Available online at www.sciencedirect.com

SCIENCE DIRECT®

Signal Processing: *Image Communication* 19 (2004) 325–339

SIGNAL PROCESSING:
IMAGE
COMMUNICATION

www.elsevier.com/locate/image

A robust image fingerprinting system using the Radon transform

Jin S. Seo[a,*], Jaap Haitsma[b], Ton Kalker[b], Chang D. Yoo[a]

[a] *Department of EECS, KAIST, 373-1 Guseong Dong, Yuseong Gu, Daejeon 305-701, South Korea*
[b] *Philips Research Eindhoven, Prof. Holstlaan 4, Eindhoven 5656AA, The Netherlands*

Received 6 May 2003; received in revised form 12 November 2003; accepted 1 December 2003

Abstract

With the ever-increasing use of multimedia contents through electronic commerce and on-line services, the problems associated with the protection of intellectual property, management of large database and indexation of content are becoming more prominent. Watermarking has been considered as efficient means to these problems. Although watermarking is a powerful tool, there are some issues with the use of it, such as the modification of the content and its security. With respect to this, identifying content itself based on its own features rather than watermarking can be an alternative solution to these problems. The aim of fingerprinting is to provide fast and reliable methods for content identification. In this paper, we present a new approach for image fingerprinting using the Radon transform to make the fingerprint robust against affine transformations. Since it is quite easy with modern computers to apply affine transformations to audio, image and video, there is an obvious necessity for affine transformation resilient fingerprinting. Experimental results show that the proposed fingerprints are highly robust against most signal processing transformations. Besides robustness, we also address other issues such as pairwise independence, database search efficiency and key dependence of the proposed method.

Keywords: Fingerprinting; Robust hashing; Robust matching; Content identification; Image indexing

1. Introduction

Multimedia fingerprinting (also known as robust hashing) is an emerging research area that is receiving increased attention. Fingerprints are perceptual features or short summaries of a multimedia object. This concept is an analogy with cryptographic hash functions which map arbitrary length data to a small and fixed number of bits [27]. Although cryptographic hashing is a proven method in message encryption and authentication, it is not possible to directly apply it to multimedia fingerprinting. Multimedia contents often undergo various manipulations during distribution including compression, enhancement, geometrical distortions and analog-to-digital conversion that may preserve perceptual value. However, cryptographic hash functions are bit sensitive: an alteration of a single bit in the

*Corresponding author.

E-mail addresses: jsseo@kaist.ac.kr (J.S. Seo), jaap.haitsma@philips.com (J. Haitsma), ton.kalker@ieee.org (T. Kalker), cdyoo@ee.kaist.ac.kr (C.D. Yoo).

0923-5965/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.image.2003.12.001

Nomenclature

$H(\ )$	hash or fingerprint function
$H_K(\ )$	hash or fingerprint function with key K
$f(x,y)$	an image signal in spatial coordinate (x,y)
$g(s,\theta)$	the Radon transform of an image
$c(l,\theta)$	the normalized auto-correlation of each radial projection
$C(\zeta_l,\zeta_\theta)$	the 2D Fourier transform of the log-mapped $c(l,\theta)$
P_{FA}	the false alarm rate
P_{FR}	the false rejection rate
$B(n,p)$	binomial distribution to a sequence of n trials with probability p
$N(\mu,\sigma)$	normal distribution with mean μ and standard deviation σ

multimedia content will result in a completely different hash value. This renders cryptographic hash functions not applicable to multimedia object. As opposed to hash functions for the binary messages (or documents), the multimedia fingerprinting function should allow for some modification of the content. To supplement these deficiencies of cryptographic hash functions we arrive at the notion of multimedia fingerprinting, sometimes referred to as robust hash functions [19,43].

A number of applications of multimedia fingerprinting have been considered [24].

- *Filtering technology for file sharing*: Filtering refers to active intervention in content distribution. The prime example is Napster. To settle legal dispute with the music industry Napster introduced the audio filtering system, which restricts copyrighted songs from being downloaded. The demand for filtering mechanism is growing as more people exchange copyrighted images and video through file sharing services. Multimedia fingerprinting is considered as a good candidate for such a filtering mechanism [19,32,36]. Besides copyright protection, the filtering scheme can be used to provide more refined file sharing service: a service that provides free material, different kinds of premium material (accessible to those with a proper subscription) and forbidden material.
- *Broadcast monitoring*: Monitoring refers to tracking of radio, television or web broadcasts for, among others, the purposes of royalty collection, program verification and people metering [19,15,32]. This application is passive in the sense that it has no direct influence on what is being broadcast: the main purpose of the application is to observe and report. A broadcast monitoring system based on fingerprinting consists of several monitoring sites and a central site where the fingerprint server is located. At the monitoring sites fingerprints are extracted from all the (local) broadcast channels. The central site collects the fingerprints from the monitoring sites. Subsequently the fingerprint server, containing a huge fingerprint database, produces the playlists of the respective broadcast channel.
- *Automated indexing of multimedia library*: Nowadays many computer users have a multimedia library containing several hundreds, sometimes even thousands, of multimedia files (song, image and video clips). When the files are obtained from different sources, such as ripping from a music CD, scanning of image and downloading from file sharing services, these libraries are often not well organized. By identifying these files with fingerprinting the files can be automatically labeled with the correct metadata, allowing easy organization based on, for example, artist, music album or genre [18,29,30,42].
- *Connected content*: Connected content is a general term for consumer applications where the content is somehow connected to additional and supporting information. Audio recognition over mobile phone [19,41] is the typical example. Imagine the following situation. You are in your car, listening to the radio and suddenly you hear a song that catches your attention. It is the best new song you have heard for a long time. However you missed the announcement and you do not recognize the

artist. Still, you would like to know more about this music. For this case, you could connect the audio fingerprinting server through your mobile phone to get the title of the music and the name of the artist [19]. It goes without saying that similar applications can be defined for other types of content.

- *Authentication*: Powerful and easy multimedia manipulation software has made it possible to alter digital contents. Authentication verifies the originality of the contents by detecting malicious manipulations. Regarding multimedia authentication, the key issue is to protect the message conveyed by the content not the particular representation of the content [33,37]. Content authentication methods can be classified into fingerprint-based [9,17,37] and watermark-based [12,46] approaches. In general the fingerprint-based approaches are more robust than watermark-based ones. However the fingerprint-based approaches need additional communication channel or storage to check the authenticity of the content; the watermark-based ones do not need these.

The mentioned applications have boosted the interest in multimedia fingerprinting and also a number of image fingerprinting methods have been proposed so far. Schneider and Chang proposed a method that uses the intensity histogram of each image block as a fingerprint to verify authenticity of an image [37]. Venkatesan et al. proposed a method based on randomized processing [43]. Fridrich proposed a method that projects each image block by a random pattern and thresholds it to get fingerprint bits [16]. Lefebvre et al. proposed a fingerprinting method based on the Radon transform [25]. Also patents [13,20] and other papers [1,11,28,44] have been published on this topic. Already a number of companies [2–4,7] have realized the business potential of multimedia fingerprinting. Other signs of a growing awareness of the potential of fingerprinting are a European project on audio recognition [35], the announcement of cooperation between a file-sharing company and content recognition company [36] and a Call for Information by the IFPI and the RIAA [22].

In this paper, we propose the affine transformation resilient image fingerprinting method using the Radon transform [23]. The Radon transform has some useful properties to achieve affine resilience and has proved to be robust against many image processing steps such as sharpening, blurring and compression. Most of the previous methods show limited robustness [16,28,37,43] or require search [25] for the affine transformations. The proposed method is highly robust against affine transformations without searching for the original orientation and achieves collision-free property with relatively small amount of fingerprint bits (400 bits per image). Clearly these will increase the practical use of the proposed method.

This paper is organized as follows. Section 2 presents the definition and requirements of cryptographic hashing and multimedia fingerprinting. Section 3 describes the proposed affine invariant fingerprinting system. Section 4 evaluates the performance of the proposed method.

2. Overview of multimedia fingerprinting

2.1. Cryptograhic hashing vs. multimedia fingerprinting

A cryptographic hash function $H(X)$ maps an (usually large) object X to a (usually small) hash value. It allows comparing two large objects X and Y, by only comparing their respective hash values $H(X)$ and $H(Y)$. *Mathematical equality* of $H(X)$ and $H(Y)$ implies the equality of X and Y with only a very low probability of error. For a properly designed cryptographic hash function this should be 2^{-L}, where L equals the number of bits in the hash value. Using cryptographic hash functions, an efficient method exists to check whether or not a particular data item X is contained in a given and large data set $Y = \{Y_i\}$. Instead of storing and comparing with all of the data in Y, it is sufficient to store the set of hash values $\{h_i = H(Y_i)\}$, and to compare $H(X)$ with this set of hash values. This method is more efficient because the storage requirements are usually more relaxed and fewer bits have to be compared. The only caveat is that an initial pre-computation is required to compute the hash values $\{h_i\}$. We note that good cryptographic hash functions do indeed exist. The desirable properties of cryptographic hash functions $H(\ )$ are given as follows [5]:

- *One-way hash function*: Given $H(X)$, it is hard to find an original object X, and given X and $H(X)$, it is hard to find an object $X'(\neq X)$ such that $H(X') = H(X)$.
- *Collision-free hash function*: It is hard to find two distinct objects X and Y that hash to the same result $(H(X) = H(Y))$.
- *Keyed hash function*: Without knowledge of key K, it is hard to determine $H_K(X)$ for any object X, to find X from $H_K(X)$ and to find two distinct objects X and Y that hash to the same result $(H_K(X) = H_K(Y))$. Given (possibly many) pairs of objects X_i and their hash values $H(X_i)$, it is hard to find the secret key K.

Given the above arguments, one might suppose that cryptographic hash functions are a good tool to identify multimedia content: take a hash function, store hash values for all available contents in a large database (costly, but it needs to be done only once) and identify content by hash matching. This method will however fail when using classical cryptographic hash functions. For a multimedia content we are not interested in mathematical equality, but perceptual equality since it often undergoes various quality-preserving manipulations during distribution, which include compression, enhancement, geometrical distortions and analog-to-digital conversion. Cryptographic hash functions cannot identify the processed images as the original one. This problem could be mitigated if cryptographic hash functions would have a *continuous behavior*, i.e. if perceptually similar content would at least result in mathematically similar hash values. However, cryptographic hash functions typically have rather the opposite property, in the sense that they are *bit sensitive*: a single bit of difference in the content will result in a completely different hash value.

From these observations we conclude that for multimedia identification we need multimedia fingerprinting (*perceptual hashing*) functions, functions that (i) map large multimedia objects to a small number of bits, and (ii) map perceptually similar objects to (mathematically) similar fingerprint values [24]. More precisely, for a properly designed fingerprint function F, there should be a threshold T such that $|F(A) - F(B)| < T$ if multimedia objects A and B are similar and with high probability $|F(A) - F(B)| > T$ if they are dissimilar [19]. An observing reader might wonder why instead of (ii) we do not require that perceptually similar objects have mathematically equal fingerprint values. The question is valid, but the answer is that such a modeling of perceptual similarity is not possible for reasons of transitivity. To be more precise, it is a known fact that perceptual similarity is not transitive. Perceptual similarity of a pair of objects A and B and of another pair of objects B and C does not necessarily imply the perceptual similarity of objects A and C. However, modeling perceptual similarity by equality of fingerprinting functions would lead to a transitive relationship.

2.2. Multimedia fingerprinting concept

Content recognition by comparing features can be categorized in two main classes; the class of methods based on the semantic and the non-semantic features [32]. The former class consists of extracting and representing content by semantically meaningful features. Examples of such features are scene boundaries and color-histograms. The latter class uses features that have no direct semantic interpretation but are nonetheless robust with respect to content quality preserving transformations. Both features can be used to establish perceptual equality of an image. However, it should be noted that a feature extraction method for fingerprinting must be quite different from the methods used for retrieval. In retrieval, the features must facilitate searching of images that somehow look similar to the query, of that contain similar objects as the query. In fingerprinting the requirement is to identify images that are perceptually the same, except for quality differences or the effects of other signal processing. Therefore, the features for fingerprinting need to be far more discriminatory, but they do not necessarily need to be semantic [32]. Moreover, there is no effective method yet to automatically generate good semantic features for an image [21]. Because of those reasons, the non-semantic robust features are widely used in building fingerprints.

Constructing a fingerprinting function is not a trivial task. Given the lack of a proper audio–visual perception model (or perceptual metric) and an exact discrimination criterion, it is an ill-posed problem to start with. For example, humans do not readily perceive a perceptual difference between an original image and a moderately resized or compressed version. In general, the fingerprinting function needs to have the following properties.

- *Robustness* (Invariance under perceptual similarity): The fingerprints resulting from degraded versions of an image should result in the same or at least similar fingerprints with respect to the fingerprint of the original image. Robustness refers to the ability to positively identify two perceptually similar objects as similar.
- *Pairwise independence* (Collision free): If two images are perceptually different, the fingerprints from two images should be considerably different. For pairwise independence, it is desirable that fingerprint bits have uniform distribution and are uncorrelated with each other.
- *Database search efficiency*: For the commercial applications mentioned in Section 1, fast database search is essential. Through a naive approach, every identification of an image would approximately require more than millions of comparisons for a large database (for example, millions of images). For any reasonable bit-size of the fingerprints, this would mean impractical access times. Without a proper structure in a fingerprint database, searching and retrieving will easily explode into an impractical system [8,10]. However, if the space of fingerprint values has some kind of perceptual ordering structure, search complexity can be considerably reduced [24].

3. Proposed image fingerprinting method

The fingerprinting function should be based on robust and perceptually relevant features to meet the requirements in Section 2.2. Our prime objective is to achieve high robustness against affine transformations without losing other desirable properties. Resilience to affine transformations has been one of the main issues in many image processing research areas, such as pattern recognition [31,45] and watermarking [6,26,34,40]. In order to obtain affine resilience, the affine invariant features based on the Radon transform are selected for the proposed image fingerprints. The Radon transform has proved to be robust against image processing such as sharpening, blurring, adding noise, compression and has some desirable properties with regard to affine transformations. Details of the proposed method are in the next subsections.

3.1. Radon transform and its properties

The Radon transform represents an image as a collection of projections along various directions. It is widely used in areas ranging from seismology to computer vision. The Radon transform of an image $f(x,y)$, denoted as $g(s,\theta)$, is defined as its line integral along a line inclined at an angle θ from the y-axis and at a distance s from the origin [33] as shown in Fig. 1. Mathematically, it is written as

$$g(s,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta + y\sin\theta - s)\,\mathrm{d}x\,\mathrm{d}y, \tag{1}$$

where $-\infty < s < \infty$, $0 \leqslant \theta < \pi$. The Radon transform $g(s,\theta)$ is the one-dimensional projection of $f(x,y)$ at an angle θ. The Radon transform has the following useful properties for the affine transformations of an image.

(P1) Translation of an image by (x_0,y_0) causes the Radon transform to be translated in the direction of s, i.e.,

$$f(x-x_0,y-y_0) \leftrightarrow g(s-x_0\cos\theta - y_0\sin\theta,\theta).$$

(P2) Scaling (retaining aspect ratio) of an image by a factor ρ ($\rho > 0$) causes the Radon transform to be scaled through the same factor, i.e.,

$$f(\rho x,\rho y) \leftrightarrow \frac{1}{|\rho|}g(\rho s,\theta).$$

(P3) Rotation of an image by an angle θ_r causes the Radon transform to be shifted by the same amount, i.e.,

$$f(x\cos\theta_r - y\sin\theta_r, x\sin\theta_r + y\cos\theta_r) \leftrightarrow g(s, \theta - \theta_r).$$

Fig. 2 shows the Radon transform of the Lena image. Fig. 2a–d shows the Radon transform of original, translated (64 pixels in x direction), scaled (scaling factor 0.75) and rotated (25°) Lena image, respectively. Fig. 2b–d shows that $g(s,\theta)$ is translated by $64\cos\theta$ pixels in the s direction, scaled by a factor of 0.75 in the s direction and cyclicly shifted by 25° in the θ direction, respectively, with respect to the Radon transform of the original Lena image.

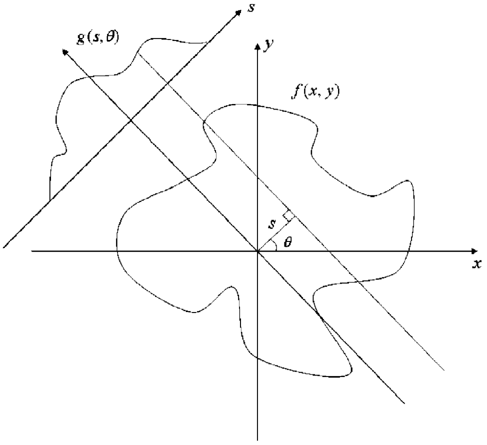

Fig. 1. Projection integral in the direction θ.

3.2. Affine invariant feature extraction

We consider here angle preserving affine transformations: translation, scaling (retaining aspect-ratio) and rotation. By using the above properties of the Radon transform, affine-invariant features are obtained. From (P1) the translation of an image causes translation in the Radon domain, but the amount of translation in each projection is different. For translation invariance, the normalized auto-correlation of each radial projection is calculated that is given as follows:

$$c(l,\theta) = \frac{\int_{-\infty}^{\infty} g(s,\theta)g(s-l,\theta)\,\mathrm{d}s}{\int_{-\infty}^{\infty} g(s,\theta)g(s,\theta)\,\mathrm{d}s}. \tag{2}$$

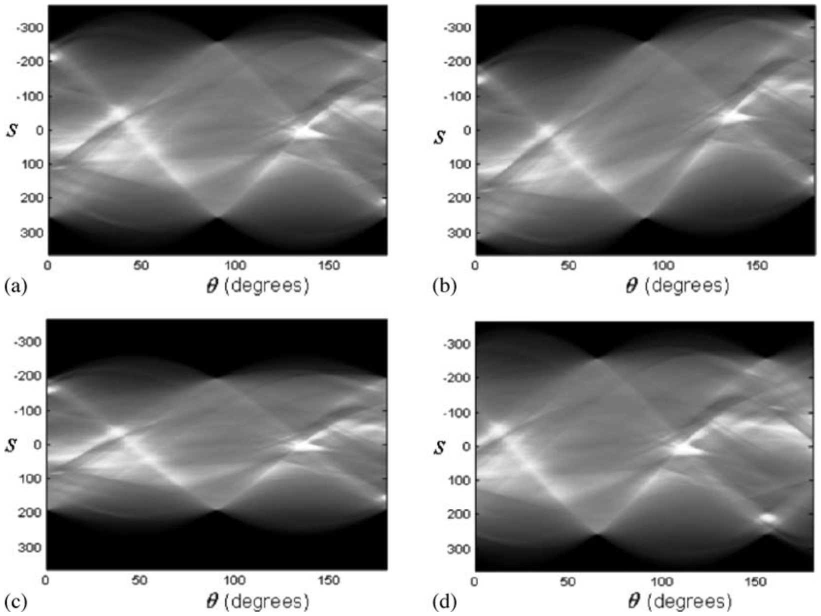

Fig. 2. The Radon transform of: (a) original, (b) translated (64 pixels in x direction), (c) scaled (scaling factor 0.75), (d) rotated (25°), Lena image (size 512 × 512).

By taking the auto-correlation, we get a translation-invariant signal $c(l,\theta)$. Among the affine transformations, scaling and rotation are remained in $c(l,\theta)$. Consider the auto-correlation $c(l,\theta)$ of the Radon transform of an original image. From (P2) and (P3), the auto-correlation of the Radon transform of a scaled and rotated image is given as $c'(l,\theta) = c(\rho l, \theta - \theta_r)$ where ρ ($\rho > 0$) and θ_r are the amount of scaling and rotation, respectively. To achieve invariance on the scaling and rotation, the log mapping and the 2D Fourier transform are used. The log mapping translates the scaling of the signal to a shift. The subsequent Fourier transform translates this shift into a phase change. By the log mapping $l = \mathrm{e}^{\mu}$, the signal $c'(l,\theta)$ can be written as

$$\begin{aligned} c'(l,\theta) &= c(\rho l, \theta - \theta_r) \\ &= c(\exp[\mu + \log \rho], \theta - \theta_r). \end{aligned} \tag{3}$$

Then the log-mapped signal $\tilde{c}'(\mu,\theta)$ is given by

$$\tilde{c}'(\mu,\theta) = \tilde{c}(\mu + \log\rho, \theta - \theta_r). \tag{4}$$

The 2D Fourier transform of the log-mapped signal is written as

$$\begin{aligned} C'(\zeta_l,\zeta_\theta) &= \int_0^{\pi}\int_{-\infty}^{\infty} \tilde{c}'(\mu,\theta)\exp[-\mathrm{j}\mu\zeta_l - \mathrm{j}\theta\zeta_\theta]\,\mathrm{d}\mu\,\mathrm{d}\theta \\ &= \exp[\mathrm{j}\zeta_l \log\rho - \mathrm{j}\zeta_\theta\theta_r] C(\zeta_l,\zeta_\theta). \end{aligned} \tag{5}$$

Then the magnitude $|C'(\zeta_l,\zeta_\theta)|$ and phase $\phi'(\zeta_l,\zeta_\theta)$ of the complex signal $C'(\zeta_l,\zeta_\theta)$ are given by

$$|C'(\zeta_l,\zeta_\theta)| = |C(\zeta_l,\zeta_\theta)| \tag{6}$$

$$\begin{aligned} \phi'(\zeta_l,\zeta_\theta) &= \zeta_l \log\rho - \zeta_\theta\theta_r + \angle C(\zeta_l,\zeta_\theta) \\ &= \zeta_l \log\rho - \zeta_\theta\theta_r + \phi(\zeta_l,\zeta_\theta), \end{aligned} \tag{7}$$

where $\angle C(\zeta_l,\zeta_\theta)$ is the phase of the complex signal $C(\zeta_l,\zeta_\theta)$.

As shown above, the log mapping translates scaling into a shift, and the subsequent Fourier transform translates the shift into a phase change. By using these properties, we find features that are invariant to scaling. From Eq. (6), $|C'(\zeta_l,\zeta_\theta)|$ is affine invariant. Since $\zeta_l \log\rho - \zeta_\theta\theta_r$ in Eq. (7) is a linear function of ζ_l and ζ_θ, the double differentiation of $\phi'(\zeta_l,\zeta_\theta)$ on ζ_l or ζ_θ is also affine invariant.

3.3. *Fingerprint bit extraction*

An overview of the proposed fingerprinting method is shown in Fig. 3. To obtain affine resilience, the affine invariant features described in Section 3.2 are used for fingerprint bit extraction. In practice, the energy compaction property of the DFT (Discrete Fourier Transform) ensures that most of the energy of the image is concentrated at low-frequency coefficients of $C'(\zeta_l,\zeta_\theta)$. We note that a large majority of the Fourier coefficients has relatively small values and therefore contribute little to the total energy of the signal. From the energy compaction property, we take the 21×21 low-frequency coefficients of $C'(\zeta_l,\zeta_\theta)$ for the filtering as shown in Fig. 4. In general, the low-frequency coefficients are more robust, and the high-frequency coefficients are more discriminatory. Therefore we should make a trade-off between robust and discriminatory features in choosing the coefficients. Fingerprints from the 21×21 low-frequency coefficients showed the best balance between the two. Experimentally it was verified that the sign of the difference between affine invariant features is very robust against many kinds of processing and reduces correlation between fingerprint bits. The coefficients of $\log|C'(\zeta_l,\zeta_\theta)|$ and $\phi'(\zeta_l,\zeta_\theta)$ are filtered by a simple 2D filter F_{ζ_l,ζ_θ} (along both ζ_l and ζ_θ axes), of which the kernel F_{ζ_l,ζ_θ} equals

$$F_{\zeta_l,\zeta_\theta} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{8}$$

The output of the filter F_{ζ_l,ζ_θ} is invariant to affine transformations as we have seen in Section 3.2. If a keyed fingerprinting function is required, we can interleave $|C'(\zeta_l,\zeta_\theta)|$ and $\phi'(\zeta_l,\zeta_\theta)$ before filtering as shown in Fig. 3. The details of the

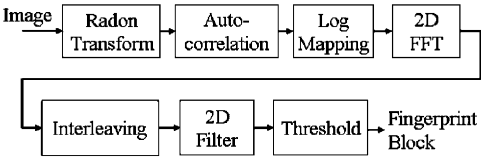

Fig. 3. Overview of affine transformation resilient fingerprint extraction.

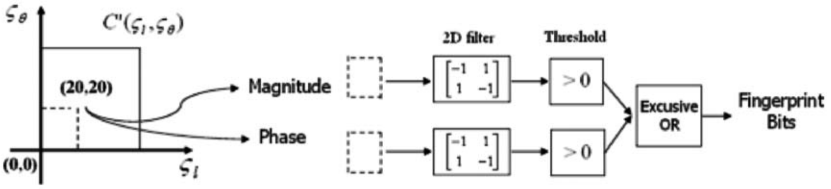

Fig. 4. Fingerprint bit extraction.

keyed fingerprinting function are described in Section 3.5. Finally the 20 × 20 filter output (only those parts of the filter output that are computed without the zero padding) is converted to bits by taking the sign of the resulting value (thresholding). Then we get two intermediate fingerprints from the magnitude $|C'(\zeta_l, \zeta_\theta)|$ and the phase $\phi'(\zeta_l, \zeta_\theta)$. The fingerprint bits are determined by taking exclusive-or (XOR) of the bits from the magnitude and the bits from the phase. It was experimentally verified that mixing the magnitude and the phase information by XOR improves the pairwise independence (collision-free). Finally we obtain 20 × 20 fingerprint bits (400 bits per image). We refer to the 20 × 20 fingerprint bits as a *fingerprint block* and the 20 bits in each row as a *sub-fingerprint*. This ordering structure can reduce the database search complexity considerably [19]. The experimental database search result is given in Section 4.2.

Fig. 5 shows an example of a fingerprint block (20 subsequent 20-bit sub-fingerprints) extracted with the proposed method from the Lena image. A ‘1’ bit corresponds to a white pixel and a ‘0’ bit to a black pixel. Fig. 5a and b shows a fingerprint block from an original image and JPEG compressed (quality factor 10%) version of it, respectively. Ideally these two fingerprints should be identical, but due to the compression some of the bits are erroneous. These bit errors, which are used as *the similarity measure*, are shown in black in Fig. 5c.

3.4. Fingerprint matching

For the fingerprint matching, the images are declared similar if the Hamming distance (bit error rate) between their fingerprints is below a certain threshold T. The problem could be formulated as the following hypothesis testing using the fingerprinting function $H(\ )$:

- L_0: Two images I and I' are from the same image if the Hamming distance $D(H(I), H(I'))$ is below a threshold T.
- L_1: Two images I and I' are from the different image if the Hamming distance $D(H(I), H(I'))$ is above a threshold T.

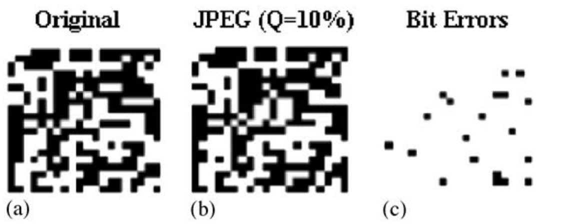

Fig. 5. (a) Fingerprint of original Lena image, (b) Fingerprint of compressed Lena image, (c) the difference between a and b showing bit errors in black (BER = 0.05).

For the selection of threshold T, the false alarm rate P_{FA} and the false rejection rate P_{FR} should be considered. The false alarm rate P_{FA} is the probability to declare different images as *similar*. The false rejection rate P_{FR} is the probability to declare the images from the same image as *dissimilar*. In practice, P_{FR} is difficult to analyze since there are plenty of image processing steps of which the exact characteristics are not known. Thus it is common to deal with P_{FA} for the selection of threshold T.

In order to analyze the choice of threshold T, we assume that the fingerprint extraction process yields random independent and identically distributed (i.i.d.) bits. Then the number of bit errors between the fingerprints from different images will have a binomial distribution $B(n,\ p)$ where n equals the number of bits extracted and p is the probability that a '0' or '1' bit is extracted. Since $n\ (= 400)$ is sufficiently large, the binomial distribution can be approximated by a normal distribution with mean np and standard deviation $\sqrt{np(1-p)}$. From that, the bit error rate (BER) has a normal distribution with mean $\mu = p$ and standard deviation $\sigma = \sqrt{p(1-p)/n}$. For the ideal case, $p = 0.5$ and thus $\mu = 0.5$ and $\sigma = 0.025$. Through the normal approximation $N(\mu,\ \sigma)$, the false alarm rate P_{FA} for BER is given as follows:

$$P_{\mathrm{FA}} = \int_{-\infty}^{T} \frac{1}{\sqrt{2\pi}\sigma} \exp\left[\frac{-(x-\mu)^2}{2\sigma^2}\right] \mathrm{d}x = \frac{1}{2}\operatorname{erfc}\left(\frac{\mu - T}{\sqrt{2}\sigma}\right). \tag{9}$$

For a certain value of P_{FA}, the threshold T for the BER can be determined. In the experiments we use $T = 0.3$. Then we arrive at a very low false alarm probability of $\operatorname{erfc}(5.66)/2 = 6.2E - 16$. It means that out of 400 bits there must be less than 120 bits in error in order to decide that the fingerprint blocks originate from the same image. It was experimentally verified in Section 4.1 that the fingerprints extracted using the proposed method follow the random i.i.d. case fairly well.

3.5. Keyed fingerprinting function

In some applications (for example image authentication), the security of the fingerprint extraction algorithm is an issue. More precisely, it is sometimes required that the fingerprint function depends on a key K. For two different keys K_1 and K_2, the fingerprinting function H should have the property that $H_{K_1}(X) \neq H_{K_2}(X)$ for any image X. Some general guidelines for keyed cryptographic hash functions are given at [5]. First, it should use the secret key and every bit of the object iteratively in the hashing process. Second, it should uniformly distribute the hash bits to thwart statistical attacks. These guidelines should be also considered in constructing keyed fingerprinting function. We construct a keyed fingerprinting function by using the interleaving as shown in Fig. 3. The coefficients of $C'(\zeta_l, \zeta_\theta)$ are randomly interleaved in either ζ_l or ζ_θ direction by the permutation table (this is key information). The construction of the permutation table as a function of key can be found in [38]. After interleaving, the same fingerprint extraction method in Section 3.3 is used. Interleaving does not have any effect on the affine invariance of the output of the filter F_{ζ_l,ζ_θ}. To get more security, a nested structure shown in Fig. 6 can be used. The fundamental building block in the nested structure is the fingerprint extraction function and the XOR operation. This is called a *round*. In each round, fingerprint bits are extracted from the image with a different key. Then, the fingerprint bits in the current round are XORed by the fingerprints in the previous rounds. Mathematically it is given as follows:

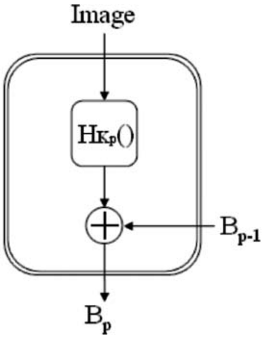

Fig. 6. Nested structure for keyed fingerprinting.

$$B_p = B_{p-1} \oplus H_{K_p}(X), \tag{10}$$

where X is the input image, $\oplus$ is the XOR, B_p is pth round fingerprints, K_p is pth round permutation table and $p = 1, \ldots, P$ (P is constant positive integer). The process is continued until Pth round, and then the final fingerprint bits are B_P.

4. Experimental results

To evaluate the proposed method, we tested our method on a thousand images that include indoor and outdoor scenes, people, vehicles, sporting events and paintings. Some of the images were taken by digital camera, and others were gathered from the Internet. The sizes of the images range from 133 × 209 to 2560 × 1920. Prior to applying the proposed method, we normalize the image by taking the luminance component of it and resizing it uniformly to either 512 × M or M × 512 where M is smaller than 512. The resized image is filtered by median filter that is somehow effective in correcting small geometric processing, such as bend, distort and stretch. The preprocessed image is projected onto N (typically, $N = 512$) radial directions using the Radon transform. The other steps are the same as in Fig. 3.

4.1. Pairwise independence and key dependence

To test pairwise independence, we extracted fingerprints from 1000 images. Thereafter the BER between all possible pairs (499,500) of the fingerprints were calculated. Fig. 7a shows the histogram of the measured BER. All the measured BER were in the range between 0.37 and 0.62. The histogram of the measured BER shows that the proposed fingerprints follow the ideal random i.i.d. case in Section 3.4 fairly well. The mean of the measured BER was 0.4930 which is close to 0.5, and the standard deviation of it was 0.0272 which is also close to 0.025. This shows that the proposed method is approximately pairwise independent.

To show validity of the interleaving as a key, we generated 1000 fingerprints from the Lena image using different interleaving. Similar with the above analysis, the BER between all possible pairs of the fingerprints were calculated. The histogram of the measured BER is shown in Fig. 7b. Mean and standard deviation of the measured BER were 0.5000 and 0.0263, respectively. This result clearly shows the fingerprint is significantly dependent on the key information (interleaving). By combining the nested structure in Fig. 6 with interleaving as in Section 3.5, the proposed method can provide efficient keying scheme. In terms of security, such a strong dependency on the key is significant. Once a key is broken, the user can simply change it, like a password [43] without modifying overall system.

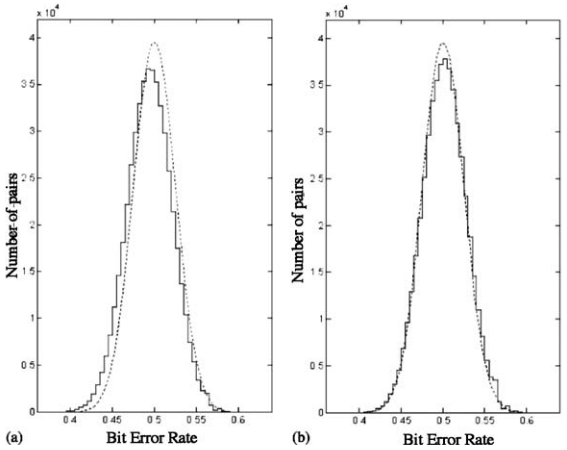

Fig. 7. (a) Histogram of measured BER between the fingerprints from different images, (b) Histogram of measured BER between fingerprints of Lena generated with different keys; The dotted line represents the ideal random i.i.d. case $N(0.5, 0.025)$.

4.2. Robustness and database search efficiency

To test robustness of the proposed method, the original images were subjected to various image processing steps (see [14] for a detailed description of the processing steps) and their respective fingerprint blocks were extracted. Mean and standard deviation and false rejection rate of the BER between the original and the processed image fingerprints are shown in Table 1 for 1000 images. Figs. 8 and 9 show histograms of the measured BER between the original and the processed image fingerprints for the geometric and the non-geometric image processing steps respectively.

Table 1
Mean, standard deviation (Std) and false rejection rate (with threshold $T = 0.3$) of the measured BER for different kinds of signal degradations for 1000 test images

Processing	Mean	Std	P_{FR}
JPEG ($Q = 10\%$)	0.0487	0.0350	0.002
Gaussian filtering	0.0206	0.0175	0
Sharpening filtering	0.0458	0.0317	0
Median filtering (4 × 4)	0.0519	0.0277	0
Rotation (worst case 45.176°)	0.1740	0.0275	0.001
Rotation (90°)	0.0806	0.0185	0
Scaling ($\rho = 0.5$)	0.0202	0.0184	0
Scaling ($\rho = 0.15$)	0.1146	0.0690	0.023
Cropping (2%)	0.1483	0.0524	0.007
17 column 5 row removed	0.1671	0.0670	0.040
Random bending attack	0.1928	0.0629	0.049

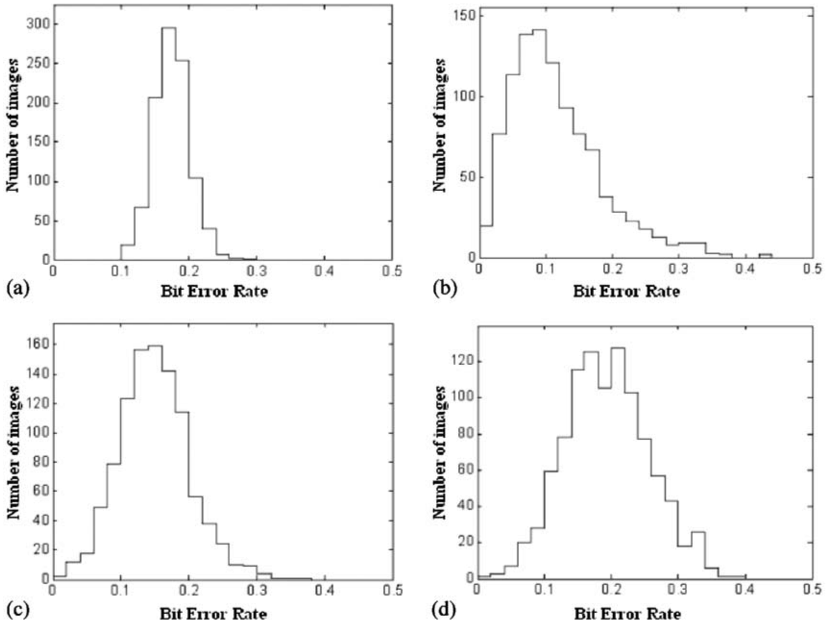

Fig. 8. Histogram of measured BER for: (a) rotation, (45.176°) (b) scaling ($\rho = 0.15$), (c) cropping (2%), (d) random bending attack.

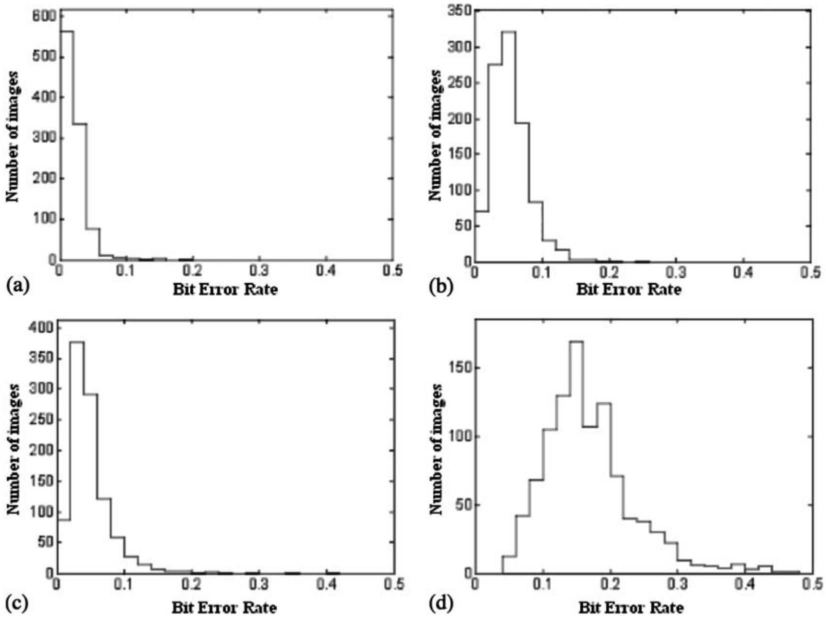

Fig. 9. Histogram of measured BER for: (a) Gaussian filtering, (b) median filtering, (c) JPEG compression, ($Q = 10\%$) (d) 17 column 5 row removed.

The result denotes that the proposed method is highly robust to affine transformations, which preserve aspect ratio (all possible angles of the rotation and the scaling factor ρ larger than 0.15) and other image processing steps including compression and various filtering.

It is important for any fingerprinting method that it not only results in a low BER but also allows efficient searching. For the efficient searching we made an ordering structure; the fingerprint block and the sub-fingerprint as described in Section 3.3. Rather than searching the database with fingerprint block every time, sub-fingerprint matching using a lookup table is more efficient [19]. The simplest assumption is that at least one sub-fingerprint has an exact match at the database. Then the only fingerprints in the database that need to be checked are the ones where one of the 20 sub-fingerprints in the fingerprint block matches perfectly. However, for heavily degraded images the simplest assumption might not be always true. All the 20 sub-fingerprints might have several bit errors. In [19] a search algorithm is presented that exploits the fact that each of the 20 sub-fingerprints in a fingerprint block has a list of most probable candidates for being an *original* sub-fingerprint. For the proposed method, the sub-fingerprints are obtained by comparing and thresholding the difference of affine invariant features. If the difference is very close to the threshold (in our case the threshold is zero), it will be more likely that the bit was received incorrectly. If the difference is much larger than the threshold, the probability of an incorrect bit will be low. From this soft-decoding information, a list of most probable candidates for being an original sub-fingerprint is determined. If we want to allow d errors in each sub-fingerprint, the number of candidates will be 2^d. In our experiments, a list of 1024 most probable candidates ($d = 10$) was created for each sub-fingerprint. Then we have 20 lists of 1024 candidates from 20 sub-fingerprints in the fingerprint block. From these lists the fingerprint database can be searched very efficiently, and with high probability at least one of the 20 lists of the fingerprint block contains a corresponding *original* sub-fingerprint. There are two measures in assessing the fingerprint database search: recall and precision. Recall refers to the rate at which the database search contains the corresponding fingerprint. Precision refers to the proportion of the database search that is actually correct. To test the recall rate of the database search, we search the database with the processed images. Table 2 shows the number of lists that contain a corresponding original sub-fingerprint for each processing step. This number is referred to as the number of database hits. Table 2 shows that the number of database hits is sufficient to database search for most of the image processing steps. We recall from [19] that only a single database hit is needed for a successful search. To test the precision of the database search, we search the database with the 1000 original test images that are used for making the database. Then it is desirable that the database search reports only one candidate image in terms of precision. Fig. 10 shows the histogram of the number of candidate images which the database search reported. The average number of candidate images was 2.396 and 51.451 for $d = 0$ and $d = 10$ cases, respectively. We note that the precision is the reciprocal of the number of candidate images. Table 2 and Fig. 10 show that there is a trade-off between the recall and

Table 2
Mean of the number of hits in the database for different kinds of signal degradations for 1000 test images

Processing	1 candidate case ($d = 0$)	1024 candidates case ($d = 10$)
JPEG ($Q = 10\%$)	9.702 (5)	17.715 (1)
Gaussian filtering	14.018 (1)	19.353 (0)
Sharpening filtering	9.861 (4)	17.921 (0)
Median filtering (4×4)	8.661 (2)	17.631 (0)
Rotation (worst case 45.176°)	1.490 (228)	9.140 (1)
Rotation (90°)	6.622 (0)	15.077 (0)
Scaling ($\rho = 0.5$)	14.271 (1)	19.364 (0)
Scaling ($\rho = 0.15$)	5.059 (54)	13.356 (2)
Cropping (2%)	3.123 (96)	10.825 (0)
17 column 5 row removed	1.684 (291)	9.384 (12)
Random bending attack	2.159 (196)	8.463 (4)

The number in the parentheses refers to the number of images without hits.

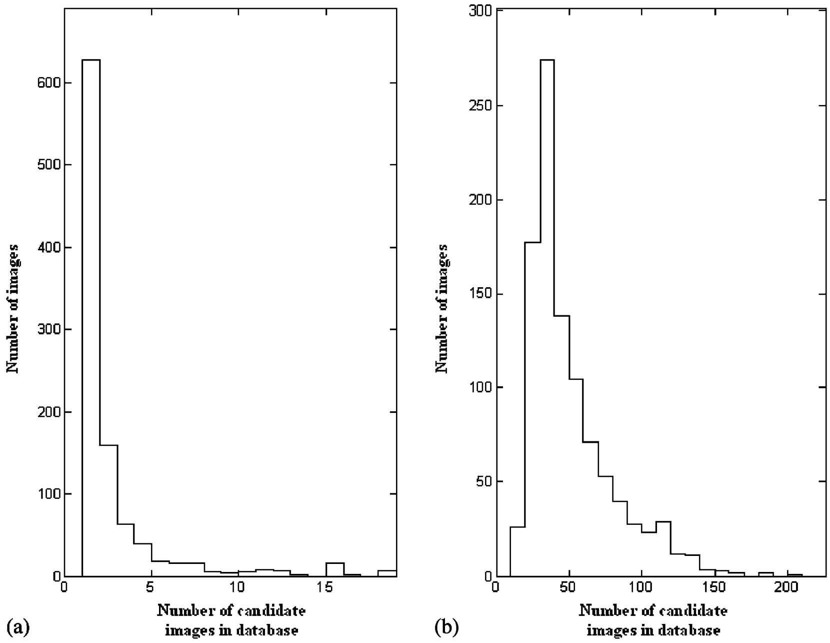

Fig. 10. Histogram of the number of candidate images which the database search reported for 1000 test images, (a) 1 candidate for each sub-fingerprint ($d = 0$ case), (b) 1024 candidates for each sub-fingerprint ($d = 10$ case).

the precision of the database search with the value of d.

5. Conclusion

For multimedia fingerprinting, extracting features that allow direct access to the relevant distinguishing information is crucial. For a good fingerprinting system, the features should be both fairly discriminative and robust. In this paper, we proposed a robust image fingerprinting method by basing on the affine invariant differential features of the Radon transform. Fingerprint bits are obtained using the nonlinear operation (taking the sign after thresholding) and random permutation of the affine invariant features. It was experimentally verified that the proposed image fingerprints satisfy the main requirements of fingerprints; robustness under quality preserving signal processing steps, pairwise independence with different inputs and database search efficiency. A variant of the proposed fingerprint scheme has also been successfully used for extracting speed-change resilient audio fingerprints [39]. Future work includes the extension of the proposed method to video.

Acknowledgements

Jin S. Seo and Chang D. Yoo were supported by Grant No. R01-2003-000-10829-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

References

[1] M. Abdel-Mottaleb, G. Vaithilingam, S. Krishnamachari, Signature-based image identification, in: Proceedings of the SPIE 3845, Multimedia Systems and Applications II, Boston, 1999.

[2] ACNielsen, http://www.acnielsen.com/.

[3] Arbitron, http://www.arbitron.com/.

[4] AudibleMagic, http://www.audiblemagic.com/.

[5] S. Bakhtiari, R. Safavi-Naini, J. Pierzyk, Cryptographic hash functions: a survey, Department of Computer

Science, University of Wollongong, Technical Report 95-09, July 1995.

[6] S. Baudry, P. Nguyen, H. Maitre, Optimal decoding for watermarks subject to geometrical attacks, Signal Processing: Image Communication 18 (4) (April 2003) 297–307.

[7] BDSOnline, http://www.bdsonline.com/.

[8] S. Berchtold, C. Bohm and H. Kriegel, ‘‘The pyramid technique: towards breaking the curse of dimensionality, in: Proceedings of the ACM SIGMOD 98, Seattle, WA, June 1998.

[9] S. Bhattacharjee, M. Kutter, Compression tolerant image authentication, in: Proceedings of the IEEE ICIP 1998, Chicago, IL, October 1998.

[10] K. Chakrabarti, S. Mehrotra, High dimensional feature indexing using hybrid trees, in: Proceedings of the International Conference on Data Engineering, Sydney, Australia, March 1999.

[11] S.S. Cheung, A. Zakhor, Video similarity detection with video signature clustering, in: Proceedings of the IEEE ICIP 2001, Thessaloniki, Greece, 2001.

[12] J. Eggers, B. Girod, Blind watermarking applied to image authentication, in: Proceedings of the IEEE ICASSP 2001, Saltlake city, UT, May 2001.

[13] M.D. Ellis, S.M. Dunn, M.W. Fellinger, F.B. Younglove, D.M. James, D.L. Clifton, R.S. Land, Method and apparatus for producing a signature characterizing an interval of a video signal while compensating for picture edge shift, US Patent 5572246, November 1995.

[14] F.A.P. Fetitcolas, Watermarking schemes evaluation, IEEE Signal Process. 17 (5) (September 2000) 58–64.

[15] D. Fragoulis, G. Rousopoulos, T. Panagopoulos, C. Alexiou, C. Papaodysseus, On the automated recognition of seriously distorted musical recordings, IEEE Trans. Signal Process. 49 (4) (April 2001) 536–540.

[16] J. Fridrich, Robust bit extraction from images, in Proceedings of the IEEE International Conference on Multimedia Computing and Systems (ICMCS), Florence, Italy, June 1999.

[17] G.L. Friedman, The trustworthy digital camera: restoring credibility to the photographic image, IEEE Trans. Consumer Electron. 39 (4) (November 1993) 905–910.

[18] Gracenote, http://www.gracenote.com/.

[19] J.A. Haitsma, T. Kalker, A highly robust audio fingerprinting system, in: Proceedings of the International Conference on Music Information Retrieval (ISMIR) 2002, Paris, October 2002.

[20] A. Hershtik, Monitoring system, Patent Application WO 070869, May 2000.

[21] J. Huang, S.R. Kumar, R. Zabih, An automatic hierarchical image classification scheme, in: Proceedings of the ACM Conference on Multimedia, Bristol, England, September 1998.

[22] IFPI, Call for information on fingerprinting technology, http://www.ifpi.org/.

[23] A.K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[24] T. Kalker, J.A. Haitsma, J. Oostveen, Issues with digital watermarking and perceptual hashing, in: Proceedings of the SPIE 4518, Multimedia Systems and Applications IV, November 2001, 189–197.

[25] F. Lefebvre, B. Macq, J.-D. Legat, RASH: Radon soft hash algorithm, in: Proceedings of the European Signal Processing Conference 2002, Toulouse, France, September 2002.

[26] C-Y. Lin, M. Wu, J.A. Bloom, M.L. Miller, I.J. Cox, Y-M. Lui, Rotation, scale, and translation resilient public watermarking for images, IEEE Trans. Image Process. 10 (5) (May 2001), 767–782.

[27] A. Menezes, P. Oorshot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, FL, 1997.

[28] M.K. Mihcak, R. Venkatesan, New iterative geometric methods for robust perceptual image hashing, in: Proceedings of the ACM Workshop on Security and Privacy in Digital Rights Management, Philadelphia, PA, November 2001.

[29] MusicBrainz, http://www.musicbrainz.org/.

[30] NISO, http://www.niso.org/.

[31] J.-R. Ohm, F. Bunjamin, W. Liebsch, B. Makai, K. Muller, A. Smolic, D. Zier, A set of visual feature descriptors and their combination in a low-level description scheme, Signal Processing: Image Communication 16 (1–2) (September 2000), 157–179.

[32] J. Oostveen, T. Kalker, J.A. Haitsma, Feature extraction and a database strategy for video fingerprinting, in: Proceedings of the International Conference on Visual Information Systems, HsinChu, Taiwan, March 2002.

[33] M.P. Queluz, Authentication of digital images and video: generic models and a new contribution, Signal Processing: Image Communication 16 (5) (January 2001), 461–475.

[34] J. Ruanaidh, T. Pun, Rotation, scale and translation invariant spread spectrum digital image watermarking, Signal Processing 66 (3) (May 1998).

[35] RAA, Recognition an analysis of audio, http://raa.joanneum.ac.at/.

[36] Relatable, http://www.relatable.com/.

[37] M. Schneider, S.-F. Chang, A robust content based digital signature for image authentication, in: Proceedings of the IEEE ICIP 96, Laussane, Switzerland, October 1996.

[38] R. Sedgewick, Permutation generation methods, ACM Comput. Surv. 9 (2) (June 1977), 137–164.

[39] J.S. Seo, J.A. Haitsma, T. Kalker, Linear speed-change resilient audio fingerprinting, in: Proceedings of the IEEE Benelux Workshop on Model Based Processing and Coding of Audio (MPCA) 2002, Leuven, Belgium, November 2002.

[40] V. Solachidis, I. Pitas, Circularly symmetric watermark embedding in 2-D DFT domain, IEEE Trans. Image Process. 10 (11) (November 2001).

[41] Shazam, http://www.shazamentertainment.com/.

[42] TASI, http://www.tasi.ac.uk/advice/delivering/meta.html/.

[43] R. Venkatesan, S.-M. Koon, M.H. Jakubowski, P. Moulin, Robust image hashing, in: Proceedings of the IEEE ICIP 2000, Vancouver, Canada, September 2000.

[44] H. Wang, F. Guo, D. Feng, J. Jin, A signature for content-based image retrieval using a geometrical transform, in: Proceedings of the Multimedia and Security Workshop at ACM Multimedia, Bristol, UK, 1998, 229–234.

[45] J. Wood, Invariant pattern recognition: a review, Pattern Recognition 29 (1) (January 1996).

[46] M. Wu, B. Liu, Watermarking for image authentication, in: Proceedings of the IEEE ICIP 1998, Chicago, IL, October 1998.