Image and video fingerprinting for digital rights management of multimedia data

Nikos Nikolaidis and Ioannis Pitas
Department of Informatics
Aristotle University of Thessaloniki, Thessaloniki 54124, Greece
Tel/Fax: +302310996304
E-mail: pitas@aiia.csd.auth.gr

***Abstract*— Multimedia fingerprinting, also know as robust/perceptual hashing and replica detection is an emerging technology that can be used as an alternative to watermarking for the efficient Digital Rights Management (DRM) of multimedia data. Two fingerprinting approaches are reviewed in this paper. The first is an image fingerprinting technique that makes use of color-based descriptors, R-trees and Linear Discriminant Analysis (LDA). The second is a video fingerprinting method that utilizes information about the appearances of actors in videos along with an efficient search strategy. Experimental performance analysis is provided for both methods.**

I. Introduction

Recent technological advances in the area of multimedia content distribution have resulted in a major reorganization of this trade. Valuable digital artworks can be reproduced and distributed arbitrarily without any control by the copyright holders. Thus, issues related to intellectual property rights protection and management arise.

Numerous systems addressing the issue of copyright protection can be found in the literature, the vast majority of them being based on watermarking. Watermarking is the technique of imperceptibly embedding information within the content of the original medium [1]. Although watermarking has attracted considerable interest from both industry and academia, it bears certain deficiencies. The requirement of embedding information inside a multimedia document before it is made publicly available, implies distortion of these data at a certain extent and automatically excludes data that are already in the public domain and need to be copyright protected. In addition, watermarking is unable to counter content leakages, when an unwatermarked copy of the original artwork is stolen.

In order to overcome these inherent watermarking deficiencies, the scientific community recently started to investigate copyright protection and digital rights management in multimedia data from another perspective i.e. as a problem of similarity of multimedia, data, the similarity being defined in a robust way. These approaches, which come under different names, i.e. multimedia fingerprinting [2], [3], [4], [5], robust or perceptual hashing [6], [7], [8], [9], [10], and replica recognition/detection [11], [12], [13], [14], [15], [16], aim at extracting from the data a feature vector, called perceptual hash, fingerprint or signature, that characterizes them in a unique and discriminative way. This feature vector can be combined with a database of multimedia documents that need to be managed with respect to their intellectual property rights, an appropriate similarity metric and an efficient search strategy in order to devise a DRM system. More specifically, such a system can decide if a query digital item resembles a reference item in the database. If this is indeed the case, the query item is identified as being a copy (replica) of the corresponding item in the database and legal action can be pursued against its owner/distributor if he is not posessing/distributing it in a legal way. In order to be of practical use, the feature vectors and the matching procedure involved in a fingerprinting system should be robust to manipulations that multimedia data might undergo, either due to their distribution and use or due to an intentional attempt to make them unrecognizable by the fingerprinting system. The major benefit of fingerprinting stems from the fact that, unlike watermarking, no information needs to be embedded within the image content, thus ensuring perfect quality for the data to be protected and furthermore making the system applicable to data that are already in the public domain. It should be noted here that the term fingerprinting as used in this and other papers, should not be confused with the fingerprinting watermarking which is essentially a variant of watermarking.

The basic hypothesis behind the aforementioned approach is that every multimedia document shares enough information with its modified copies to allow their identification as such, and yet enough discriminative information with respect to other data to allow for their identification as non-relevant. Furthermore, it is assumed that the modified data maintain sufficient quality and resemblance to the originals. Severely distorted copies are of no interest for a fingerprinting system since their commercial value is reduced. The problem of multimedia fingerprinting bears certain similarities with that of content-based indexing and retrieval but has also important differences. The major difference between fingerprinting and retrieval is that the similarity criterion is usually looser in retrieval, since the user is often interested not only in copies of a multimedia item, but also in different items that are perceptually similar to it. Moreover, the requirement of robustness to manipulations is not applicable to retrieval.

In this paper two fingerprinting systems are described. The first system is an image fingerprinting system that utilizes a database of original images that can be queried with a suspect image and decide whether this image is a possibly modified copy of a stored original. Images are represented by a feature vector comprising of color-based descriptors. The system utilizes a multidimensional indexing structure based on R-Trees. Although substantially reduced, the probability that the R-Tree returns more than a single image as candidates for being the originals of the query is existing and prevents the system to decide unambiguously. Linear Discriminant Analysis , preceded by Principal Component Analysis (PCA) is applied in order to reformulate the solution space and yield more discriminant image representations. A more detailed description of this system can be found in [16]. The second system makes use of information about the appearance of faces of distinct individuals (e.g. actors), in order to characterize a video segment in a robust way and use this information for video fingerprinting. Signals that show whether a certain actor appears or not in each frame of the video are used as feature vectors or signatures in this case. Additional details for this system (when used in a video indexing framework) can be found in [17].

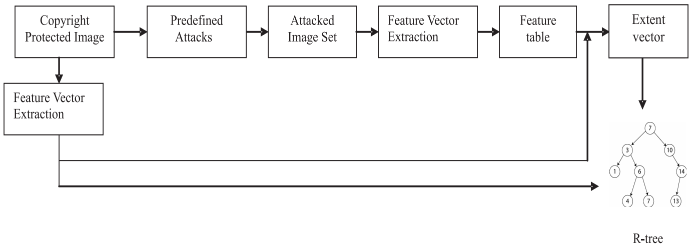

Fig. 1. Database organization of the image fingerprinting system.

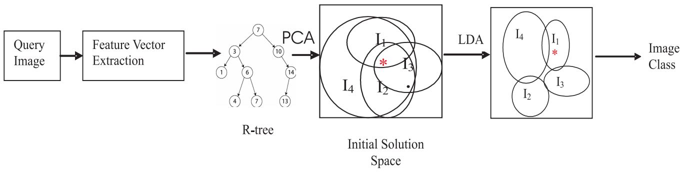

Fig. 2. Searching the database for verifying whether a query image is a copy of an image in the database or not.

II. Image Fingerprinting Using R-trees and Linear Discriminant Analysis

A. System Overview

The construction of the proposed fingerprinting - image replica detection system can be separated to two independent phases. The first phase deals with the database organization and construction. Each time a new original, copyright protected image is added into the database, the image is subjected to a series of predefined attacks (image manipulations) selected according to the system's design specifications. Feature vectors are extracted from each attacked version resulting in a feature table which contains samples from the feature space neighborhood of the original image. The latter is utilized for the calculation of an extent vector that specifies the neighborhood extent for each original image . Finally, the original image is indexed within the R-Tree structure, according to the extent vector (Fig. 1).

The second phase implements the actual fingerprinting functionality, once the database has been organized. An arbitrary image is submitted as a query to the indexing structure. The R-Tree prunes the redundant branches according to the query image feature vector and results inprovides a set of candidate images or an empty set. The next step attempts to enhance the system performance through LDA, preceded by a PCA dimensionality reduction step. Finally, the system returns the closest image based on some similarity metric. Alternatively, the query image may be found to reside outside the formulated neighborhoods. In this case the result is an empty set. Thus, the decision that the query image is not a (possibly modified) copy of the images in the database may take place either during the R-Tree traversal or after the application of LDA. The way the queries are handled is demonstrated in Fig 2.

B. Feature Extraction Method

The proposed system is based on work conducted in [18] for image feature extraction. In this work a comparative evaluation of various feature extraction methods has been performed. All methods are based on the color histogram and try to benefit from its inherent resilience to a number of common manipulations and especially geometrical transforms. Dimensionality reduction is performed by quantizing the color histogram distribution with respect to a specific color palette. The comparative evaluation indicates that the best tradeoff between compaction and retained information is accomplished when the quantization scheme utilizes the Macbeth color palette for constructing a feature vector containing 24 scalar values. It must be noted that the scheme described here can be combined with other feature vectors.

C. Indexing Multidimensional Feature Vectors

In order to reduce computational cost and improve accuracy of the system an R-Tree [19] was introduced. Originally, R-Trees were created to index spatial objects using their bounding boxes (BBs). For a given image query, the R-Tree returns all the records whose BBs include the image query. Our system works under the assumption that the features of modified copies of an original image are localized around those corresponding to the original image. Therefore, the used R-Tree is constructed by associating a bounding box to every original image. These bounding boxes are defined using an extent vector. In order to determine the extent vector for each image in the database, we simulate all attacks that an image may undergo and we wish the system to be able to withstand. Thus, before inserting an original copyright protected image into the database, a series of predefined attacks are performed. The produced images are used for determining the extent vector.

More specifically, a feature vector is extracted from every modified (attacked) version of an image and the distances in each feature dimension between each attacked image and the original image are calculated. The maximum distance for each dimension is selected as the extent in this dimension. Thus, the aforementioned procedure derives for each original image an extent vector consisting of 24 scalar values that determines the extent of the neighborhood in each feature dimension.

It is obvious that the extent vector selection is crucial for the system behavior. In our case we have chosen an implementation where a different extent is kept for every image and for every feature vector dimension. Moreover, in order to fine-tune the system performance, a constant a that is multiplied with the values of all elements in the extent vector was introduced. Changing the value of a allows for extending or shrinking the bounding boxes and thus modifying the system performance.

D. Applying Linear Discriminant Analysis

The fact that the R-Tree can return more than one original image as candidates for being related to the query image does not allow the system to decide unambiguously. In order to obtain a single result and, at the same time, reduce the number of decision errors LDA [20] was used for discriminant feature selection. Prior to the application of the LDA, a dimensionality reduction step using PCA is performed. By eliminating the dimensions that correspond to the smaller eigenvalues, an implicit denoising of the data is achieved. The set of participating classes in the LDA coincides with the set of images (classes) returned by the R-tree. Each of these classes is comprised of the feature vector of the original image along with the feature vectors of its attacked versions. These are the observations used for calculating the class statistics. The LDA space is trained every time a query is submitted. The result of LDA is a linear transformation $\mathbf{W}_o$ that transforms and/or reduces the dimensionality of the image feature vectors $\mathbf{x}_k$ as:

$$\acute{\mathbf{x}}_k = \mathbf{W}_o^T \mathbf{x}_k. \qquad (1)$$

The goal of the linear transformation $\mathbf{W}_o$ is to maximize the between class scatter while minimizing the within class scatter. By projecting the samples to the newly created solution space, better separation of classes is achieved. A similarity metric is then used to find the closest class (image) and an extent vector is used to accept or reject the query image as a copy of an original image. The selection of this vector is done using a procedure analogous to the one described in Section II-C.

E. Experimental Performance Evaluation

A sample of 2.232 original images were used to compose the database of copyright protected images. The images were selected so as to form 12 different content categories, each corresponding to a world famous cultural monument. Evaluation of the system performance over a database containing groups of similar images was done in an effort to assess its behavior under the least favorable situation.

A training set comprised of attacked versions of the originals is involved in two different stages of the fingerprinting system functional chain. Initially, it is utilized for estimating the optimal extent value for each image feature dimension, i.e., evaluating the extent vector which is used in the R-Tree bounding boxes. Moreover, it is used for providing the samples for the evaluation of the linear transform $\mathbf{W}_o$ in the LDA and deriving the extent vectors used in the LDA space. Essentially, the goal of the training set is to effectively model all possible distortions that an original image may undergo.

TABLE I
ATTACKS ON ORIGINAL IMAGES (TRAINING IMAGE SET).

Attack Category & Severity Range	Step	Produced Images
Jpeg compression 10-90 (quality factor)	5	17
Rotation 1^o-359^o (degrees)	5^o	36
Resizing 0.3 - 2.0 (scale factor)	0.1	16
Cropping 99% - 50% (Remaining Portion)	5%	8
Total		77

TABLE II
ATTACKS ON ORIGINAL IMAGES (EXTENDED IMAGE SET USED IN TESTING).

Attack Category & Severity Range	Step	Produced Images
Jpeg 10-90 (quality factor)	1	90
Rotation 1^o-359^o (degrees)	1^o	360
Resizing 0.3 - 2.0 (scale factor)	0.05	35
Cropping 99% - 50% (Remaining Portion)	1%	50
Total		535

A total number of 77 attacks per original image were applied for constructing a training set consisting of 171.864 images. Table I summarizes the attacks, which are performed by the preprocessing procedure applied each time a new original image is being inserted into the image database.

In order to evaluate the performance of the proposed fingerprinting system two sets of experiments were conducted. In the first set the performance when the system is being queried with images that are replicas of the images in the database was evaluated. The percentage of the query images that are falsely identified as not being copies (false rejection rate) as well as the percentage of the query images that are identified as copies but are assigned to a different image in the database (misclassification rate) were evaluated. An extended set containing more attacks than the training set was utilized during this experiment. This query set contains attacked versions derived by the same attack categories and within the same severity range as those used in training, though using more dense attack steps (Table II). 240 images that reside in the database were randomly selected for this experiment and the 535 modified versions of each of them were used as query images (total: 128400 images). It is obvious that, in this case, images not included in the training set are incorporated in the query set, thus introducing a sense of fairness. For the queries on this set, the false rejection and misclassification rates were 0.54% and 1.28% respectively.

The second set of experiments aimed to evaluate the performance of the proposed system when being queried with images that are not copies of the images in the database. The percentage of such query images that are falsely identified as copies (false acceptance rate) was used as a performance measure in this case. 450 images that were not included in the image database formulated the content of this query set. The false acceptance rate in this case was equal to 7.33%. Currently, a variant that further reduces this error by introducing after, the R-tree step, a module that performs image similarity computation using SIFT features [21] is under investigation.

III. Video Fingerprinting Using Face-related Information

The method presented in this section takes advantage of information about the existence of faces of distinct individuals (e.g. actors), in order to characterize a video segment in a robust way and use this information for video fingerprinting. Using face-related information for video fingerprinting or indexing is not a new idea. However, most works until now [22], [23] are works on face recognition with a view to its application on indexing. In the work described in this section, we do not propose a face detection and recognition method since ample work has been performed on both subjects [24], but we investigate the effect of different parameters of the face detection and recognition process on the retrieval performance of our method. The advantages of the proposed algorithm are firstly that it is based on semantic information, and is thus robust with respect to video noise and manipulations, secondly that it is convolution-based and thus robust to change of query segment boundaries and to malfunctions of the face detector and recognizer, and thirdly that it is well suited to large video databases. The details of the proposed system will be described in the next sections.

A. Format of Signature

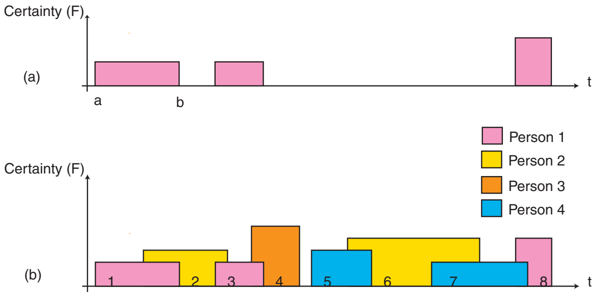

Fig. 3. Example of the characterization of a video segment by quartets. (a) Signature for a single person. (b) Signature of a video segment. Different colors correspond to distinct individuals. Signature quartets are represented by rectangles.

Let $\mathbf{V} = \{f_1 \ f_2 \ \ldots \ f_N\}$ be a video consisting of a number of consecutive frames $f_n, n = 1, \ldots, N$. that we wish to characterize through an appropriately constructed signature. Let $\mathbf{S} = \{s_1 \ s_2 \ \ldots \ s_M\}$ be the set of all the individuals $s_m, m = 1 \ldots M$ that have been imaged in the video. Optionally, with no loss of generality, we can assume $\mathbf{S}$ to contain only the individuals of interest.

Let us then assume a face detector and recognizer F whose output is the certainty:

$$G(n,m) = \text{Prob}\{s_m \text{ is imaged in } f_n\} \qquad (2)$$

The face recognizer can either be of the hard (binary) decision type, in which case $G(n,m) \in \{0,1\}$ or a soft one, in which case $G(n,m) \in [0,1]$. For each person s_m, it is then possible to find all frame intervals $I_i^m = [a_i^m, b_i^m]$ such that $G(n,m) > 0, \ n \in [a_i^m, b_i^m]$ and $I_i^m \not\subset I_j^m, \forall i \neq j$. Using I_i^m we can then define a *face occurrence* $F_i^m = \overline{F(n,m)}|_{n=a_i^m}^{b_i^m}$ as the average certainty within the interval I_i^m, that a specific person is imaged. So we can approximate $G(n,m)$ with

$$F(n,m) = \sum_i F_i^m \left[u(n - a_i^m) - u(n - b_i^m)\right] \qquad (3)$$

where $u(n)$ is the unit step function.

For each person s_m, her signature triplets (F_i^m, a_i^m, b_i^m), $i = 1, \ldots, N$ form a pulse series in the video time domain, as can be seen in Figure 3(a). Therefore, the video $\mathbf{V}$ is characterized by a *signature* consisting of quartets of values $(s_m, F_i^m, a_i^m, b_i^m), m = 1, \ldots, M, \ i = 1, \ldots, N$. An example of such a signature signal is given in Figure 3(b). Each quartet corresponds to a unique face appearance, i.e., it conveys the information that person s_m has been detected from frame a_i^m to frame b_i^m with a confidence of F_i^m. The extraction of the signature described above from the video is straightforward if a face detection and recognition module is available.

B. Signature Similarity

Let us assume two signatures $F_1(n,m)$ and $F_2(n,m)$, derived as per Equation (3), which are extracted from two video segments and refer to a common set of faces $\mathbf{S}$. Let us assume that we move F_2 by a specific displacement d. We will define as *co-occurrence* C the evidence that the two signatures are the same. At a specific frame n for a specific person m and in the case of a binary decision recognizer, such evidence exists if and only if the person exists at both signatures, i.e. $C_{hard}(d,n,m) = F_1(n,m) \cdot F_2(n+d,m)$. If the detector produces a detection certainty, the evidence that a specific person occurs in both signatures depends on the certainty of detection. Thus in this case $C_{soft}(d,n,m) = \min(F_1(n,m) \cdot F_2(n+d,m))$. The overall evidence of similarity of $F_1(n,m)$ and $F_2(n,m)$ for a specific displacement can be computed by summing over all frames and persons. If the lengths of the two video segments are N_1 and N_2, and assuming without loss of generality that $N_1 \leq N_2$, C can be regularized by dividing by N_1 and the number of possible persons, M. In the case of a hard detector (whose output is 0 or 1) this corresponds to:

$$C_{hard}(d) = \sum_{n=1}^{N_1} \sum_{m=1}^{M} \frac{F_1(n,m) \cdot F_2(n+d,m)}{N_1 M} \qquad (4)$$

In the case of a detector that produces detection certainties, we have:

$$C_{soft}(d) = \sum_{n=1}^{N_1} \sum_{m=1}^{M} \frac{\min(F_1(n,m), F_2(n+d,m))}{N_1 M} \qquad (5)$$

Geometrically, C can be visualized as the overlap between the rectangles that correspond to the quartets which refer to the same person in the two signatures. The similarity of the two signatures is defined as the maximum value of co-occurrence $C_{max} = \max_d C(d)$, obtained when sliding one signature in relation to the other. C_{max} is computed by a process similar to a convolution. Thus C_{max} tends to be insensitive to small changes in the signature, such as splits, shifts, changes in height or in width of the quartet rectangles. Having established a method for computing the similarity between two signature segments, searching for a specific video in a database entails simply comparing a candidate segment with the whole database and declaring a match when the similarity exceeds a certain threshold. An algorithm that does this in near-logarithmic time with respect to the size of the database is presented in the next section.

C. Search-Matching Algorithm

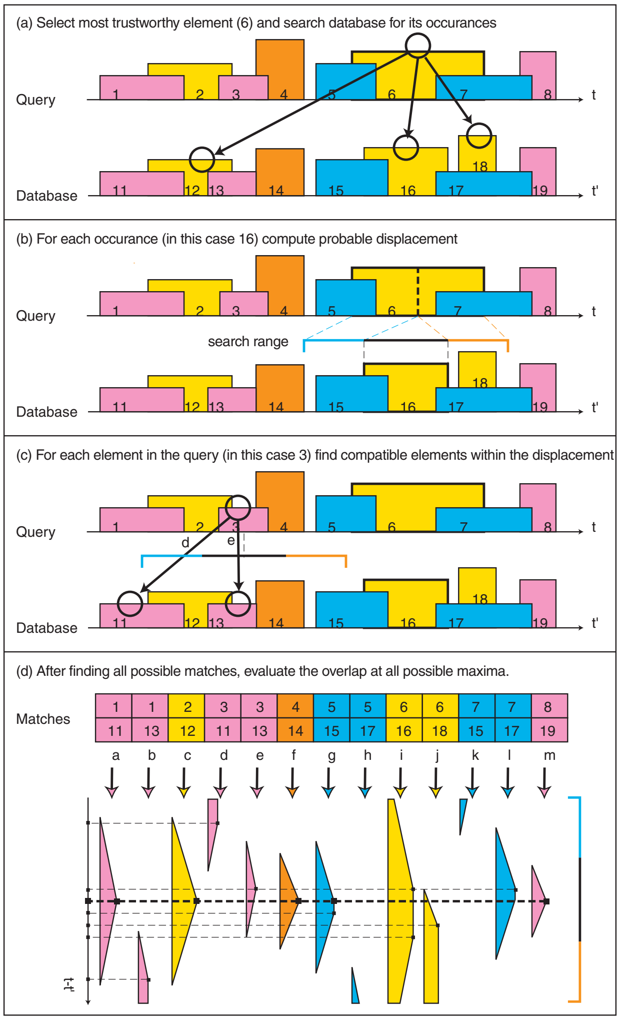
(a) Select most trustworthy element (6) and search database for its occurances
Query
Database
1
2
3
4
5
6
7
8
t
11
12
13
14
15
16
17
18
19
t'
(b) For each occurance (in this case 16) compute probable displacement
search range
(c) For each element in the query (in this case 3) find compatible elements within the displacement
d
e
(d) After finding all possible matches, evaluate the overlap at all possible maxima.
Matches
1 1 2 3 3 4 5 5 6 6 7 7 8
11 13 12 11 13 14 15 17 16 18 15 17 19
a b c d e f g h i j k l m

Fig. 4. Graphical overview of the signature search and matching algorithm. Different colors correspond to distinct individuals. Signature quartets are represented by numbered rectangles.

In the following, when we declare a sub- or super-scripted Q, we will assume it is a quartet of the form $Q = \{s, F, a, b\}$, where s, F, a and b have the same sub- and super-scripts as Q. Sets of quartets will be noted in bold.

When the video database is initialized, a database index I_{sa} is created over all the signature quartets $\mathbf{Q}^{db}$ in the database, indexing them first on the person identity s and then on the start frame a. Two other indexes I_a and I_b are created based on start frame a and end frame b alone. These indexes are crucial for enabling near-logarithmic access to the quartets in the database.

The following algorithm (illustrated in Figure 4) is proposed for finding matching segments in the database with respect to a query segment $\mathbf{V}_{query}$, which is characterized by a signature consisting of a set of quartets $\mathbf{Q}^{query}$:

1) Find the quartet in $\mathbf{Q}^{query}$ that has the greatest area (duration × certainty) in order to use it as a base for searching, and name it the *trusted* quartet Q^{trust}. Thus the trusted quartet has the following property:

$$F^{trust}(b^{trust} - a^{trust}) = \max_j F_j^{query}(b_j^{query} - a_j^{query}) \tag{6}$$

2) Find (through the index I_{sa}) all quartets $\mathbf{Q}^{base}$ in the database that refer to the same person as Q^{trust}:

$$\mathbf{Q}^{base} = \{Q^{base} \in \mathbf{Q}^{db} : s^{base} = s^{trust}\} \tag{7}$$

These will be used as the base for evaluating the segments around them, and be named *base* quartets (Figure 4a).

3) For each base quartet $Q_i^{base} \in \mathbf{Q}^{base}$ found in the previous step:
   a) Add the pair consisting of the current base quartet Q_i^{base} and the trusted quartet Q^{trust} into a new list L, which will contain pairs of compatible quartets, i.e. quartets from the candidate segment (in the database) and the query segment which refer to the same person.
   b) Calculate a displacement window $[a_i^{disp}, b_i^{disp}]$, centered on Q_i^{base}, for finding possible matches in the database (Figure 4b), where:

$$b_i^{disp} = \frac{(b^{base} - a^{base})}{2} + \frac{(b^{trust} - a^{trust})}{2} \tag{8}$$

$$a_i^{disp} = -b_i^{disp} \tag{9}$$

   c) Then using the current base quartet Q_i^{base}, which we have found in the database, do the following for each query quartet $Q_j^{query} \in \mathbf{Q}^{query}$:
      i) Find (through the database indices I_a and I_b) the set of compatible quartets $\mathbf{Q}_{ij}^{comp}$ in the database, i.e. those quartets that belong to person s_j^{query} and which overlap with a window of size $b_i^{disp} - a_i^{disp}$ which is centered on Q_i^{query}.
      ii) If no quartets are found, increment a counter n. If, for all query quartets examined so far for the current base quartet Q_i^{base} we have $n > T_{reject}$, where T_{reject} a threshold, then proceed to the

next database quartet Q_{i+1}^{base} that has the same person with Q^{trust}.

iii) Add the pairs consisting of Q_j^{query} on the one hand, and all the recovered $Q_{ij}^{comp} \in \mathbf{Q}_{ij}^{comp}$ on the other, into the list L.

d) Extract from list L a pair of quartets that have been accumulated in the above steps. Name the pair Q_l^{left}, Q_l^{right}. Then, for each pair:

i) Evaluate the area of overlap v_{il} of Q_l^{left}, Q_l^{right} for all displacements d_{il} between the query segment and the candidate segment that correspond to possible maxima of the value of this area. These displacements can be proven to be those and only those that have $a^{left} + d_{il} = a^{right}$ or $b^{left} + d_{il} = b^{right}$.

ii) Select the maximum match quality $v_i^{optimal} = \max_l v_{il}$ and also keep the corresponding displacement $d_i^{optimal}$. If all Q_i^{base} were rejected due to T_{reject}, do not return a $v_i^{optimal}$ (effectively set it to ∞).

e) Clear the list L.

4) Select the final similarity $v^{optimal} = \max_i v_i^{optimal}$. If no $v_i^{optimal}$ were returned due to T_{reject}, the algorithm returns a result of *"not found"*. If the difference between this similarity and the area of the original query segment (i.e. the error) is below a threshold T_v (which depends on the size of $\mathbf{Q}^{query}$), then declare a match, otherwise declare no match. Also keep the corresponding displacement $d^{optimal}$.

5) Optionally, if no match is found repeat all above for the next most trustworthy quartet. In our experiments we have done so.

D. Experimental Performance Evaluation

Our aim was to evaluate the performance of the proposed video fingerprinting method when applied on large video databases. However, the effort of applying different types of face detectors and recognizers on such a database, in order experimentally test the performance of the proposed method, is extremely high. Thus, we performed the experimental testing of our algorithm on appropriately constructed artificial data. We have formulated a probabilistic model which describes the ground truth of the appearance of faces in videos, and a second probabilistic model which describes the imperfect behavior of the face detection and recognition modules when used to derive the signature from the query segment. The face appearance ground truth sequences are modified by the face detector and recognizer model. The distributions used to model the random variables appearing in these models were selected by using a combination of statistical testing and analysis of the physical meaning of the variables whereas the parameters of the random variables of the model (mean, standard deviation etc) were derived by a manually annotated moderately large corpus of video data. The output of these models is a set of video signatures consisting of quartets.

The following types of noise, introduced during face detection and recognition, have been considered:

1) Change of quartet start and end frames (hereafter called *face detector noise*). This is one of the most typical errors made by face detectors and trackers. Exponential noise has been added to the start time of a quartet, and zero mean Gaussian noise to the end time of the quartet. The standard deviation of the noise varied from 1 to 2 seconds, for both the start and end frames of the quartets. The mean of the noise was zero in the case of a quartet's end frame, and equal to the standard deviation in the case of a quartet's start frame (since the distribution is exponential).
2) Change of the person's identity in a quartet (face recognizer noise). This is a typical error made by face recognizers. Here we assumed a probability (between 5% and 10%) that a person's identity would be randomly changed to another one.

We should note that no explicit modelling of phenomena such as compression, cropping, video noise etc are required. Such manipulations ultimately only affect the output of the face detection and recognition modules and, thus, their effect can be included in the model of the employed face detection/tracking/recognition module.

In order to test our algorithm we have run a series of experiments to verify its performance when using query video segments that existed in the database, and when using query video segments that did not exist in the database.

As already mention in section II-E, two sorts of errors can occur in the first case: false rejection and misclassification. The set of experiments in this case involved an artificial signature database of 1000 videos, each 60 minutes long. From this database we randomly extracted 3 sets of 100 segments each. The segments in the three sets contained 32, 48 and 64 quartets respectively, or equivalently 5, 7.5 and 10 minutes of video each. On each set we added noise representing face detector and recognizer errors, as described above, and then proceeded to seek them in the database.

In the second case, we used the model described above to create a set of 1000 videos that were different from the ones in the database. Since the content of the new videos was completely unrelated to those in the database, there was no need to alter them in order to represent failures in face detection and recognition. Three sets of 1000 segments each, containing 32, 48 and 64 quartets per segment, were derived and used as query segments. Using a value of T_u equal to 30%, we ran the fingerprinting algorithm on the same artificially created face signature database described above. False acceptance is the only type of error in this case.

The results are given in Table III, where the strength of the detector noise is given as the mean deviation of the change in the start and end frames of the quartets (in seconds), and the strength of the recognizer noise is given as the percentile probability of false recognition. The results show that the method performs very satisfactorily, especially when large query segments are used. For example, for a query segment

TABLE III
PERFORMANCE OF THE VIDEO FINGERPRINTING ALGORITHM

Threshold T_v		30%		
Query Length (quartets)		32	48	64
False Acceptance (%)		6.7	3	2.1
Recognizer noise [a]	Detector Noise [b]	**False Rejection (%)**		
5%	1sec	2	4	3
10%	1 sec	7	4	4
5%	2 sec	4	3	4
10%	2 sec	11	7	12
Recognizer noise	Detector Noise	**Misclassification (%)**		
5%	1 sec	0	0	0
10%	1 sec	0	0	0
5%	2 sec	4	1	1
10%	2 sec	2	0	0

[a] Mean deviation of the noise added to start and end quartets.
[b] Probability of false recognition.

of 64 quartets, and with moderate noise (2 seconds detector noise and 5% recognizer noise) the false acceptance rate is 2.1%, the false rejection rate is 4% and the misclassification rate 1%.

The computational performance of our algorithm was also evaluated by using artificial video databases The experiments proved that the length of the query segments did not influence the search time and that the performance of the algorithm is near-logarithmic with respect to the size of the database, requiring just 41 seconds for a search in a database consisting of 10000 videos of duration 60 minutes each.

CONCLUSIONS

Multimedia fingerprinting is an efficient alternative to watermarking, having the additional advantage of being a passive technique, i.e. one that does not alter the content of the data. Two different fingerprinting approaches have been presented in this paper. The first approach utilizes color-based descriptors along with R-trees and LDA in order to achieve identification of (possibly modified) copies of images from a database of originals. The second method deals with video data and combines semantic (and thus robust to manipulations) information about the appearances of actors in videos with a convolution-like search strategy to achieve the same goals. Experimental performance analysis shows that the proposed techniques can be used for the efficient DRM of images and video.

ACKNOWLEDGMENT

This work has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

REFERENCES

[1] A. Tefas, N. Nikolaidis, and I. Pitas, “Watermarking techniques for image authentication and copyright protection,” *in The Handbook of Image and Video Processing, 2nd edition, edited by Al Bovik, Elsevier*, 2005.

[2] J. Oostveen, T. Kalker, and J. Haitsma, “Feature extraction and a database strategy for video fingerprinting,” in *Proc. 5th International Conference on Recent Advances in Visual Information Systems (VISUAL 2002)*, March 2002, pp. 117–128.

[3] J. Haitsma and T. Kalker, “A highly robust audio fingerprinting system,” in *2002 International Conference on Music Information Retrieval (ISMIR 02)*, October 2002.

[4] J. Seo, J. Haitsma, T. Kalker, and C. Yoo, “Affine transform resilient image fingerprinting,” in *2003 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP 03)*, vol. III, April 2003, pp. 61–64.

[5] J. S. Seo, J. Haitsma, T. Kalker, and C. D. Yoo, “A robust image fingerprinting system using the radon transform,” *Signal Processing: Image Communication*, vol. 19, pp. 325–339, 2004.

[6] C. Y. Hsu and C. Lu, “Geometric distortion-resilient image hashing system and its application scalability,” in *ACM International Conference on Multimedia (Proceedings of the 2004 workshop on Multimedia and security)*, Magdeburg, Germany, 2004, pp. 81–92.

[7] S. Yang and C. Chen, “Robust image hashing based on SPIHT,” in *International Conference on Information Technology: Research and Education (ITRE 05)*, June 2005, pp. 110–114.

[8] A. Swaminathan, Y. Mao, and M. Wu, “Image hashing resilient to geometric and filtering operations,” in *Proc. of IEEE Workshop on Multimedia Signal Processing (MMSP’04)*, Siena, Italy, Sept. 2004, pp. 355–358.

[9] R. Venkatesan, S. Kaon, M. H. Jakubowski, and P. Moulin, “Robust image hashing,” in *2000 IEEE International Conference on Image Processing (ICIP 00)*, September 2000.

[10] M. K. Mihcak and R. Venkatesan, “New iterative geometric methods for robust perceptual image hashing,” in *ACM Workshop on Security and Privacy in Digital Rights Management,*, vol. LNCS 2320, 2001, pp. 13–21.

[11] Y. Ke, R. Sukthankar, and L. Huston, “An efficient parts-based near-duplicate and sub-image retrieval system,” in *Proceedings of the 12th annual ACM international conference on Multimedia*, New York, USA, 2004, pp. 869–876.

[12] S. Roy and E.-C. Chang, “Watermarking with retrieval systems,” *ACM Multimedia Systems*, vol. 9, no. 5, pp. 433–440, March 2004.

[13] S. Roy, E.-C. Chang, and K. Natarajan, “A unified framework for resolving ambiguity in copy detection,” in *Proceedings of the 13th annual ACM international conference on Multimedia*, Singapore, 2005, pp. 648–655.

[14] A. Qamra, Y. Meng, and E. Chang, “Enhanced perceptual distance functions and indexing for image replica recognition,” *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 379–391, March 2005.

[15] Y. Maret, S. Nikolopoulos, F. Dufaux, T. Ebrahimi, and N. Nikolaidis, “A novel replica detection system using binary classifiers, R-trees and PCA,” in *2006 IEEE International Conference on Image Processing (ICIP 06)*, Atlanta, GA, 2006.

[16] S. Nikolopoulos, S. Zafeiriou, P. Sidiropoulos, N. Nikolaidis, and I. Pitas, “Image replica detection using R-trees and linear discriminant analysis,” in *2006 IEEE International Conference on Multimedia and Expo (ICME 06)*, Toronto, Canada, 2006.

[17] C. Cotsaces, N. Nikolaidis, and I. Pitas, “Video indexing by face occurrence-based signatures,” in *2006 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP 06)*, vol. II, Toulouse, France, 2006, pp. 137–140.

[18] M. Gavrielides, E. Sikudova, and I. Pitas, “Color-based descriptors for image fingerprinting,” *IEEE Transactions on Multimedia*, vol. 8, no. 4, pp. 740–748, August 2005.

[19] V. Gaede and O. Gunther, “Multidimensional access methods,” *ACM Computing Surveys*, vol. 30, no. 2, pp. 170–231, 1998.

[20] R. Duda, P. Hart, and D. Stork, *Pattern Classification (2nd ed.)*. Wiley Interscience, 2000.

[21] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, November 2004.

[22] S. Eickeler, F. Wallhoff, U. Iurgel, and G. Rigoll, “Content-based indexing of images and video using face detection and recognition methods,” in *Proc. IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001)*, May 2001.

[23] S. Satoh, “Comparative evaluation of face sequence matching for content-based video access,” in *Proc. 4th International Conference on Automatic Face and Gesture Recognition(FG2000)*, 2000, pp. 163 – 168.

[24] W. Zhao, R. Chellappa, P.-J. Phillips, and A. Rosenfeld, “Face recognition: A literature survey,” *ACM Computing Survey*, vol. 35, pp. 399–458, 2003.