Copy Protection for Multimedia Data based on Labeling Techniques

G.C. Langelaar, J.C.A. van der Lubbe, J. Biemond

Department of Electrical Engineering, Information Theory Group
Delft University of Technology
P.O.Box 5031, 2600 GA Delft, The Netherlands
E-mail: *{Gerhard, vdLubbe}@it.et.tudelft.nl*

Abstract

Service providers are reluctant to distribute their multimedia data in digital form because of their fears for unrestricted duplication and dissemination. Therefore robust methods must be developed to protect the proprietary rights of the multimedia data owners and to realize a copy protection mechanism.

In this paper the existing methods for labeling multimedia data are discussed and evaluated. A possibility is given to extend these methods towards a robust copy protection system for new mass storage devices. Some methods suitable for a copy protection method are selected and weak points are discussed. One of these methods is extended to embed a bit sequence instead of one bit in an image and to make it more resistant to lossy compression techniques. Using this method some true color images (size about 500 x 500) were labeled with about 200 bits. The label turned out to be resistant to JPEG compression, with quality parameter set up to 40% (compression rate >1:20).

1. Introduction

Nowadays digital recording devices are available for recording audio. Using personal computers it is also possible to store digital video on a harddisk. The storage capacity of a harddisk is, however, not sufficient to store a complete full resolution home video. For the consumer it would be easier to have one digital storage device that can handle huge amounts of multimedia data. Such a device can replace all other recording equipment in the home, like tape or DAT recorder, VCR and tape streamer.

The aim of the SMASH* project, supported by several companies and universities, is to develop a popular mass-home-storage-device. The development rate of such a digital mass storage system is dependent on not only technical advances, but also on the existence and evolution of adequate protection methods on it. Therefore, robust methods must be developed to protect the proprietary rights of the data owners and to realize a copy protection mechanism limiting the easiness of duplication of multimedia data.

A copy protection system called SCMS [1] (Serial Copy Management System) exists for digital audio recorders, like the DAT, DCC and minidisk recorders. Using this system, a consumer can make only one digital copy of any digital source. Such a copy can not be duplicated further using storage devices equipped with this protection method.

The protection is embedded in the transfer protocol. Together with the music data some sub-code data is transmitted. One bit in this sub-code is called the copy prohibit bit. This bit is set to “one” for every recording. If the consumer tries to record audio data containing a copy prohibit bit, the storage device simply refuses to record.

* This work is funded in part by the European Community under the ACTS contract AC18 (SMASH -**S**torage of **M**ultimedia **A**pplications **S**ystems in the **H**ome), and started in September 1995. More information about the SMASH project can be obtained from the WWW-pages: http://www-it.et.tudelft.nl/pda/smash.

However, this system can not be applied on a mass storage system for multimedia data with several interfaces to different kinds of devices. For example, if data is transferred from the storage device to the harddisk of a personal computer, the copy prohibit bit is lost, because it was only part of the transfer protocol. Therefore, the copy prohibit bit needs to be directly encoded into the audio, image or video signal itself. In this way the bit remains intact across varying data file formats. It is obvious that the copy prohibit bit must be inaudible or invisible for the user and that it must be difficult to remove the bit by using lossy compression techniques, filtering or other processing techniques, that change the data, but do not considerably affect the quality of the data.

By embedding a copy prohibit bit in the data it is still possible to copy the data to devices that are not equipped with this copy protection system. However, if data is copied to such devices it can not directly be played back due to a lower transfer rate of that recording device (e.g. tape streamer) or the amount of data is too big to fit because of a limited storage capacity. So, a few images out of a digital library or some audio fragments can be copied, but it is probably more expensive and time consuming to store this data on other media than buying the original data and using the new mass storage device.

In this paper the existing methods for labeling multimedia data are discussed and evaluated. After that, some methods suitable for a copy protection method as described above are selected and weak points are discussed. One of these methods is extended to embed a bit sequence instead of one bit in an image and to make it more resistant to lossy compression techniques. Finally, conclusions are drawn.

2. Existing Copyright Labeling Techniques

The technique of embedding information in image, video and data is called steganography. It is mainly used in the field of copyright labeling, where data is labeled to identify it uniquely as property of the copyright holder. A label normally consists of a binary serial number or an ASCII text string. Several projects are working or have worked on this subject, like the EC RACE project ACCOPI [2] and the ACTS project TALISMAN [3].

Labels can be added in almost every domain (Spatial, DCT, Wavelet, Fourier, etc.) using different methods. There are two possibilities to extract the label from the image, some methods only use the labeled image, others also uses the original image. The simplest method manipulates the least significant bit of the luminance values or color components of an image, in a manner which is undetectable to the eye [4]. However, this method is not resistant to for instance JPEG compression.

The two following methods embed a label of one bit in the spatial domain. Bender *et al* [5] describe a statistical labeling method called "Patchwork". Using this method, n pairs of image points (a_i,b_i) are randomly chosen. The brightness of a_i is increased by one and the brightness of the corresponding b_i is decreased by one. The expected value of the sum of the differences of the n pairs of points is then $2n$. The authors show that after JPEG compression, with quality parameter set to 75%, the label can still be decoded with a probability of recovery of 85%.

Pitas and Kaskalis [6] describe a similar method. Using this method the picture is split in two subsets of equal size (for example by using a random generator) and the brightness of the pixels of one subset is altered by adding a positive integer factor k. This factor k is calculated using the sample variances of the two subsets. To check the label the difference between the means of the two subsets of pixels is calculated. The expected value is k if a label was added. This method is only resistant to JPEG compression ratios up to 4:1 (quality factor of more than 90%). The major drawback of these two methods is the extremely low bit capacity, usually one bit.

Caronni [7] also describes a method which embeds a bitstream in the luminance values of an image. The image is divided up into blocks. Every pixel in a block is incremented by a certain factor to encode a '1' and is left untouched to encode a '0'. To recover a label, the brightness of each pixel in the labeled image is subtracted from the original one. If the mean of a block of pixel differences exceeds a certain threshold, the corresponding bit is taken as '1', otherwise as '0'. After

JPEG compression, with quality parameter set to 30%, the label can still be recovered. A disadvantage of this method is that the original unlabeled image is required to decode the label.

Zhao and Koch [8] propose a method to embed a bitstream in the DCT domain. The image is divided up into 8x8 blocks (like the JPEG algorithm does). From pseudo-random selected 8x8 blocks the DCT coefficients are calculated. These coefficients are quantized using a quality factor Q and the standard quantization matrix of the JPEG software. Three quantized coefficients are selected and adapted in such a way that they have a certain order in size. For example if a bit '1' must be embedded in a block, the third coefficient must be smaller than the other two. In an earlier proposal by the same authors [9], two instead of three coefficients were used. After JPEG compression, with quality parameter set to 50%, the label can still be recovered. Advantages of this method are that the original unlabeled image is not required to decode the label and that a quite large bitstream can be embedded.

Cox *et al.* [10] embed a sequence of real numbers of length n in an N x N image by computing the N x N DCT and adding the sequence to the n highest DCT coefficients, excluding the DC component. To extract the sequence, the DCT transform of the original image is subtracted from the DCT transform of the labeled one and the sequence is extracted from the highest coefficients. A disadvantage is that the original unlabeled image is required to decode the label.

Boland *et al.* [11] describe a method that works with different image transforms (DCT, Walsh-Hadamard, Wavelet, Fast Fourier). An image is divided into blocks, the mean of the block is subtracted from each pixel in the block and the remaining values are normalized between -127 and 127. The transform is carried out on the image block and some coefficients are modulated to embed a number of bits, for instance by adding one to a coefficient for bit '1' or subtracting one for bit '0'. A reverse transformation is carried out and the original block is replaced by the labeled one. A disadvantage of this method is that the original unlabeled image is required to decode the label. After JPEG compression, with quality parameter set to 90%, a label could be recovered from an image with a bit error rate of 14% using the DCT transform technique, using other transforms the bit error rates were higher.

3. Evaluation labeling methods

The labeling methods described above can add information to an image in an invisible way, but there is always a trade-off between the size of the label, the resistance to JPEG compression and the effect on the image quality, although estimating the quality degradation due to labeling is a completely subjective matter.

The methods, that add the label in the spatial domain, seem to have the lowest bit capacity and the lowest resistance to JPEG compression (methods of Bender, Pitas and Caronni).

Adding the label in another domain sometimes improves the capacity and the resistance. The use of the DCT transform gives the best results (methods of Zhao, Cox and Boland), obviously because the JPEG algorithm makes use of the same DCT transform. The resistance can be increased further if the quantization step is also taken into account (method of Zhao).

If the original unlabeled image can be used together with the labeled one to extract the label, the capacity and the resistance to JPEG compression seem to be higher (methods of Caronni and Cox). In the latter case, the method is also more robust to other attacks, like cropping, rotation, translation, scaling etc. Using the original image some preprocessing can be done before the label is checked. Rotation angles, translation and scale vectors can be estimated and missing parts of the image can be replaced by parts from the original image.

4. Suitable methods for a copy protection system

For the copy protection system described in the introduction, the following requirements must be met:

- The method must have a bit capacity of at least 1 bit, but a bit capacity up to a few hundred bits is preferable, because of extra options like adding timestamps.

- It must be possible to extract the embedded code without using the original unlabeled data.
- The label must be resistant to lossy compression techniques (like JPEG / MPEG), filtering or other processing techniques, that change the data, but do not considerably affect the quality.
- The labeling is allowed to cause degradation of the quality of the data if the data was already labeled before. Normally data is labeled only once. But if a hacker changed for example the image by a slight translation or rotation, the storage device might be unable to read out the original label and deals with the data as new unlabeled data. The new label should now affect the quality.

The only methods which meet these requirements, are the methods of Bender, Pitas and Zhao. However, from these three methods only the last one (Zhao) has a sufficient bit capacity and an acceptable resistance to JPEG compression, the other two must be developed further to achieve the same results. In the next section a proposal is given to extend one of the first methods.

A weak point of the method of Zhao is that the quality of the picture is heavily reduced by a label, which is resistant to JPEG compression up to a quality of 50%. This is illustrated in Figure 1. In the left half of the picture (1a) the unlabeled image and a corresponding zoom view of the shoulder is given. In the right half (1b) the labeled image (quality 50%) and the corresponding zoom view of the shoulder is represented.

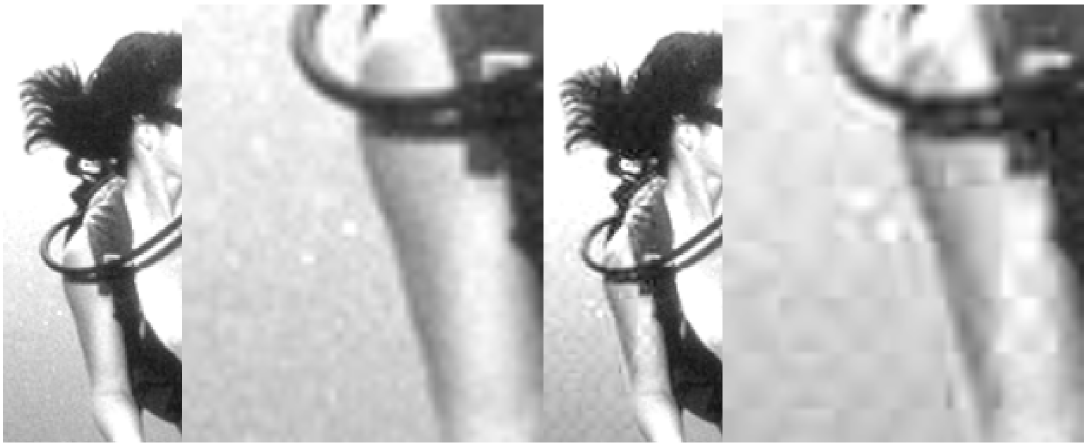

Figure 1a. Unlabeled image and zoom view **Figure 1b.** Labeled image using Zhao's method

If bits are added with a certain quality factor, the quality of many parts in the image (a number of 8x8 blocks) is reduced. Another disadvantage of this method and also of the methods of Bender and Pitas is that the labeling techniques are not resistant to attacks like cropping, rotation, translation and scaling.

5. Extending spatial labeling method

In this section a new block based method is proposed, which adds a bit sequence in the spatial domain. This method is based on the method of Pitas described in the previous section.

Different variants of this method have been tested, but the method as described below gave the best experimental results (concerning the resistance to JPEG compression).

Labeling procedure:
A label consists of a few hundred bits. Each label bit is embedded in a block of luminance values. The width and height of this block are multiples of 8. The X and Y positions of the top corner of the block in the image are also multiples of 8 to be compatible with the YUV based JPEG compression algorithm (e.g. the JFIF standard).

1. First the RGB color image is converted to the YUV domain.
2. A block **B** is pseudo-randomly selected from the image to embed one label bit.
3. A fixed binary pseudo-random pattern of the same size as the block is generated, consisting of the integers **"0"** and **"1"**.
4. The mean $\mathbf{I_o}$ is calculated of the luminance values in the block, where the random sequence is 0. The mean $\mathbf{I_1}$ is calculated of the luminance values in the block, where the random sequence is 1. After that, the difference **Difference_High_Quality_Block($\mathbf{I_0,I_1}$)** is calculated between the two means.
5. In a similar way, the difference **Difference_Low_Quality_Block($\mathbf{I'_0,I'_1}$)** is calculated for a copy **B'** with reduced quality of the block **B** by taking the 8x8 DCT transform, quantizing the coefficients with a certain quality factor **Q** followed by an inverse DCT transform.
6. If **label bit "1"** must be embedded skip step 7.
7. In order to embed the **label bit "0"**, the integer random pattern is subtracted from the original block, if one of the two differences exceeds the value zero. The procedures (4,5,7) are repeated iteratively until both differences are below zero. Step 8 is skipped.
8. In order to embed **label bit "1"**, the integer random pattern is added to the original block, if one of the two differences is smaller than a certain threshold **T**. The procedures (4,5,8) are repeated iteratively until both differences exceed **T**.
9. The procedures (2..8) are applied to all pseudo-randomly selected blocks until all bits of the label are embedded.
10. Finally the YUV values are converted to the RGB domain.

The algorithm is more robust to JPEG (JFIF) compression, if a higher threshold **T** and a lower quality factor **Q** is chosen.

Label extracting procedure:
Reading out the label is simple and is described below.

1. First the RGB color image is converted to the YUV domain.
2. A block **B** is pseudo-randomly selected from the image to read out one bit.
3. The fixed binary pseudo-random pattern of the same size as the block is generated, consisting of the integers **"0"** and **"1"**.
4. The mean $\mathbf{I_o}$ is calculated of the luminance values in the block, where the random sequence is 0. The mean $\mathbf{I_1}$ is calculated of the luminance values in the block, where the random sequence is 1. After that, the difference **Difference($\mathbf{I_0,I_1}$)** is calculated between the two means.
5. If this difference **Difference** exceeds the value zero the bit embedded in the block is one, otherwise zero.
6. The procedures (2..5) are applied to all pseudo-randomly selected blocks until all bits of the label are extracted.

Applying a simple edge-enhance filter to the luminance pixel values before checking the label reduced the percentage of bit errors considerably. The method can further be improved by adapting the random pattern. If the ratio between the numbers of ones and zeros in the random pattern is forced to be 1:4 the labeling is significantly less visible to the human eye, but marginally weaker. If the dotsize of the random pattern is increased to 2x2 instead of 1x1, the robustness increases.

6. Experimental results

Using the method described in the previous section four color images were labeled (see table 1 for more information about these images). The ratio between the numbers of ones and zeros in the random pattern was forced to be 1:4 and the pattern dotsize was adapted as described in the previous section. Each bit was embedded in a block of 32 x 32 pixels, the threshold T was set to 1 and a quality factor of 75% was used.

Table 1. Information about the *labeled* test images.

Name	**Resolution (pixels)**	**Compression ratio using JPEG quality factor of**					
		90%	**80%**	**75%**	**60%**	**50%**	**40%**
Diver	302 x 323	1:9	1:14	1:16	1:23	1:27	1:30
Mountain	733 x 487	1:8	1:11	1:12	1:18	1:21	1:25
Lena	512 x 512	1:11	1:17	1:20	1:28	1:33	1:39
Kielp	720 x 576	1:7	1:10	1:12	1:16	1:18	1:21

In Table 2, Figure 2 and 3 the bit errors in the label are represented, after compressing the images with the JPEG compression algorithm, with quality parameter set to different values.

Table 2. Number of bit errors after JPEG compression *(without / **with edge enhance filtering**)*.

Name	**Label length**	**bit errors after JPEG compression with quality factor of**					
		90%	**80%**	**75%**	**60%**	**50%**	**40%**
Diver	90 bits	0 / **0**	2 / **0**	2 / **0**	7 / **2**	17 / **9**	15 / **7**
Mountain	208 bits	20 / **5**	11 / **3**	3 / **0**	23 / **9**	19 / **5**	43 / **23**
Lena	208 bits	1 / **0**	5 / **0**	6 / **0**	30 / **1**	37 / **5**	50 / **10**
Kielp	208 bits	0 / **0**	1 / **1**	5 / **2**	16 / **6**	25 / **13**	33 / **15**

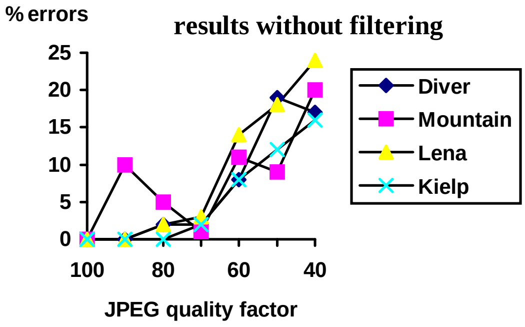

Figure 2. % bit errors after JPEG compression withhout using edge-enhance-filtering

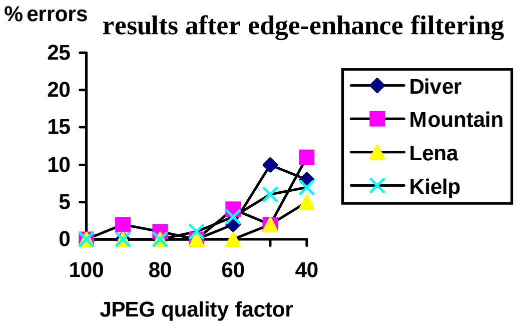

Figure 3. % bit errors after JPEG compression using edge-enhance-filtering

Applying a simple edge enhance filter improves the results considerably, because it amplifies the differences of the adapted and the unaffected luminance values. The maximal percentage of bit errors in the label is only 11% after compressing the image using a quality factor of 40%. Heavy smoothing, obviously, makes the results worse, however the method is immune to light smoothing.

7. Conclusions

Different methods for labeling digital images are investigated. The methods, that add the label in the spatial domain, seem to have the lowest bit capacity and the lowest resistance to JPEG compression. Adding the label in another domain sometimes improves the bit capacity and the resistance. The use of the DCT transform gives the best results, obviously because the JPEG algorithm makes use of the same transform. The resistance can be increased further if the quantization step is also taken into account. If the original unlabeled image can be used together with the labeled one to check the label, the capacity and the resistance to JPEG compression seem to be higher.

Only a few existing labeling techniques are suitable for a copy protection system. However, from these methods only one DCT based method has a sufficient bit capacity and an acceptable resistance to JPEG compression. Therefore, the spatial labeling methods are developed further to achieve the same results. By allowing smaller blocks to embed one label bit, making the embedding level dependent on a lower quality JPEG compressed version of the image and adapting the random pattern, this aim is reached. Using the extended method some true color images were labeled with a few hundred bits. The label turned out to be resistant to JPEG compression, with quality parameter set to 40% (compression rate >1:20).

This method can be improved further by rejecting blocks if the embedding level becomes to high.

A disadvantage of almost all methods mentioned in this paper including the extended one, is that they are not resistant to rotations, cropping, translations and scaling. This problem could maybe be solved by taking into account contour information to find one or two orientation points in the image.

References

[1] Digital Audio Interface, International Standard IEC 958

[2] RACE M 1005: Access control and copyright protection for images (ACCOPI), Workpackage 5, June, 1995

[3] TALISMAN: http://www.tele.ucl.ac.be/IMAGES/ACTS/talisman.html

[4] R.G. van Schyndel, A.Z. Tirkel, C.F. Osborne : "A Digital Watermark", Int. Conf. on Image Processing, volume 2, pages 86-90, IEEE, 1994

[5] W. Bender, D. Gruhl, N. Morimoto : "Techniques for Data Hiding", Proceedings of the SPIE, 2420:40, San Jose CA, USA, February 1995

[6] I. Pitas, T. Kaskalis : "Signature Casting on Digital Images", Proceedings IEEE Workshop on Nonlinear Signal and Image Processing, Neos Marmaras, June, 1995

[7] Caronni G.: "Assuring Ownership Rights for Digital Images", Proceedings of Reliable IT Systems, VIS '95, Vieweg Publishing Company, Germany, 1995

[8] J. Zhao, E. Koch : "Embedding Robust Labels into Images for Copyright Protection", Proceedings of the International Congress on Intellectual Property Rights for Specialized Information, Knowledge and New Technologies, Vienna, Austria, August 1995

[9] E. Koch, J. Zhao : "Towards Robust and Hidden Image Copyright Labeling", Proceedings IEEE Workshop on Nonlinear Signal and Image Processing, Neos Marmaras, June, 1995

[10] I.J. Cox, J. Kilian, T. Leighton, T. Shamoon : "Secure Spread Spectrum Watermarking for Multimedia", NEC Research Institute, Technical Report 95 - 10

[11] F.M. Boland, J.J.K. O Ruanaidh, C. Dautzenberg : "Watermarking Digital Images for Copyright Protection", Proceedings of the 5th International Conference on Image Processing and its Applications, no 410, Edinburgh, July, 1995