file formats. Another advantage of this approach is that the protection can not be removed without affecting the quality of the video data.

Embedding protection signals into video data can be done by using labeling or watermarking techniques. These techniques embed additional information, the label or watermark, into the video data by modifying the original data under the constraint that the signal's quality is not significantly affected. The label or watermark is a bitstream L, consisting of label bits L_i ($i = 0, 1, 2, \dots, l$), with a much lower bit-rate than the video data bit-rate.

Besides the quality constraint, the label should meet several other requirements. It should be possible to extract the label using only the labeled video data, since a video recording device does not have the original video data at its disposal. The label embedding and extracting procedure should not be too complex, because a copy protection system in a consumer storage device should be inexpensive. Further, the procedures should be performed in real-time. Since video is distributed in MPEG compressed form to save bandwidth and storage space, the label must not increase the size of the compressed bitstream. If the size increases, the buffers in hardware decoders can run out of space or the synchronisation between audio and video can be disturbed. Finally, it should be very difficult to remove the label from the video stream. Processing techniques which do not significantly reduce the quality of the video, like re-encoding the MPEG-stream at a lower bit-rate, can be used as attack to remove the label. The label should be resistant to such processing techniques. A trade-off can be made between robustness and label size. The more label bits are added per video-frame, the less robust the label will be against processing techniques.

In literature, much attention is paid to labeling of still images and video. An overview of recent labeling methods is given in [2]. In [3] a survey is given of possible image-watermarking application scenarios. The labeling methods can roughly be divided into two groups. The methods in the first group only use the labeled data to extract a label, where the methods in the other group use the labeled data

1047-3203/98 $25.00

together with the original unlabeled data. Since for copy protection only the first kind of methods can be used, we focus on this group. Most labeling methods add the label in the spatial [4 . . 7], the Discrete Cosine Transform (DCT) [8 . . 13] or the Discrete Wavelet Transform (DWT) [14, 15] domain. For rotation, scale, and translation invariant labels the Fourier domain is preferable [16].

To perform the labeling in real-time on compressed data, the compressed format must be taken into account, because decoding, labeling and finally re-encoding the data is computationally demanding. Only the methods presented in [8 . . 12] deal with labeling of compressed data. In [8] a technique is proposed which extends the spatial labeling technique described in [6] to the DCT-domain. The label is incorporated in the DCT-coefficient domain of MPEG-2 coded video, without fully decoding and re-encoding the stream. DC and AC coefficients are changed and drift-compensation is performed while the requirement is met that the size of the labeled data should not increase. The label can only be retrieved from the decoded video using correlation techniques in the spatial domain. In [9] a similar, but less complex method is proposed that embeds the label in the DC-coefficients of the DCT-blocks. In [10] another DCT-domain method is proposed in which a label bit is represented by the order of three selected quantized coefficients. These three coefficients are selected out of a randomly chosen 8×8 DCT-block, requantized and adapted in such a way that they have a certain order in size.

The methods described above are not suitable for MPEG video stream copy protection, because they require a full decoding operation, drastically decrease the quality of the video, or increase the size of the labeled data. Therefore we need other techniques which can be applied directly on the MPEG compressed data and which do not have the mentioned drawbacks. In this paper two such techniques are proposed and evaluated [11, 12]. The methods are inspired by the same concept of changing DCT-block's variable length codes (VLCs), but they differ considerably in complexity, robustness and label bit-rate. The first method adds the label directly in the MPEG-2 bitstream by replacing VLCs. Since decoding and re-encoding the video stream is not necessary, the labeling procedure is highly computationally efficient. The second method is computationally slightly more demanding but also more robust. The method is based on partially discarding quantized DCT-coefficients in the compressed MPEG-video stream. For the labeling only partial decoding is needed and no re-encoding is required. We will show that the label can be extracted even after the MPEG-stream has been re-encoded at a lower bit-rate, illustrating the robustness against attacks to remove the label.

The labeling methods proposed in this paper heavily rely on the MPEG video compression standard [17]. In Section 2 the concept of the labeling methods in relation to the relevant parts of the MPEG-standard is discussed. Subsequently the two real-time labeling techniques for compressed video are introduced and evaluated in Sections 3 and 4. Finally, the robustness and complexity of the two methods and their possible usage in a copy protection system are discussed in Section 5.

2. CONCEPT OF LABELING MPEG VIDEO STREAMS

A real-time labeling algorithm for compressed video should closely follow the compression standard to avoid computationally demanding operations, like DCT and IDCT transforms or motion vector calculation. Therefore, the algorithm should work on the lowest layer, called the block-layer, in which spatial 8×8 pixel blocks are represented by 64 quantized DCT-coefficients. Figure 1 shows the different domains in which such a quantized DCT-block can be represented.

The first domain in the block-layer is the *coefficient domain* (cd). In this domain a block contains $N*N$ ($N = 8$) integer entries that correspond with the quantized DCT-coefficients. Many of the entries are usually zero, especially those entries that correspond with the spatial high frequencies. In the *run-level domain*, the non-zero AC coefficients are re-ordered in a zig-zag scan fashion and are subsequently represented by a tuple (r, l), where the run (r) is equal to the number of zeros preceding a certain coefficient and the level (l) is equal to the value of the coefficient. In lowest level domain, the *bit domain* (bd), the (r, l) tuples are represented by variable length coded (VLC) codewords. The codewords for a single DCT-block are terminated by an end of block (EOB) marker.

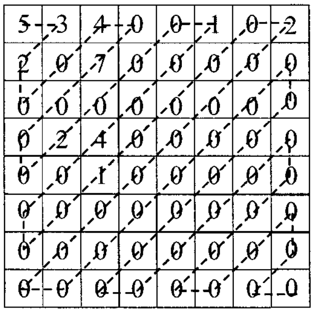
The labeling method proposed in Section 3 operates on the lowest level domain, the *bit domain*, and is based on replacing VLCs. Replacing VLCs in an encoded MPEG stream is not computationally demanding, the label can contain a very large amount of bits and remains intact across different user platforms and interfaces. The method, described in Section 4, is applied on the *coefficient domain* and based on discarding coefficients close to the end of the zig-zag scan. Since VLC decoding and run-level decoding is required, the complexity of this method is higher.

3. LABELING IN THE BIT DOMAIN

3.1. Bit Domain Labeling Concept

The label bitstream L, consisting of bits L_i ($i = 0, 1, 2, \ldots, l$) is embedded in the MPEG-stream by selecting suitable VLCs and forcing the LSB (least significant bit) of their *quantized level* to the value of L_i. To ensure that the change in the VLC yields perceptually invisible degradations after decoding and that the MPEG-bitstream keeps

*N***N* block



coefficient domain (*cd*)

tuples (*r*,*l*)

(0,5), (0,3), (0,2), (2,4), (1,7), (3,2), (3,1), (2,4), (4,1), (4,2)

run-level domain

VLC codewords

001001100
001010
01000
0000000101000
00000010100
001001000
001110
0000000101000
001100
00000011110
10

bit domain (*bd*)

FIG. 1. DCT-block representation domains.

its original size, we select only those VLCs for which another VLC exists with the same run length, a level difference of 1 and the same codeword length. A VLC that meets this requirement is called a label bit-carrying-VLC (*lc-VLC*).

According to Tables B.14 and B.15 of the MPEG-2 standard [17], an abundance of such *lc-VLCs* exists. Furthermore, all fixed-length-coded DCT-coefficients following an Escape-code meet the requirement. Some examples of *lc-VLCs* are listed in Table 1, where the symbol *s* represents the sign-bit. The VLCs in the intra and inter coded macroblocks can be used in the labeling process. The DC coefficients in intra blocks are not used, because they are predicted from other DC coefficients and coded with a different set of VLCs and Escape-codes.

To add the label bitstream L to an MPEG-video bitstream, the VLCs in each macroblock are tested. If an *lc-VLC* is found and the least significant bit of its level is unequal to the label bit L_i ($i = 0, 1, 2, \ldots, l$), this VLC is replaced by another, whose LSB-level represents the label bit. If the LSB of its level equals the label bit L_i the VLC is not changed. The procedure is repeated until all label bits are embedded.

To extract the label bitstream L, the VLCs in each macroblock are tested. If an *lc-VLC* is found, the value represented by its LSB is assigned to the label bit L_i. The procedure is repeated for $i = 0, 1, 2, \ldots, l$ until no *lc-VLCs* can be found anymore.

TABLE 1
Example of *lc*-VLCs in Table B.14 of the MPEG-2 Standard

Variable length code	VLC size	Run	Level	LSB of level
0010 0110 s	8 + 1	0	5	1
0010 0001 s	8 + 1	0	6	0
0000 0001 1101 s	12 + 1	0	8	0
0000 0001 1000 s	12 + 1	0	9	1
0000 0000 1101 0 s	13 + 1	0	12	0
0000 0000 1100 1 s	13 + 1	0	13	1
0000 0000 0111 11 s	14 + 1	0	16	0
0000 0000 0111 10 s	14 + 1	0	17	1
0000 0000 0011 101 s	15 + 1	1	10	0
0000 0000 0011 100 s	15 + 1	1	11	1
0000 0000 0001 0011 s	16 + 1	1	15	1
0000 0000 0001 0010 s	16 + 1	1	16	0

3.2. *Evaluation of the Bit Domain Labeling Algorithm*

The maximum label bit-rate is the maximum number of label bits that can be added to the video stream per second. This label bit-rate is determined by the number of *lc-VLCs* in the video stream and is not known in advance. Therefore, we first experimentally evaluate the maximum label bit-rate by applying the labeling technique to an MPEG-2 video-sequence. The sequence lasts 10 s, has a size of 720 by 576 pixels, is coded with 25 frames per second, has a GOP-length of 12 and contains P, B, and I-frames. The sequence contains smooth areas, textured areas, and sharp edges. During the 10 s of the video there is a gradual frame-to-frame transition and the camera turns fast to another view at the end. This sequence coded at different bit-rates (1.4, 2, 4, 6, and 8 Mbit/s) is used for all experiments in this paper and will be referred to as the "sheep-sequence."

In Table 2 the results of the labeling are listed. Only the intra-coded macro blocks are used for the labeling process. In this table the "number of VLCs" equals the number of all coded DCT-coefficients in the intra coded macroblocks, including the fixed length coded coefficients and the DC-values.

It is possible to store up to 7 kbit of label information per second in the MPEG streams if only intra coded macro blocks are used. If also the inter coded blocks are used the maximum label bit-rate increases to 29 Kbit/s. The results of this experiment are listed in Table 3. In this case

TABLE 2
Total Number of VLCs and Number of *lc*-VLCs in the Intra-coded Macroblocks of 10 Seconds MPEG-2 Video Coded Using Different Bit-Rates and the Maximum Label Bit-Rate

Video bit-rate	Number of VLCs	Number of *lc*-VLCs	Max. label bit-rate
1.4 Mbit/s	334.433	1.152 (0.3%)	0.1 kbit/s
2.0 Mbit/s	670.381	11.809 (1.8%)	1.2 kbit/s
4.0 Mbit/s	1.401.768	34.650 (2.5%)	3.5 kbit/s
6.0 Mbit/s	1.932.917	52.337 (2.7%)	5.2 kbit/s
8.0 Mbit/s	2.389.675	69.925 (2.9%)	7.0 kbit/s

the "number of VLCs" equals the number of all coded DCT-coefficients in the intra and inter coded macroblocks, including the fixed length coded coefficients and the DC-values.

Informal subjective tests show that the labeling does not result in any visible artifacts in the streams coded at 4, 6, and 8 Mbit/s. It was not possible to reliably evaluate the quality degradation due to labeling at less than 2 Mbit/s, because the unlabeled MPEG-streams already contain too many coding artefacts.

Although the visual degradation of the video due to labeling is not noticeable, the degradations are numerically measurable. In particular the maximum local degradations and the drift due to accumulation is of relevance. In Fig. 2a an original I-frame of the MPEG-2 coded "sheep-sequence" is represented. If the original I-frame is subtracted from the corresponding labeled I-frame and the absolute value of the difference signal is amplified by 60, the images, shown in Figs. 2b and 2c, are obtained. Since more bits are stored in an I-frame of a video stream coded at 8 Mbit/s more degradations are introduced (Fig. 2c) than in an I-frame of a video stream coded at 4 Mbit/s (Fig. 2b).

According to Fig. 2 most differences are located around the edges and in the textured areas. The smooth areas are left unaffected. From the experiments it appears that the *lc-VLC*s are fairly uniformly distributed over the DCT-spectrum. Therefore, we can expect each nonzero DCT-coefficient represented by a VLC to have an equal probability of being modified. If we take into account that according to Table 3 at most 3.3% of all VLCs are *lc*-VLCs, the probability of a VLC being modified can roughly be estimated as follows:

$$\begin{aligned} P[\text{VLC modified}] = {} & P[\text{VLC} = lc\text{-VLC}] \\ & \times P[\text{label bit} \neq \text{LSB level of VLC}] \\ & < 0.033 \times 0.5 = 0.016. \end{aligned} \quad (1)$$

Smooth blocks are coded with only one or a few DCT-coefficients. Because only 1.6% of them is replaced, most of the smooth areas are left unaffected. The textured blocks and the blocks containing sharp edges are coded with far more VLCs. These blocks will therefore contain the greater part of the *lc*-VLCs.

The maximum local degradation or the number of *lc*-VLCs per block must be as low as possible. The visual impact of the labeling will be much smaller if the degradations introduced by modifying an *lc*-VLC are distributed more or less uniformly over the frame, instead of being concentrated and accumulated in a relative small area of the frame or even worse being accumulated in a single DCT-block. In Fig. 3 the numbers of *lc*-VLCs per 8×8 block for the "sheep-sequence" coded at 8 Mbit/s are plotted in a histogram. It shows that 87% of all coded 8×8 blocks do not contain any *lc*-VLC. The rest of the coded 8×8 blocks contain one or more *lc*-VLCs. Most blocks (186.662) contain only one *lc*-VLC, which is about 64% of all *lc*-VLCs in the sequence.

These numbers can be explained by examining Table B.14 and B.15 of the MPEG-2 standard [17]. The most frequently occurring run-level pairs are coded with short VLCs. Almost all short VLCs do not qualify as an *lc*-VLC. This means that the chance on a large number of *lc*-VLCs in one 8×8 block is relatively low.

To limit the maximum number of *lc*-VLC replacements per DCT-block to T_m, a threshold mechanism can be used. If the number of *lc*-VLCs exceeds T_m, only the last T_m *lc*-VLCs are used for the labeling, the rest is left unchanged. In Table 4 the label bit-rates for the "sheep-sequence" coded at 8 Mbit/s are listed for several values of T_m. If at most 2 *lc*-VLC replacements per block are allowed ($T_m = 2$), the label bit-rate is only decreased to 83% of the maximum label bit-rate for which T_m = unlimited. So limiting the number of *lc*-VLC replacements per block can avoid unexpected large local degradations without drastically affecting the maximum label bit-rate.

In an MPEG-video stream P frames are predicted from the previous I or P frame. The B frames are predicted from the two nearest I or P frames. Since intra and inter coded macroblocks are used for the labeling, errors are

TABLE 3
Total Number of VLCs and Number of *lc*-VLCs in the Intra- and Inter-Coded Macroblocks of 10 Seconds MPEG-2 Video Coded Using Different Bit-Rates and the Maximum Label Bit-Rate

Video bit-rate	Number of VLCs	Number of *lc*-VLCs	Max. label bit-rate
1.4 Mbit/s	350.656	1.685 (0.5%)	0.2 kbit/s
2.0 Mbit/s	1.185.866	30.610 (2.6%)	3.1 kbit/s
4.0 Mbit/s	4.057.786	135.005 (3.3%)	13.5 kbit/s
6.0 Mbit/s	7.131.539	222.647 (3.1%)	22.3 kbit/s
8.0 Mbit/s	10.471.557	289.891 (2.8%)	29.0 kbit/s

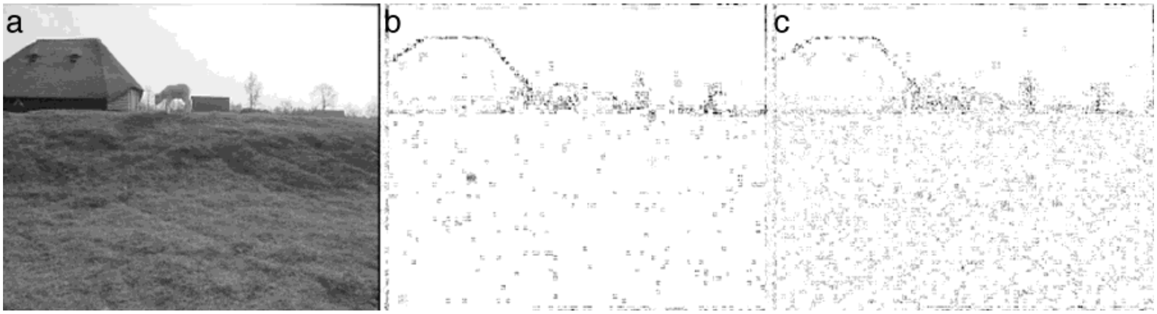


FIG. 2. (a) Unlabeled I-frame. (b) Frame difference (4 Mbit/s) label bit-rate 13.5 kbit/s. (c) Frame difference (8 Mbit/s) label bit-rate 29.0 kbit/s.

introduced in all frames. However, error accumulation (drift) from the frames used for the prediction occurs in the predicted P and B frames. The drift can clearly be seen in Fig. 4, where the difference $\Delta MSE = MSE_l\text{-}MSE_u$ is plotted. The MSE_u is the MSE per frame between the uncompressed "sheep-sequence" and the sequence coded at 8 Mbit/s. The MSE_l is the MSE per frame between the uncompressed sequence and the labeled sequence coded at 8 Mbit/s.

In Fig. 4 it can be seen that the I-frames (numbered 1, 13, 25, 37, . .) have the lowest ΔMSE, the ΔMSE of a predicted B frame is 2 to 3 times larger than the error in the I-frames in the worst case. The average peak-signal-to-noise-ratio ($PSNR$) between the MPEG-compressed original and the uncompressed original is 37 dB. If the labeled compressed video stream at 8 Mbit/s is compared with the original compressed stream, the ΔMSE causes an average $\Delta PSNR$ of 0.1 dB and a maximum $\Delta PSNR$ of 0.2 dB.

3.3. *Robustness*

A large label bit-stream can be added and extracted in a very fast and simple way, but it can also be removed without significantly affecting the quality of the video. However, it still takes a lot of effort to completely remove a label from a large MPEG video stream. For example,

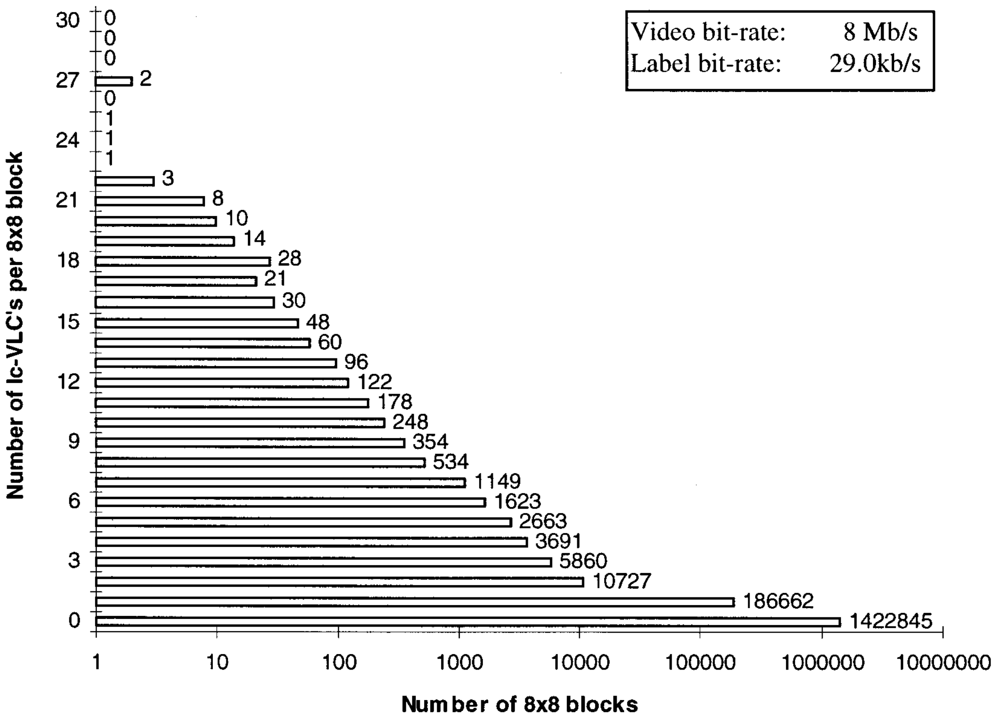


FIG. 3. Number of *lc*-VLCs per 8 × 8 block.

TABLE 4
Label Bit-Rates Using a Threshold for at Most T_m *lc*-VLC Replacements per 8 × 8 DCT-Block (Video Bit-Rate 8 Mbit/s)

T_m = max. *lc*-VLC replacements per block	Max. label bit-rate
2	24.2 Kbit/s
4	26.9 Kbit/s
6	28.1 Kbit/s
8	28.6 Kbit/s
10	28.8 Kbit/s
Unlimited	29.0 Kbit/s

decoding the labeled MPEG-stream and encoding it again using another bit-rate will destroy the label. But re-encoding is a computationally and memory (disk) demanding operation. The easiest way to remove the label is by labeling the stream again using another random label bitstream. In this case the quality is slightly affected. During the re-labeling phase the adapted *lc*-VLCs in the labeled video stream can either return to their original values or change to VLCs that represent DCTs that differ two quantisation levels from the original ones in the unlabeled video stream. Nonadapted *lc*-VLCs in the labeled video stream can change to a value that differs one quantisation level from the one in the original video stream. This means that there is extra distortion, although the quality is only slightly affected. Since relabeling of a large MPEG video stream still requires special hardware or a very powerful computer, the bit domain labeling method is still suitable for consumer applications.

4. LABELING IN THE COEFFICIENT DOMAIN

4.1. Coefficient Domain Labeling Concept

The coefficient domain labeling method embeds the label bitstream L consisting of bits L_i $(i = 0, 1, 2, \ldots, l)$ in the I-frames only. Each bit out of the label string has its own label bit-carrying-region, *lc-region*, in an I-frame. For instance, the first bit is located in the top-left-corner of the I-frame, in an *lc-region* of n 8 × 8 pixel blocks, as illustrated

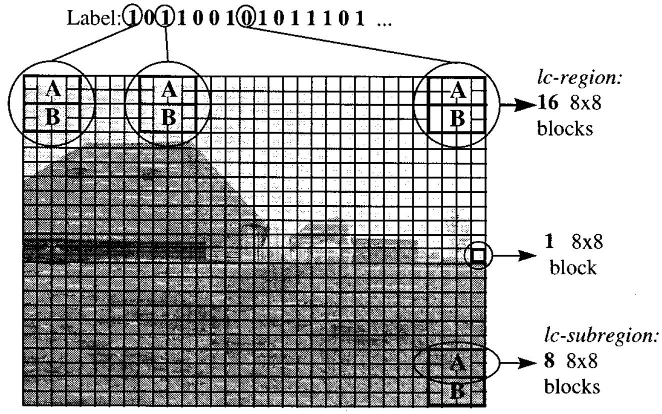


FIG. 5. Bit positions and block definitions in a frame.

in Fig. 5 for $n = 16$. The size of this *lc-region* determines the label bit-rate. The higher n, the lower the label bit-rate.

A label bit is embedded in an *lc*-region by introducing an "energy" difference between the high frequency DCT-coefficients of the top half of the *lc*-region (denoted by *lc*-subregion A) and the bottom half (denoted by B). The energy E_A equals the squared sum of a particular subset of DCT-coefficients in *lc*-subregion A. This subset is denoted by $S(c)$ and is illustrated in Fig. 6 by the white triangularly shaped areas in the DCT-blocks. We define the total energy in $S(c)$, computed over the $n/2$ blocks in subregion A, as

$$E_A(c) = \sum_{b=0}^{n/2-1} \sum_{u \in S(c)} (DCT\, coeff_{(u,b)})^2. \tag{2}$$

In (2) the numbering of the DCT-coefficients is via the subscripts (u, b), where u represents the DCT-coefficient number after the zig-zag scanning and b represents the DCT-block number in *lc*-subregion A (see Fig. 6). Similarly the energy in subregion B, denoted by E_B, can be defined. $S(c)$ is typically defined according to a cutoff point c in the zig-zag scanned DCT-coefficients,

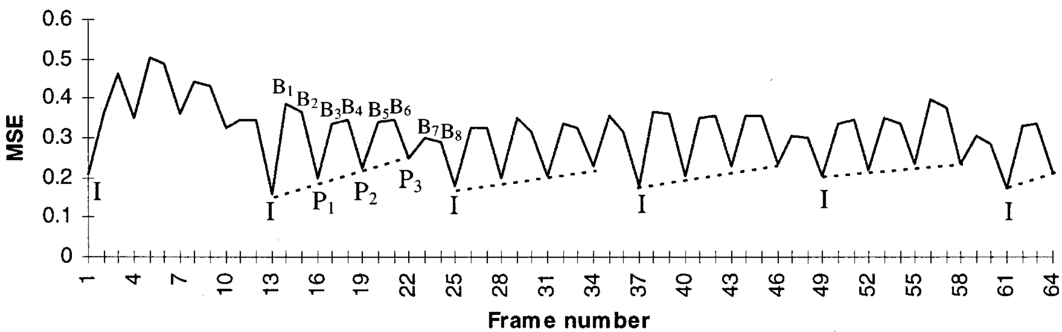


FIG. 4. ΔMSE of labeled sheep-sequence at 8 Mbit/s, label bit-rate 29.0 kbit/s.

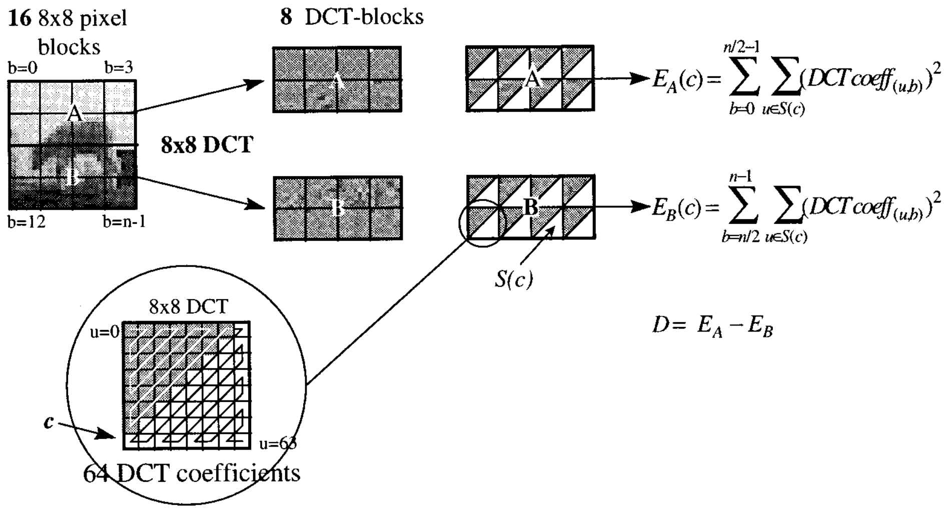


FIG. 6. Extracting a label bit from an $n \times n$ pixel area.

$$S(c) = \{u \in \{0, 63\} \mid (u > c)\}. \tag{3}$$

The selection of a suitable cutoff point for an *lc*-region is very important for the robustness and the visibility of the label bit and will be discussed later. First we focus on how the labeling procedure works, assuming that we have available a suitable cutoff point c. The energy difference D between top and bottom half of an *lc*-region is defined as

$$D = E_A - E_B. \tag{4}$$

In Fig. 6 the complete procedure to calculate the energy difference D of a *lc*-region ($n = 16$) is graphically illustrated.

We now define the label bit value as the sign of the energy difference D. Label bit "0" is defined as $D > 0$ and label bit "1" as $D < 0$. The label embedding procedure must therefore adapt E_A and E_B to manipulate the energy difference D. If label bit "0" must be embedded, all energy after the cutoff point in the DCT-blocks of *lc*-subregion B is eliminated by setting the corresponding DCT-coefficients to zero, so that

$$D = E_A - E_B = E_A - 0 = +\boldsymbol{E_A}. \tag{5}$$

If label bit "1" must be embedded, all energy after the cutoff point in the DCT-blocks of *lc*-subregion A is eliminated, so that

$$D = E_A - E_B = 0 - E_B = -\boldsymbol{E_B}. \tag{6}$$

There are several reasons for using this energy difference in the *triangularly shaped areas.* The most important reason is that calculating the energy difference and changing E_A and E_B can easily be done on the compressed stream. All DCT-coefficients, needed for the calculation of E_A or E_B are conveniently located at the end of the compressed 8×8 DCT-block after zig-zag ordering. The coefficients can be forced to zero to adapt the energy without re-encoding the stream by shifting the end of block marker (EOB) towards the DC-coefficient. Because coefficients are removed to add a label, the labeled compressed video stream will always be smaller than the unlabeled video stream. If it is necessary that the labeled compressed video stream keeps its original size, stuffing bits can be inserted before each macroblock. Figure 7 graphically illustrates the procedure to calculate E in a single compressed DCT-block and to change E by removing DCT-coefficients located at the end of the zig-zag scan (i.e., high frequency DCT-coefficients).

4.2. *Detailed Algorithm Description*

Equations (3) and (4) indicate that the energy present in both *lc*-subregions A and B is of much importance, because during the label extraction the difference between the energies will be used to determine the label bit. The values of E_A and E_B are determined by two factors:

- the size of the *triangular shaped areas* (i.e., the cutoff point c)
- the spatial contents of the subregions A and B

If a positive energy difference $D = E_A - E_B$ must be generated in the *lc*-region presented in Fig. 8a, all DCT-coefficients in *lc*-subregion B must be eliminated by selecting an extremely low cutoff point to make $E_A > E_B$. Figure 8b illustrates this situation, where the presence of the label bit becomes clearly visible in *lc*-subregion B. In order to

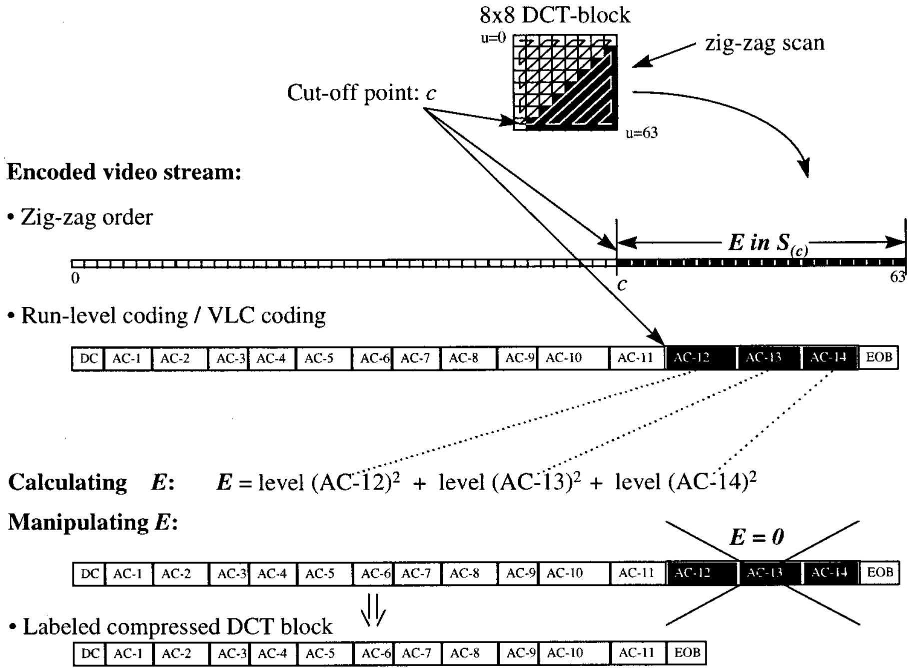


FIG. 7. Calculating and adapting the energy in an 8×8 compressed DCT-block.

avoid the use of an extremely low cutoff point in case one of the subregions is smooth, we propose to randomly shuffle all DCT-blocks in the I-frame prior to embedding the label bits (Fig. 8c). This does not pose any problems in practice when using MPEG streams, because effectively we now select randomly DCT-blocks from the compressed stream to define an *lc*-region, instead of spatially neighbouring blocks. The crucial point for the quality/robustness trade-off is now the choice for a cutoff point c in each *lc*-region.

To determine the cutoff point c for an *lc*-region, we first calculate the energies $E_A(c_{tmp})$ and $E_B(c_{tmp})$ for all possible cutoff points $c_{tmp} = 0 \cdot\cdot 63$. If D is the energy difference which will be enforced to add a label bit in an *lc*-region, the cutoff point c is given by

$$c = \mathrm{MAX}(c_T), \quad \text{where } c_T = \{c_{tmp} \in \{0, 63\} \\ |(E_A(c_{tmp}) > D) \wedge (E_B(c_{tmp}) > D)\} \tag{7}$$

The parameter D influences the number of DCT-coefficients which are discarded during labeling and determines the visibility and robustness of the label together with parameter n, the number of 8×8 blocks per label bit. This energy difference D will decrease if the MPEG stream is modified, e.g., by re-encoding or filtering. If a relative large value for n is chosen, each label bit will spread over a large area of the frame. The energies E_A and E_B will now increase and fewer DCT-coefficients per 8×8 block need to be discarded to create a difference D.

To extract a label bit from an *lc*-region we have to find back the cutoff point c, which was selected during the

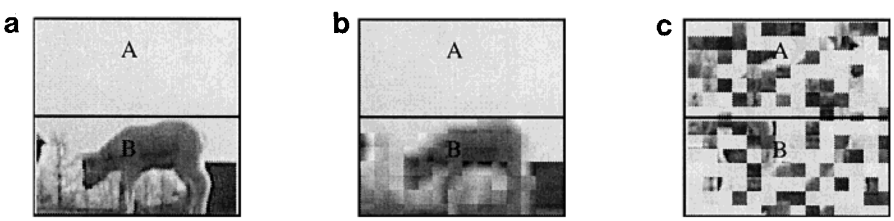


FIG. 8. (a) Smooth and textured area; (b) positive energy difference enforced $E_A > E_B$; (c) shuffled frame.

Labeling procedure:

Shuffle all 8x8 DCT-Y-blocks of an I-frame pseudo-randomly

FOR all label bits L_i in label string L DO

Select *lc-subregion A* consisting of *n/2* 8x8 DCT-blocks,
Select *lc-subregion B* consisting of *n/2* other blocks (Fig. 5)

FOR c_{tmp} = 1 to 63 DO

Calculate $E_A(c_{tmp})$ and $E_B(c_{tmp})$: $E_A(c) = \sum_{b=0}^{n/2-1} \sum_{u \in S(c)} (DCT\,coeff_{(u,b)})^2$

$c = MAX(c_T)$, *where* $c_T \in \{ c_{tmp} \in \{0,63\} \mid (E_A(c_{tmp}) > D) \wedge (E_B(c_{tmp}) > D)\}$

IF (L_i = 0) THEN discard coefficients in $S_B(c)$ (all coefficients after c in all 8x8 blocks of area B)

IF (L_i = 1) THEN discard coefficients in $S_A(c)$ (all coefficients after c in all 8x8 blocks of area A)

Shuffle all 8x8 DCT-Y-blocks back to their original locations

Label extracting procedure:

Shuffle all 8x8 DCT-Y-blocks of an I-frame pseudo-randomly

FOR all label bits L_i in label string L DO

Select *lc-subregion A* consisting of *n/2* 8x8 DCT-blocks,
Select *lc-subregion B* consisting of *n/2* other blocks (Fig. 5)

FOR c_{tmp} = 1 to 63 DO

Calculate $E_A(c_{tmp})$ and $E_B(c_{tmp})$: $E_A(c) = \sum_{b=0}^{n/2-1} \sum_{u \in S(c)} (DCT\,coeff_{(u,b)})^2$

$c_A = MIN(c_T)$, *where* $c_T \in \{ c_{tmp} \in \{0,63\} \mid (E_A(c_{tmp}) < T)\}$

$c_B = MIN(c_T)$, *where* $c_T \in \{ c_{tmp} \in \{0,63\} \mid (E_B(c_{tmp}) < T)\}$

$L_i = 0$

IF $(c_A < c_B)$ THEN L_i = 1

IF $(c_A = c_B) \wedge (E_A(c_A) < E_B(c_A))$ THEN L_i = 1

FIG. 9. Complete procedure for label embedding and extracting.

embedding process. First the energies $E_A(c_{tmp})$ and $E_B(c_{tmp})$ are calculated for all possible cutoff points $c_{tmp} = 0\cdot\cdot 63$. Since only in one *lc*-subregion A or B DCT-coefficients are eliminated, we search for two cutoff points c_A in *lc*-subregion A and c_B in *lc*-subregion B:

$$c_A = \mathrm{MIN}(c_T), \quad \text{where } c_T = \{c_{tmp} \in \{0,63\} \mid (E_A(c_{tmp}) < T)\} \tag{8}$$

$$c_B = \mathrm{MIN}(c_T), \quad \text{where } c_T = \{c_{tmp} \in \{0,63\} \mid (E_B(c_{tmp}) < T)\}. \tag{9}$$

Then we conclude that the actually used cutoff point was

$$c = \mathrm{MIN}(c_A, c_B). \tag{10}$$

The detection threshold T influences the determination of the cutoff point. This value must be smaller than the enforced energy difference D, but larger than 0. If $T = 0$ the label can correctly be extracted only if the video-stream is not affected by processing like adding noise, filtering or re-encoding. However, if a small amount of noise is introduced in the highest DCT-coefficients, cutoff points will be detected, which are higher than the originally enforced ones. T determines which amount of energy will be seen as noise. The complete embedding and extracting procedures are shown in Fig. 9.

4.3. Evaluation of the Coefficient Domain Labeling Algorithm

The label bit-rate is fixed and determined by n, the number of 8×8 DCT-blocks per *lc*-region. To evaluate the visual impact of the labeling and the label bit-rate we

TABLE 5
Number of 8 × 8 DCT-Blocks per Bit, Number of Bits Discarded by the Labeling Process, Percentage Label Bit Errors and Label Bit-Rate for the "Sheep-Sequence" Coded at Different Bit-Rates

Video bit-rate	n	Discarded bits	Percentage bit errors	Label bit-rate
1.4 Mbit/s	64	1.6 Kbit/s	24.6	0.21 kbit/s
2.0 Mbit/s	64	4.6 Kbit/s	0.1	0.21 kbit/s
4.0 Mbit/s	64	3.8 Kbit/s	0.0	0.21 kbit/s
6.0 Mbit/s	32	7.2 Kbit/s	0.0	0.42 kbit/s
8.0 Mbit/s	32	6.6 Kbit/s	0.0	0.42 kbit/s

applied the labeling technique to the "sheep-sequence" coded at different bit-rates. Over a wide range of sequences we have found a reasonable setting for the energy difference $D = 20$ and the detection threshold $T = 15$. The cutoff points c for each label bit are allowed to vary in the range from 6 to 63. Informal subjective tests show that the label, embedded with $n = 32$, is not noticeable in video streams coded at 8 and 6 Mbit/s. If MPEG streams coded at a lower bit-rate are labeled with $n = 32$, blocking artefacts around edges of smooth objects appear. By increasing n further to 64 the artefacts disappear in the MPEG stream coded at 4 Mbit/s. At a rate of 1.4 and 2 Mbit/s the compression artefacts always dominate the additional degradations due to labeling. In Table 5 the results of the experiments are listed. The third column shows the number of bits, which are discarded by the labeling process. The fourth column presents the percentage bit errors found by extracting the label L' from the labeled stream and comparing L' with the originally embedded one, L. It appears that not enough high frequency coefficients exist in the compressed stream coded at 1.4 Mbit/s to create the energy differences D for the label bits, since only 75% of the extracted label bits are correct.

In Fig. 10a the original I-frame of the MPEG-2 coded sheep-sequence is represented. If the original I-frame is subtracted from the corresponding labeled I-frame and the absolute value of the difference signal is amplified by 60, the images, which are shown in Figs. 10b and 10c, are obtained.

If we compare Fig. 10 with Fig. 2, we see that this labeling method causes fewer differences per frame than the method described in Section 3, although the differences per block are larger. Using the bit domain labeling method a DCT-coefficient is only altered by one quantisation level, here complete DCT-coefficients are discarded. All degradations are located in DCT-blocks with a relatively large number of high frequency DCT-components, textured blocks, and blocks with edges. Only blocks with energy in DCT-coefficients above the local cutoff point are affected. The histograms of the cutoff points in the "sheep-sequence coded at 1.4 and 8 Mbit/s" are plotted in Fig. 11. The minimum cutoff point for the stream coded at 8 Mbit/s is 33, for a stream coded at 1.4 Mbit/s the minimum is 6. The lower the bit-rate is, the lower the cutoff points have to be because of the lack of high energy components in the compressed video stream.

The visual impact of the labeling will be much smaller if the degradations introduced by discarding DCT-coefficients are distributed more or less uniformly over the frame. Removing all VLCs representing DCT-coefficients from a few textured blocks will cause highly visible artefacts. In Fig. 12 the numbers of discarded VLCs per 8 × 8 block for the "sheep-sequence" coded at 8 Mbit/s are plotted in a histogram. It appears that 95% of all coded 8 × 8 blocks in the I-frames are not affected by the algorithm. From an *lc*-region only the DCT-coefficients above a certain cutoff point in the half, an *lc*-subregion, are eliminated. This means that from an *lc*-subregion only a few (average 10%) 8 × 8 blocks have energy above the cutoff point.

Like in the bit domain labeling algorithm a limit T_m can be set on the number of VLCs per 8 × 8 block that are discarded during the labeling process. Whereas in the bit domain labeling algorithm this limit decreases the label

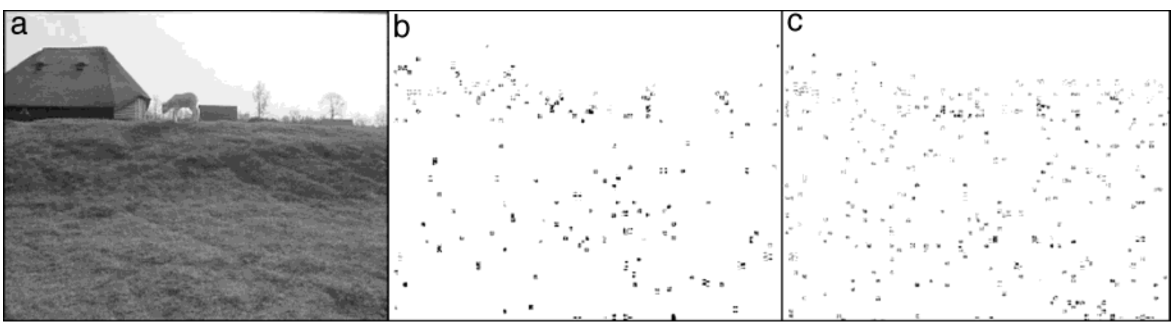


FIG. 10. (a) Unlabeled I-frame; (b) frame difference (4 Mb/s); label bit-rate 0.21 kbit/s ($n = 64$); (c) frame difference (8 Mb/s); label bit-rate 0.42 kbit/s ($n = 32$).

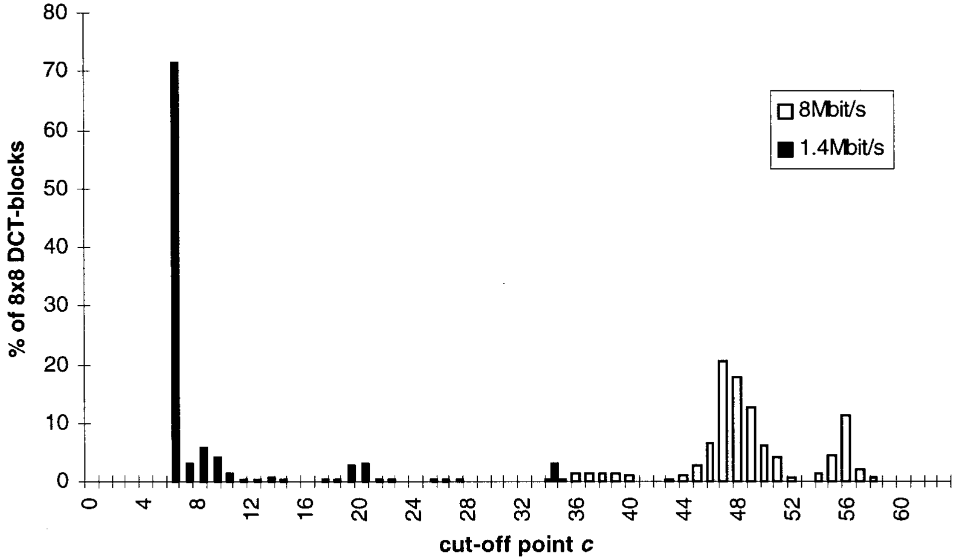


FIG. 11. Histograms of the cutoff points in an MPEG-2 sequence coded at 1.4 and 8 Mb/s, label bit-rates are respectively 0.21 kbit/s and 0.42 kbit/s.

bit-rate, the coefficient domain labeling has a fixed label bit-rate. Instead, setting a limit T_m affects the robustness of the label. If some DCT-coefficients in one 8×8 block of an *lc*-subregion are not eliminated, because the limit T_m prohibits it, in the worst case one label bit error can occur if the label extracted from this stream is compared with the originally embedded one. However, since each label bit is dependent on n 8×8 blocks, the likelihood that this error occurs is relatively small. In Table 6 the worst case percentage bit errors, which are introduced in the label of the "sheep-sequence" coded at 8 Mbit/s, are listed for several values of T_m. With proper error correcting codes on the label stream, the number of VLCs to be removed can be greatly limited at the advantage of a better visual quality without significantly effecting the label retrieval.

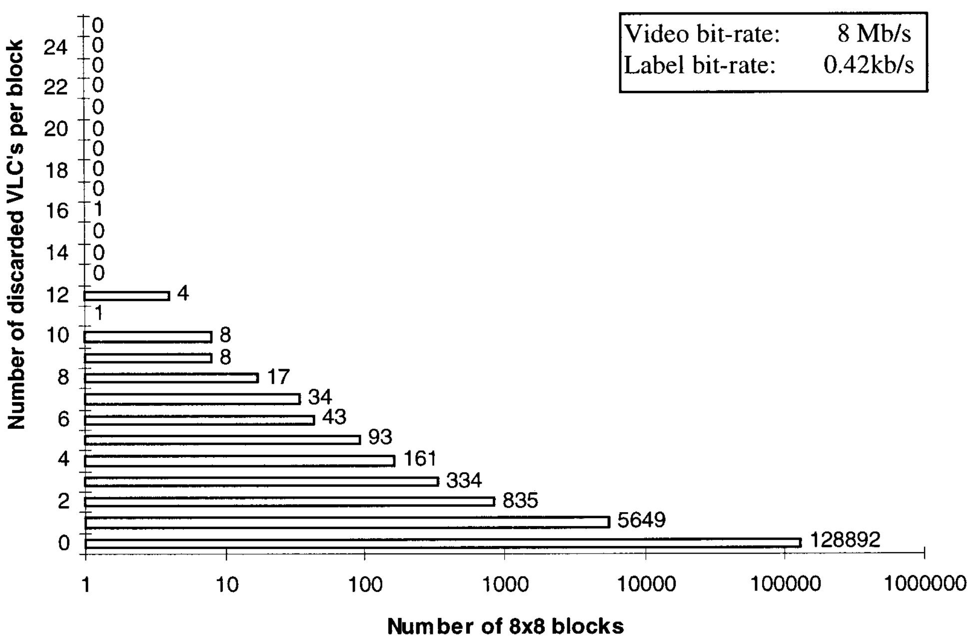


FIG. 12. Number of discarded VLCs per 8×8 block.

TABLE 6
Worst Case Percentage Label Bit Errors Introduced by Limit T_m, the Maximum Number of Discarded VLCs per 8 × 8 Block (Video Bit-Rate 8 Mbit/s, Label Bit-Rate 0.42 kbit/s)

T_m = Max. number of discarded VLCs per block	Worst case percentage bit errors
2	17%
3	9%
4	5%
5	3%
6	2%
Unlimited	0%

Since P and B frames are predicted from I and P frames, the degradations introduced in the I-frames by labeling appear also in the predicted frames. Because the P and B frames are only partially predicted from other frames and partially intra coded, the degradations will fade out. No degradations are introduced in the intra coded parts of the predicted frames by the labeling. The error fade-out can clearly be seen in Fig. 13, where the difference MSE_l-MSE_u is plotted. The MSE_u is the MSE per frame between the uncompressed "sheep-sequence" and the sequence coded at 8 Mbit/s. The MSE_l is the MSE per frame between the uncompressed sequence and the labeled sequence coded at 8 Mbit/s.

The average *PSNR* between the MPEG-compressed original and the uncompressed original is 37 dB. If the labeled compressed video stream at 8 Mbit/s is compared with the original compressed stream, the ΔMSE causes an average $\Delta PSNR$ of 0.06 dB and a maximum $\Delta PSNR$ of 0.3 dB. It appears that this method has less impact on the average $\Delta PSNR$ and more impact on the maximum $\Delta PSNR$ than the method described in Section 3.

4.4. *Robustness*

Like the method described in Section 3, the label embedding and extracting procedures can be applied to a video stream in real-time; however, the method is slightly more complex. Removing the label by labeling the stream again using another label and other pseudo-random block shuffling is not possible. Other more time-consuming computationally and memory (disk) demanding methods have to be applied to the labeled compressed video stream to attempt to remove the label. For simple filtering techniques the compressed stream must be decoded and completely re-encoded. A less complex and disk demanding, but still very computationally demanding operation would be transcoding. To see if the label is resistant to transcoding or re-coding at a lower bit-rate, the following experiment is performed. The "sheep-sequence" is MPEG-2 encoded at 8 Mbit/s and this compressed stream is labeled (n = 32). Hereafter, the labeled video sequence is transcoded at different lower bit-rates.

The label stream is extracted from the transcoded streams and each label bit is compared with the originally embedded label bitstream. If 50% bit errors are made the label is completely removed. The bit errors introduced by decreasing the bit-rate are represented in Fig. 14. It appears that if the video bit-rate is decreased by 25%, only 7% label bit errors are introduced. Even if the video bit-rate is decreased by 38%, 79% of the label bitstream can be extracted correctly. Error correcting codes can improve this result further.

5. DISCUSSION

If the two labeling methods proposed in this paper are compared with each other, a trade-off can be seen between complexity, label bit-rate, and robustness. The bit domain labeling method is computationally highly efficient and

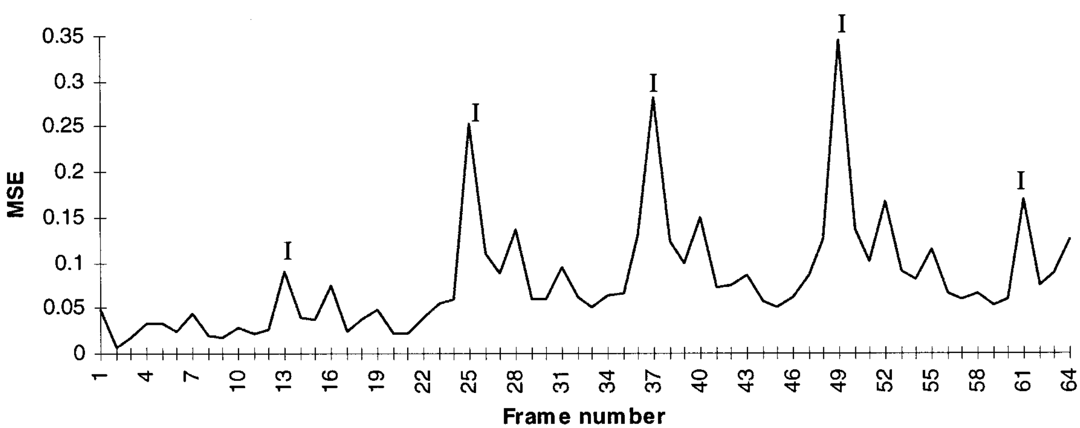


FIG. 13. ΔMSE of labeled "sheep-sequence" at 8 Mbit/s, label bit-rate 0.42 kbit/s.

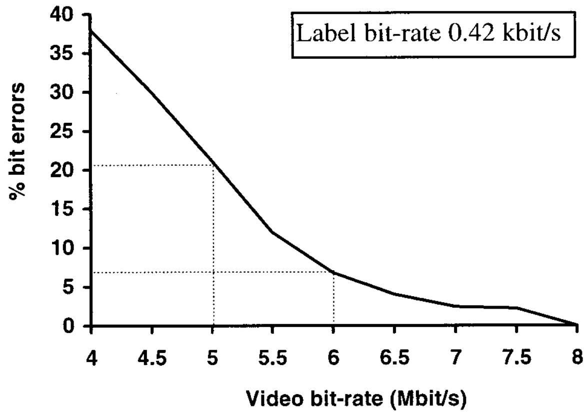


FIG. 14. Percentage of bit errors after transcoding a labeled 8 Mbit/s Mpeg-2 sequence at a lower bit-rate.

has a very high label bit-rate, but is not robust against transcoding or re-labeling. The coefficient domain labeling method is more complex and has a lower label bit-rate, but is resistant to a wide range of attacks like transcoding and re-labeling.

The two labeling methods are both suitable for a real-time copy protection system for digital video recording devices. A copy protection scheme using our labeling methods is shown in Fig. 15. This system must be implemented in a video recording device and adds a "copy-prohibit" label to the video stream which is stored. Before the recording devise starts recording, the incoming video stream is checked for this label. The recording device will only start if the "copy-prohibit" label is not detected. This means that data can only be copied once.

The "copy-prohibit" label must be added at regular intervals in the video stream to enable the recorder to detect the label at any arbitrary position in the video stream. In a high quality video stream the label bit-rate can be higher than the required label bit-rate. In this case the extra bandwidth can be used to improve the robustness of the label by applying error correcting codes or, if the second method is used, by using larger *lc*-regions.

The question which labeling method should be used for a digital video recording device is currently under investigation within the European ACTS SMASH project (AC018: Storage for Multimedia Applications Systems in the Home) [18]. The strength of the copy protection system is dependent on the robustness of the labeling algorithm. If it appears that the strength of the SCMS [19] copy protection system currently in use for digital audio recording devices is also sufficient for digital video recording devices, the bit domain labeling method can be used and the complexity of the required hardware can be kept low. Although a label embedded with this method can be removed by relabeling, a copy protection system using this method will still be stronger than the SCMS system. If a stronger copy protection system is required, the more complex coefficient domain labeling method can be used.

APPENDIX: LIST OF SYMBOLS

A, B	A, top half of an *lc*-region; B, bottom half of an *lc*-region
b	DCT block number $[0 \cdot\cdot n-1]$ in an *lc*-region
c	cutoff point $[0 \cdot\cdot 63]$ in an *lc*-region
c_A, c_B	cutoff point $[0 \cdot\cdot 63]$ in *lc*-region A or B
c_{tmp}	temporary cutoff point
D	energy difference between *lc*-subregion A and *lc*-subregion B
E_A, E_B	energy in *lc*-subregion A or *lc*-subregion B
EOB	end of block marker
L	label bit string
L_i	ith bit out of the label bitstring
l	length of label bit string
lc-VLC	label bit carrying variable length code
lc-region	label bit carrying region in an I-frame
lc-subregion	half of label bit carrying region in an I-frame
LSB	least significant bit
N	dimension of a DCT block ($N = 8$)
n	number of 8×8 DCT blocks in an *lc*-region
$S(c)$	Subset of DCT coefficient numbers dependent on c $S(c) \in \{0 \cdot\cdot 63\}$
T	detection threshold
T_m	threshold for maximum number of changes per 8×8 DCT block
u	DCT-coefficient number after zig-zag scanning $[0 \cdot\cdot 63]$
VLC	variable length code.

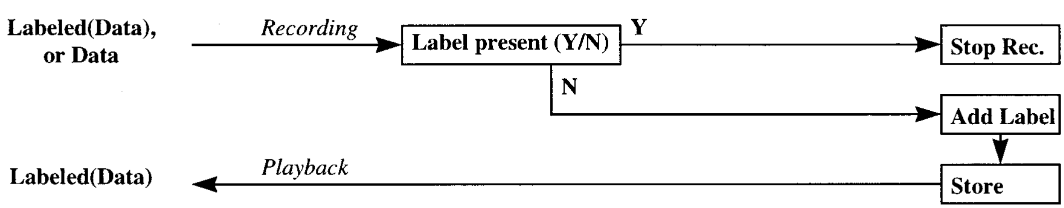


FIG. 15. Simple copy protection scheme allowing only one copy.

REFERENCES

1. J.-L. Renaud, PC industry could delay DVD, *Advanced Television Markets*, Issue 47, May 1996.
2. I. J. Cox and M. L. Miller, A review of watermarking and the importance of perceptual modeling, in *Proceedings of SPIE Electronic Imaging '97, San Jose, California, February 1997.*
3. F. Mintzer, G. W. Braudaway, and M. M. Yeung, Effective and ineffective digital watermarks, in *Proceedings IEEE International Conference on Image Processing (ICIP '97), Santa Barbara, California, October 1997.*
4. W. Bender, D. Gruhl, N. Morimoto, Techniques for data hiding, in *Proceedings of the SPIE, San Jose, California, February 1995,* 2420:40.
5. I. Pitas and T. Kaskalis, Signature casting on digital images, in *Proceedings IEEE Workshop on Non-linear Signal and Image Processing, Neos Marmaras, Greece, June 1995.*
6. J. R. Smith, B. O. Comiskey, Modulation and information hiding in images, in *Preproceedings of Information Hiding, an Isaac Newton Institute Workshop, University of Cambridge, May 1996.*
7. G. C. Langelaar, J. C. A. van der Lubbe, and J. Biemond, Copy protection for multimedia data based on labeling techniques, in *17th Symposium on Information Theory in the Benelux, Enschede, The Netherlands, May 1996.*
8. F. Hartung and B. Girod, Digital watermarking of raw and compressed video, in *Proc. European EOS/SPIE Symposium on Advanced Imaging and Network Technologies, Berlin, Germany, October 1996.*
9. T. L. Wu and S. F. Wu, Selective encryption and watermarking of MPEG video, in *International Conference on Image Science, Systems, and Technology, CISST'97, June 1997.*
10. J. Zhao and E. Koch, Embedding Robust Labels into Images for Copyright Protection, in *Proceedings of the International Congress on Intellectual Property Rights for Specialized Information, Knowledge and New Technologies, Vienna, Austria, August 1995.*
11. G. C. Langelaar, J. C. A. van der Lubbe, and R. L. Lagendijk, Robust labeling methods for copy protection of images, in *Proc. Storage and Retrieval for Image and Video Databases V, San Jose, California, February 1997.*
12. G. C. Langelaar, R. L. Lagendijk, and J. Biemond, Real-time labeling methods for MPEG compressed video, in *18th Symposium on Information Theory in the Benelux, Veldhoven, The Netherlands, May 1997.*
13. A. Piva, M. Barni, F. Bartolini, and V. Cappellini, DCT-based watermark recovering without resorting to the uncorrupted original image, in *Proceedings IEEE International Conference on Image Processing (ICIP '97), Santa Barbara, California, October 1997.*
14. D. Kundur and D. Hatzinakos, A robust digital image watermarking method using wavelet-based fusion, in *Proceedings IEEE International Conference on Image Processing (ICIP '97), Santa Barbara, California, October 1997.*
15. X.-G. Xia, C. G. Boncelet, and G. R. Arce, A multiresolution watermark for digital images, in *Proceedings IEEE International Conference on Image Processing (ICIP '97), Santa Barbara, California, October 1997.*
16. J. J. K. O Ruanaidh, and T. Pun, Rotation, scale and translation invariant digital image watermarking, in *Proceedings IEEE International Conference on Image Processing (ICIP '97), Santa Barbara, California, October 1997.*
17. ISO/IEC 13818-2:1996(E), *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video,* International Standard, 1996.
18. The SMASH project home page, available *http://www-it.et.tudelft.nl/pda/smash*.
19. Digital Audio Interface, International Standard IEC 958.

GERRIT C. LANGELAAR was born in Leersum, The Netherlands. He received his M.Sc. degree in electrical engineering from Delft University of Technology, The Netherlands in June 1995. His M.Sc. thesis was on artificial intelligence. He is currently working towards his Ph.D. at the same university. His research interests are in the areas of image and video processing with emphasis on watermarking techniques for multimedia data. Part of his research was performed in the European ACTS SMASH-project.

R. (INALD) L. LAGENDIJK received the M.Sc. and Ph.D. degrees in electrical engineering from the Technical University of Delft in 1985 and 1990, respectively. In 1987 he became an assistant professor at the Technical University of Delft. He was a visiting scientist in the Electronic Image Processing Laboratories of Eastman Kodak Research in Rochester, New York in 1991. Since 1993 he has been an associate professor in the Information and Communication Theory Group of the Technical University of Delft. Dr. Lagendijk is author of the book *Iterative Identification and Restoration of Images* (Kluwer, 1991), and co-author of the book *Motion Analysis and Image Sequence Processing* (Kluwer, 1993). He has served as associate editor of the *IEEE Transactions on Image Processing* and is currently a member of the IEEE SP society's Technical Committee on Image and Multidimensional Signal Processing. At present his research interests include signal processing and communication theory, with emphasis on visual communications, compression, analysis, searching, and watermarking of image sequences. Dr. Lagendijk has been involved in the European research projects DART, SMASH, and DISTIMA. He is currently leading several projects in the field of wireless visual communications.

JAN BIEMOND was born in De Kaag, The Netherlands. He received the M.S. and Ph.D. degrees in electrical engineering from

Delft University of Technology, Delft, The Netherlands in 1973 and 1982, respectively. Currently, he is professor and chairman of the Information and Communication Theory Group of the Faculty of Information Technology and Systems at Delft University of Technology. His research interests include multidimensional signal processing, image enhancement and restoration, video compression (digital TV, stereoscopic TV, and HDTV), and motion estimation with applications in image coding and computer vision. He has published extensively in these fields. He was a recipient of the Dutch Telecom Award ''Vederprijs'' in 1986 for his contributions in the area of digital image processing, in particular in image restoration and subband coding. He was a Distinguished Lecturer of the IEEE Signal Processing Society for 1993–1994 and he is a Fellow of the Institute of Electrical and Electronics Engineers (IEEE). In 1983 he was a visiting professor at Rensselaer Polytechnic Institute, Troy, NY, and at Georgia Institute of Technology, Atlanta, GA. He is a former member of the IEEE-SPS Technical Committee on Image and Multidimensional Signal Processing and a member of the IEEE-CAS Technical Committee on Visual Signal Processing and Communications since 1988. He served as the general chairman of the *Fifth ASSP/EURASIP Workshop on Multidimensional Signal Processing, Noordwijkerhout, The Netherlands, September 1987.* Further, he is a former member of the Administrative Committee of the European Association for Signal Processing (EURASIP) and a former member of the Board of Governors of the IEEE-SP Society. He is a former co-editor of the *International Journal on Multidimensional Systems and Signal Processing,* and he serves on the editorial boards of *Image Communication* and the *Journal of Visual Communication and Image Representation.* He is the scientific editor of a series of books on image communication with Elsevier Science BV. Currently, he is a member of the European Science Foundation for European Research Conferences. He was the general chairman of the *1997 Visual Communication and Image Processing Conference (VCIP-97), San Jose, CA.*