On security threats for robust perceptual hashing

O. Koval[a], S. Voloshynovskiy[a], P. Bas[b] and F. Cayre[b]

[a] Computer Vision and Multimedia Laboratory, University of Geneva,
7 route de Drize, 1227 Carouge 4, Switzerland

[b] Gipsa-lab INPG/CNRS, BP. 46, Saint Martin d'Hères, 38402, France

ABSTRACT

Perceptual hashing has to deal with the constraints of robustness, accuracy and security. After modeling the process of hash extraction and the properties involved in this process, two different security threats are studied, namely the disclosure of the secret feature space and the tampering of the hash. Two different approaches for performing robust hashing are presented: Random-Based Hash (RBH) where the security is achieved using a random projection matrix and Content-Based Hash (CBH) were the security relies on the difficulty to tamper the hash. As for digital watermarking, different security setups are also devised: the Batch Hash Attack, the Group Hash Attack, the Unique Hash Attack and the Sensitivity Attack. A theoretical analysis of the information leakage in the context of Random-Based Hash is proposed. Finally, practical attacks are presented: (1) Minor Component Analysis is used to estimate the secret projection of Random-Based Hashes and (2) Salient point tampering is used to tamper the hash of Content-Based Hashes systems.

1. DEFINITIONS FOR ROBUST PERCEPTUAL HASHING

Robust perceptual hashing consists of extracting a small-dimensional vector called either a hash, a signature or a fingerprint from a high-dimensional content. This technique enables to perform authentication or identification of contents that have been referenced in a database. Robust perceptual hashing has many applications:

- It can be used to prevent forgery of physical contents like medicine packages, bank notes or valuable watches by checking that the hash of the content under scrutiny belongs to the database of genuine contents ;[1]

- It can be used to filter digital contents, for example this technology is used on open Web2.0 services like YouTube to filter the contents that are to be uploaded into the data-base ;[2]

- It can also be used in digital watermarking in order to generate content-dependent watermarks and perform content authentication via watermark detection.[3]

Moreover, robust perceptual hashing has to fulfill three constraints:

1) **Robustness to distortions**: refers to the ability of the hash function to produce asymptotically the same output based on inputs that differ by legitimate distortion level that can be a consequence of signal processing and/or desynchronization transformations applied to a multimedia data;

2) **Security**: refers to the property that the modification of the hash should not be easily tractable for an adversary;

3) **Universality**: refers to the performance of the hash which has to be optimal or asymptotically optimal in the case of lack of prior knowledge about the statistics of input source distribution.

Additional constraints come from the facts that on one hand the **hash length** has to be as small as possible in order to guarantee a fast scan in the hash database, and on the other hand, as will be shown in this paper, the **hash length** has to be as large as possible to satisfy performance requirements.

It is important to notice that the constraint of security means that either the hash relies on a secret key (cf. Kerckhoffs' principle[4]), or that modification of the hash is not possible. Note that while typical security attacks in watermarking are solely associated with the estimation of the secret key of the algorithm ,[5,6] security threats

Further author information: (Send correspondence to P.B)
P.B: E-mail: Patrick.Bas@lis.inpg.fr

for perceptual hashing are numerous since the adversary can devise two different attacks: she can either try to estimate the secret key used to generate the hash or she can try to simply modify the content in order to tamper the hash if its extraction scheme is public.

Robust hashes are constructed by extracting robust features from contents which are afterward quantized using either a scalar quantizer or a vector quantizer. To decide whether a content belongs to a database of contents or not, a query is performed using the hash database. It consists of extracting the hash of the queried content and looking for the most similar hash in the hash database. For example, the similarity can be computed using a L_n distance or the angle between each vector. From a geometrical point of view, generating a set of N robust distinct hashes from N distinct contents consists of extracting N feature vectors of smaller dimension. The robustness is guaranteed by the fact that the distance between each vector and its nearest neighbours is as large as possible.

Our notations denote a set $\mathcal{H} = \{\mathbf{x} \in \mathbb{R}^{N_v}\}^{N_h}$ of vectors/contents to keep track of with some secret key K, where N_v (resp. N_h) is the dimensionality of the input data vector (resp. the hash). Moreover, $|\mathcal{H}| = N_c$. $\psi_K(\mathbf{x}) \in \mathbb{R}^{N_h}$ denotes the robust hash representation of $\mathbf{x}$. To draw a general picture of the context of robust hashing, we can distinguish two different ways to create robust hashes (see Fig. 1): the Random-Based Hashes (RBH) which are motivated by statistical considerations and the Content-Based Hashes (CBH)which are motivated by content analysis.

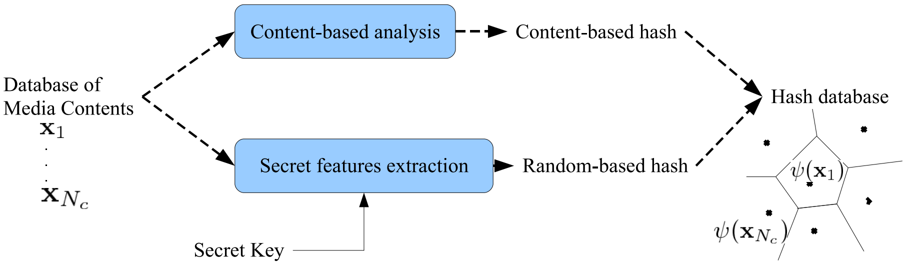


Figure 1. Two different ways to extract robust hashes: content-based analysis and random feature extraction.

2. RANDOM-BASED HASHES AND CONTENT-BASED HASHES

The goal of this section is to present the two classes of robust hash and to give a theoretical framework for each.

2.1 Random-Based Hashes (RBH)

Random-Based hashes have been devised in order to cope with both robustness and security constraints, ideally disregarding the statistics of the input data. In a general way, RBH are generated using a secret key, the hash is built using secret projections [3,6] and the identification or query procedure is done by evaluating the distance between the stored hashes and the one extracted from the content. It should be pointed out that besides this identification problem, the authentication problem can be handled by accepting/rejecting the content based on a single verification trial that consists of the mentioned distance evaluation between the hash extracted from the content and some piece of side information. However, we mostly concentrate on the analysis of issues relevant with the former setup (identification) but we shall give some highlights on authentication issues when needed.

Building a RBH consists of projecting the content vector $\mathbf{x}$ on a set of random vectors to obtain a N_h-dimensional projected vector $\tilde{\mathbf{y}}$ and quantizing $\tilde{\mathbf{y}}$ to obtain the hash vector $\psi_K(\mathbf{x})$ (see Fig. 2).

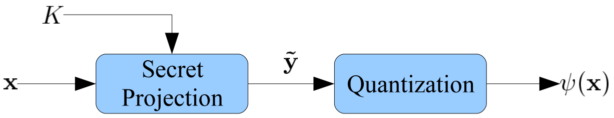


Figure 2. Different fundamental blocks for computing a Random-Based Hash.

In order to ensure robustness, the whole system has to be designed in such a way that contents with different hashes must have non-overlapping identification regions. Using the Central Limit Theorem we can state that the i^{th} projection $\tilde{y}_i(\mathbf{x}_k)$ of the content $\mathbf{X}_k \sim \mathcal{N}(0, \sigma_x^2 \mathbf{I}_{N_v})$ on the secret vector $\mathbf{a}_i$ is Gaussian with law $\mathcal{N}(0, \sigma_x^2)$ provided that $||\mathbf{a}_i|| = 1$.

Let d denote the Euclidean distance between two hash vectors related to two distinct contents $\mathbf{x}_1$ and $\mathbf{x}_2$. We have:

$$d^2 = \sum_{i=1}^{N_h} (\tilde{y}_i(\mathbf{x}_1) - \tilde{y}_i(\mathbf{x}_2))^2.$$

$\frac{d^2}{\sigma_{\tilde{y}}^2}$ follows a χ^2 distribution with N_h degrees of freedom, which means that the average square distance between two hashes is given by $E[d^2] = \sigma_{\tilde{y}}^2 N_h$. This means that the average distance between two distinct hashes increases regarding the dimension of the hash. However, this distance tends to decrease regarding the number of contents to be indexed. Fig. 3 depicts the cumulative density function of the distance from one projection to its nearest neighbour for different database sizes[*]. Hence, in order to maximize the robustness of a RBH scheme w.r.t. a given database size and a given hash size, the scheme needs to maximize the minimum distance between each hashes. It can be performed in two different ways:

- either by modifying the content before computing its hash. This strategy is similar to a watermarking technique. In the following, we call this hash an Active Random-Based Hash (ARBH). It can be used when the system that produces the content has a way to alter the content before extracting the hash;

- either by specifying the random projections and the quantization cells is such a way that the distances between each hash and its closest one are as high as possible. We call this strategy a Passive Random Based Hash: PRBH.

Note also that in order to guaranty robustness towards geometrical transforms, the private hashes rely on an external synchronisation system based for example on the content geometry.

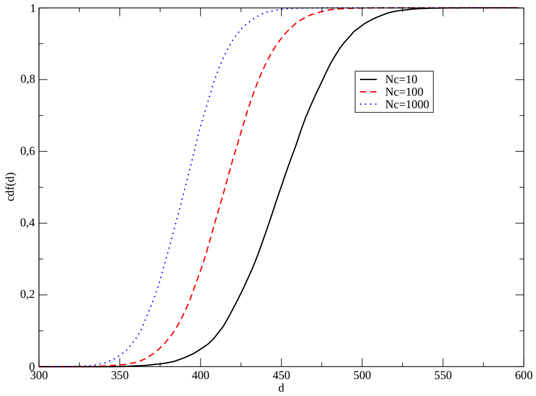


Figure 3. Cumulative density function of the distance between one vector and its nearest neighbour for different number of contents ($N_v = 256$).

2.2 Content-Based Hashes (CBH)

Content-Based Hashes (CBH) are based on the content itself. They come from the scientific community working on Content-Based Retrieval and they use features such as Scale-Invariant Feature Transform (SIFT) descriptors[7] in order to build descriptors for an image.
¿From this construction, the robustness toward both geometrical transforms (affine or projective transforms) and classical processing (noise addition, blur, compression) is important. However, since secret selection of feature points is usually not robust to geometrical transforms, these features have to come from a public set of small cardinality.

[*] The law of the closest distance can be approximated with a Fisher-Tippet density.

Here, we will focus on an example where the CBH originates from a feature point detector, this assumption encompasses the most popular descriptors.[7,8]

An overview of the presented feature detection algorithm is illustrated in Fig. 4. First, the image is preprocessed using a low-pass filter to improve the robustness of the detector. The Harris corner detector function is afterward used. We may approximate this function by a 2D local cross-correlation filter that is applied on the derivative of the image. Then a local competition step is used to tend to an uniform distribution of the feature points inside the image and finally the N most important feature points are selected using the Harris criterion $R(u, v)$ after the competition process .[9]

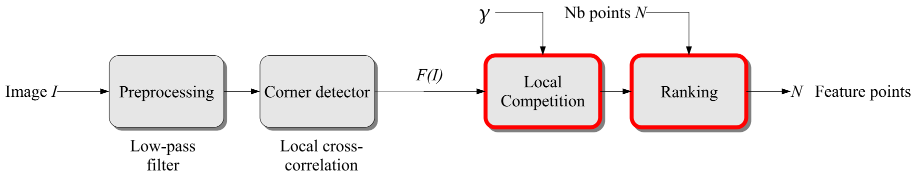


Figure 4. Overview of a feature-point detector used for generating Content-based Hashes.

It is first important to point out that a feature detector can be considered by definition as a public detector. As far as we know, attempts to find image features that are both secure and robust to geometric distortion are seldom [10] and are efficient only for simple geometric transforms such as rotations and scaling operations. We can therefore make the assumption that the attacker can have access to the location of the feature points and that one possible attack is to identify and try to remove feature points that are initially extracted.

The figure 4 outlines the four different modules that composes the presented scheme. The two first modules are equal or similar to 2D FIR filters, one low-pass filter and one filter that acts as a cross-correlation function. They can be considered as an basic pre-processing function whose output, denoted as $R(u, v)$ is used to decide if a feature point is present or not.

3. SECURITY THREATS: OBJECTIVES AND SETUPS

3.1 Adversary's objectives

We have to distinguish two different objectives an adversary may have. One possible goal of the adversary is to trick the authentication system by producing fake objects that are accepted as genuine by the system. Another objective is to alter the content in such a way that the corrupted content is no longer associated with the former hash. Both can be considered tampering attacks: the first one is a *copy attack* and the second one is a *removal attack*.

As to the copy attack, the adversary starts from a fake content and modifies it while minimizing the attack distortion in order to create a content that will have the same hash as one in the hash database.

The constraint of minimizing the distortion corresponds to the assumption that the means of the adversary are limited: if he wants to create a perfect copy, we assume that he cannot clone the original content but can only try to produce a fake that will fall into the correct identification region. Let us call $\mathbf{z}$ a content that does not belong to the hash database. In the simplest additive scenario, the goal of the adversary here is to find the attacking vector $\mathbf{d}$ such that the hash of $\mathbf{z} + \mathbf{d}$ belongs to the hash database while minimizing the power of $\mathbf{d}$ (see Fig. 5). For the removing attack, the adversary's aim will be to find $\mathbf{d}$ such that $\mathbf{x} + \mathbf{d}$ does not belong to the hash database while minimizing the power of $\mathbf{d}$.

3.2 Security threats

Depending on a particular application and available prior knowledge about the identification/authentication system and accessible counterfeiting/reproduction means, different attacking strategies might be applied. Basically, the whole security of a robust hash system can be jeopardized by two threats:

1. Identify the secret features involved in the computation of the hash;

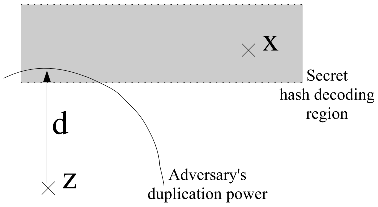


Figure 5. Because the adversary will have a limited power to try to duplicate a fake content, she will look for the vector **d** which will move the fake content into the hash decoding region.

2. Copy or alter the hash. Given a genuine object represented by the codeword $\mathbf{x}$, the physical clonability attack targets to physically produce a codeword $\mathbf{x}_c$ that will be in the acceptance region R .[11] This is only possible if threat (1) is achieved or if the features are public.

Threat (1) can be considered as a first security barrier and consequently is a sufficient condition for the whole security of the system. However, because CBH use public features, only threat (2) will be considered in this framework.

3.3 Security setups

To increase her chances for success, the counterfeiting adversary may apply a learning strategy similar to *supervised learning* used in classification and pattern recognition .[12] Several strategies are possible depending on the availability of the unlabeled or somehow protected codebook of the original codewords and access to the identification/authentication service/device. These strategies are ranging from a *brute force attack*, when the adversary sequentially tests all available sequences from the genuine codebook for each m and then applies the above informed attacks (or directly tries to learn the appropriate indexes for the faked codewords), to the smart "sensitivity"-like attacks [13, 14] recently proposed to fool watermark detectors, when the adversary is using the results of previous tests to learn the decision regions and in this way to gain the lacking knowledge.

Basically, the adversary can devise different attacks scenarios depending on the material she has access to. Those setups are close to the setups used for watermarking security:[15]

1) the **Batch Hash Attack (BHA)**: the adversary observes N_o contents having the same hash. This setup is equivalent to the Constant Message Attack (CMA) proposed in watermarking;[16]

2) the **Group Hash Attack (GHA)**: the adversary observes N_g groups of contents, each having the same hash. The total of observed content is N_o;

3) the **Unique Hash Attack (UHA)**: the adversary observes N_o contents associated with N_h different hashes. Since hashes may be unknown but can be substituted by virtual label, this setup is equivalent to the Known Message Attack in watermarking.

4) the **Sensitivity Attack (SA)**: the adversary has access to the hash generator and can observe the pair (content/hashes) for any contents. This attack is similar to the sensitivity attack known in watermarking.

4. THEORETICAL ANALYSIS OF RANDOM-BASED HASH

This section presents a security analysis of RBH systems when both contents and hashes are known. Our analysis is accomplished within Kerckhoffs security framework.[4] Thus, we assume a complete transparency of the protocol, meaning that the only unknown random parameter of the scheme is a secret key $K = k$. It is assumed that it comes from a given set $\mathcal{K} = \{1, 2, ..., |\mathcal{K}|\}$. The ultimate goal of this analysis is to estimate the complexity to reveal the secret $K = k$ provided all the available prior knowledge.

A similar strategy as in paper[17] is used, the problem under analysis is formulated in terms of Shannon equivocation [18] that is defined in cryptography as the ambiguity about the secret that remains after observing the cyphertexts. For the case of security analysis of robust perceptual hashing, equivocation can be redefined as follows:

$$H(K|\psi_K(\mathbf{x})) = H(K) - I(K; \psi_K(\mathbf{x})), \tag{1}$$

where $H(K)$ and $H(K|\psi_K(\mathbf{x}))$ are entropy of the secret key K and conditional entropy of K given a particular content $\mathbf{x}$ and its hash $\psi_K(\mathbf{x})$, respectively. $I(K;\psi_K(\mathbf{x}))$ stands for mutual information between K and $\psi_K(\mathbf{x})$.[19] It is worth mentioning here that according to the assumptions adopted in this paper K and $\mathbf{x}$ are independent and $H(K|\psi_K(\mathbf{x}),\mathbf{x}) = H(K|\psi_K(\mathbf{x}))$.

The technique presented in[17] provides an upper limit on the equivocation. Therefore, the main goal of this section is to provide a more accurate security leakage characterization.

4.1 Security evaluation of practical robust RBH

A block diagram of the first method[1] inspired by[20] is presented in Fig. 6.

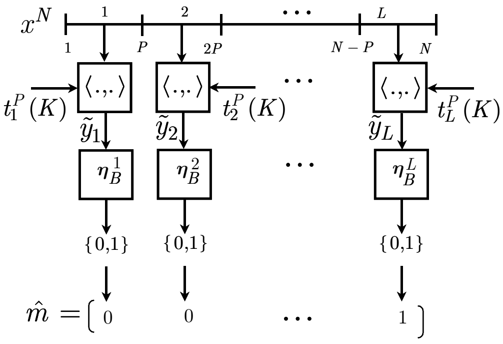


Figure 6. A practical robust perceptual hashing algorithm based on block random projections.

Adopting the same notations as in,[17] it is supposed in this section that identification / authentication of $\mathbf{x}$ follows the next steps. First, it is block-wise randomly projected onto a set of random K-dependent patterns $t_i^P(K), i \in \{1,2,...,L\}$. Then, the outputs of this stage $\tilde{\mathbf{y}}_i, i \in \{1,2,...,L\}$, are used by binary hypothesis testing blocks $\eta_B^i, i \in \{1,2,...,L\}$ to form the hash. However, we do not pay attention to this last step and concentrate on the secret part of the method:

$$\tilde{\mathbf{y}}^L = T^{LP}(K)\mathbf{x}, \tag{2}$$

where $T^{LP}(K)$ is a random projection operator with the following structure:

$$T^{LP}(K) = \begin{pmatrix} T_1^{P\times 1}(K) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & T_2^{P\times 1}(K) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & T_L^{P\times 1}(K) \end{pmatrix}, \tag{3}$$

where $\mathbf{0}$ is a $P \times 1$ vector of zeros and $T_i^{P\times 1}(K) = t_i^P(K), i \in \{1,2,...,L\}$.

Due to the unique correspondence between $K = k$ and $T^{LP}(K)$, Eq. 1 can be redefined as:

$$I(K;\psi_K(\mathbf{x})) = I(T^{LP}(K);\psi_K(\mathbf{x}))). \tag{4}$$

Similar to,[17] the main goal of this section is the estimation of the amount of information about $T_{i,j}(K)$, where $T_i^{P\times 1}(K) = [T_{i,1}(K), T_{i,2}(K), ..., T_{i,j}(K), ..., T_{i,L}(K)], 1 \le j \le P$, from observing a binary hash value $\{0,1\}$. The analysis targeting this security leakage evaluation will be performed based on the following assumptions:

- projection operators $T_i^{P\times 1}(K)$ acting as $T : \mathbb{R}^P \to \mathbb{R}^1$ are i.i.d. distributed accordingly to a multivariate zero mean Gaussian distribution with a covariance matrix $\sigma_T^2 \mathbb{I}^{P\times P}$. Media data vectors on the block level $\mathbf{x}_i^P = [\mathbf{x}_i[1], \mathbf{x}_i[2], ..., \mathbf{x}_i[P]]$ are considered to be composed of i.i.d. realizations of a uniform distribution supported on the interval $[0,1]$ similarly to;[21]

- hash bits are obtained from a projection outputs as binary labels of an optimal one-bit rate-distortion code[22] that will minimize the mutual information between a projection output and a hash bit.

Based on these assumptions, an equivalent protocol of a transmission of $T_{i,j}(K)$ can be presented as it is shown in Fig. 7.

The binary channel output (Fig. 7) is obtained by assigning a binary label to a codeword that performs an optimal rate-distortion reconstruction of $(\mathbf{x}_i[j]T_{i,j}(K) + Z_{i,j})$, where $Z_{i,j} = \sum_{l=1,l\neq j}^{P} \mathbf{x}_i[l]T_{i,l}(K)$ follows a zero mean Gaussian distribution with variance $\sigma^2_{\tilde{Y}_i} = \sigma^2_T \sum_{j=1}^{P}(\mathbf{x}_i[j])^2$. It was assumed that $\sigma^2_{\tilde{Y}_i} = \sigma^2_{\tilde{Y}}$ for all blocks. Such a statistical behavior of $Z_{i,j}$ is justified similarly to[21] using the Central Limit Theorem.[19]

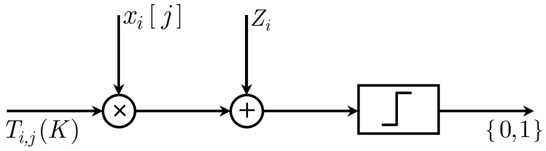


Figure 7. An equivalent protocol for a lossy transmission of a random projection component $T_{i,j}(K)$.

The last block in the equivalent channel diagram (Figure 7) can be further detailed using an equivalent Gaussian forward test channel from the rate distortion theory.[22] Then a complete channel model for noisy transmission of the secret key related information can be designed as presented in Figure 8.

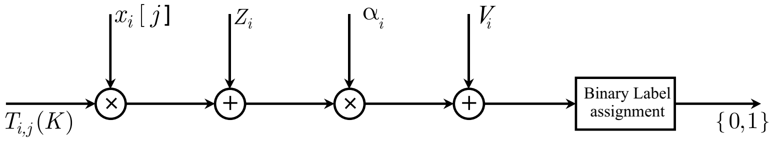


Figure 8. A complete equivalent protocol for a lossy transmission of a random projection component $T_{i,j}(K)$.

Thus, the optimal reconstruction is performed by amplitude scaling of $(\mathbf{x}_i[j]T_{i,j}(K) + Z_{i,j})$ by a factor α_i that follows by addition of an independent Gaussian noise $V_i \sim \mathcal{N}(0, \sigma^2_{V_i})$. In order to identify parameters α_i and $\sigma^2_{V_{i,j}}$) the following observations are necessary.

First, as it was already mentioned, the projection output $\tilde{\mathbf{y}}_i = \mathbf{x}_i[j]T_{i,j}(K) + Z_{i,j}$ has a zero-mean Gaussian distribution with variance $\sigma^2_{\tilde{\mathbf{y}}_i} = \sigma^2_T \sum_{j=1}^{P}(\mathbf{x}_i[j])^2$. Second, the rate distortion function of a Gaussian source with zero-mean and variance $\sigma^2_{\tilde{\mathbf{y}}_i}$ is given by:[22]

$$R_i = \frac{1}{2}\log_2\left(\frac{\sigma^2_{\tilde{\mathbf{y}}_i}}{D}\right) \tag{5}$$

for the distortion level satisfying $D \leq \sigma^2_{\tilde{\mathbf{y}}_i}$. Since, according to the adopted assumption $R = 1$, one has that $D = \frac{\sigma^2_{\tilde{\mathbf{y}}_i}}{4}$. Then, following a similar reasoning as in,[22] one obtains:

$$\alpha_i = 1 - \frac{D}{\sigma^2_{\tilde{\mathbf{y}}_i}} = \frac{3}{4}; \tag{6}$$

$$\sigma^2_{V_i} = D - \frac{D^2}{\sigma^2_{\tilde{\mathbf{y}}_i}} = \frac{3\sigma^2_{\tilde{\mathbf{y}}_i}}{16}. \tag{7}$$

Then, denoting the input to the binary label assignment block as $\tilde{T}_{i,j}(K)$, one has that:

$$\tilde{T}_{i,j}(K) = \alpha_i \mathbf{x}_i[j]T_{i,j}(K) + \alpha_i Z_{i,j} + V_i \tag{8}$$

is a zero-mean Gaussian random variable with variance $\sigma_{\tilde{T}}^2 = \frac{12}{16}\sigma_T^2 \sum_{j=1}^{P}(\mathbf{x}_i[j])^2$. Thus, $\{T_{i,j}(K), \tilde{T}_{i,j}(K)\}$ follows a jointly Gaussian distribution with a zero mean and the following covariance matrix:

$$\Sigma_{T\tilde{T}} = \begin{pmatrix} \sigma_T^2 & \frac{3}{4}\mathbf{x}_i[j]\sigma_T^2 \\ \frac{3}{4}\mathbf{x}_i[j]\sigma_T^2 & \frac{12}{16}\sigma_T^2 \sum_{j=1}^{P}(\mathbf{x}_i[j])^2 \end{pmatrix}. \tag{9}$$

Finally, the security leakage about the secret of the scheme provided in the RBH setting can be bounded by the following mutual information:

$$I(T_{i,j}(K);\tilde{T}_{i,j}(K)) = \frac{1}{2}\log_2\Big(\frac{\det(\Sigma_{TT})\det(\Sigma_{\tilde{T}\tilde{T}})}{\det(\Sigma_{T\tilde{T}})}\Big) = \frac{1}{2}\log_2\Big(\frac{\sum_{l=1}^{P}(\mathbf{x}_i[j])^2}{\sum_{l=1}^{P}(\mathbf{x}_i[l])^2 - \frac{3}{4}(\mathbf{x}_i[j])^2}\Big), \tag{10}$$

where $\det(\cdot)$ denotes a matrix determinant.

In order to quantify the information leakage about the secret key provided in a RBH setting a set of experiments was accomplished summarized in Fig. 9. The main goal was to measure the mutual information $I(T_{i,j}(K);\tilde{T}_{i,j}(K))$ as a function of a block size P and to compare these results with those reported in.[17] For this purpose, in this paper we adopted the experimental setup used in:[17]

- hashing input on a block level $\mathbf{x}^P$ is an i.i.d. vector of dimensionality $P, P = \{32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 262144\}$, such that $\mathbf{x}$ is uniformly distributed on the interval $[0, 1]$;

- random projection operator $T_i^{1\times P}(K)$ is an i.i.d. zero mean unit variance Gaussian vector of dimensionality P.

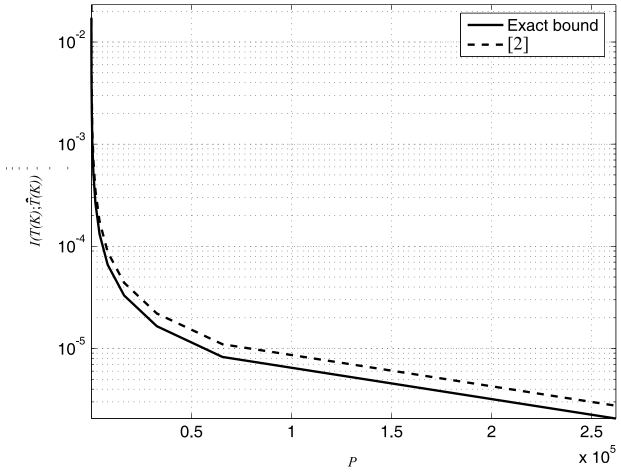


Figure 9. Information leakage about the secret key (per component) measured in the projected domain.

The obtained experimental results allow to conclude that the robust perceptual hashing based on random projections is not provably secure in sense that the scheme is leaking information about the secret to the opponent. The amount of this information is a decreasing function of a hash input block size and thus the conclusion drawn in[17] about the way of increasing security of the scheme by enlarging P remains valid. Finally, in[17] there was only an upper bound on the measured security leakage while a more accurate characterization of this information is provided in this paper.

5. PRACTICAL ATTACKS FOR RBH

The goal of this section is to present different practical security attacks for RBH. Note that contrary to the previous section, we will make no assumption about the value of the hashes.

Since the construction of such a hash relies on a private projection, a security weakness will come if it is possible to estimate the set of secret projections and the position of the quantization cells. The system used to generate the RBH will be the same as the one presented in the previous section, except for the quantization step that will use vector or scalar quantization.

5.1 Contents and hashes generation

The following system is used to generate contents and associated hashes. Contents $\mathbf{y}$ are N_v dimensional vectors and a subspace of dimension N_h is used as a secret subspace, it is fully defined by the matrix of orthogonal basis vectors $\mathbf{A}$. $\mathbf{B}$ is any complementary orthogonal matrix of the subspace orthogonal to span($\mathbf{A}$): span($\mathbf{A}$) $\cup$ span($\mathbf{B}$) $= \mathbb{R}^{\mathrm{N_v}}$ and span($\mathbf{A}$) $\cap$ span($\mathbf{B}$) $= \emptyset$. The quantization step used to compute the hash is a multidimensional lattice.
In order to simulate Active-RBH (ARBH) we use a Spread Transform Distortion Compensation Quantization technique which is close to the Spread Transform Dither Modulation used in data-hiding .[23] The contents are generated from random latent vectors $\mathbf{x}$ using the following formula:

$$\mathbf{y} = \mathbf{A}\mathcal{Q}_{\mathrm{dc}}(\mathbf{A}^T\mathbf{x}, \alpha) + \mathbf{B}\mathbf{B}^T\mathbf{x}$$

where $\mathcal{Q}_{\mathrm{dc}}(\mathbf{u}, \alpha)$ is the distortion compensated quantization function, which is defined by:

$$\mathcal{Q}_{\mathrm{dc}}(\mathbf{u}, \alpha) = \mathbf{u} + \alpha\left(\mathcal{Q}_{\mathcal{L}}(\mathbf{u}) - \mathbf{u}\right)$$

and $\mathcal{Q}_{\mathcal{L}}(.)$ is the quantizer based on the lattice $\mathcal{L}$ (α is the compensation parameter). If $\alpha = 0$, no compensation is done and $\mathbf{y} = \mathbf{x}$. The function $h(\mathbf{x}) = \mathcal{Q}_{\mathcal{L}}(\mathbf{A}^T\mathbf{x})$ enables to return the hash of the content $\mathbf{x}$ and the property $h(\mathbf{x}) = h(\mathbf{y})$ is straightforward.

The simulations were conducted using $N_v = 128$, $N_h = 8$ and each vector $\mathbf{x}$ is generated using N_v Gaussian independent random variables $x_i \sim \mathcal{N}(0, 1)$. The lattice $\mathcal{L}$ is the E_8 lattice ,[24] we have chosen it for its capabilities to provide a solution to the sphere packing problem in 8 dimensions and because it has already been using to create robust hashes.[25]

As it is presented in section 3, the adversary's first goal is to find an accurate estimation of the secret projection matrix $\mathbf{A}$. This first step will enable to make a copy attack or a removal attack easier. The assessment of the security for different setups is done using the principal angles between the two subspaces.[26] This measurement has already been proposed for security assessment of watermarking schemes.[27] From the estimate $\hat{\mathbf{A}}$ of the secret matrix $\mathbf{A}$, the Singular Value Decomposition of $\hat{\mathbf{A}}^T\mathbf{A}$ is performed and yields to N_h singular values sv_i. Afterwards the square-chordal distance d^2 is given by:

$$d^2 = \sum_{i=1}^{N_h} \sin(\arccos(v_i))^2.$$

When span($\hat{\mathbf{A}}$) = span($\mathbf{A}$), the properties of the chordal distance are that $d^2 = 0$ and when span($\mathbf{A}$)$\cap$span($\hat{\mathbf{A}}$) $=$ $\emptyset$ $d^2 = N_h$.

5.2 Subspace estimation via Minor Component Analysis

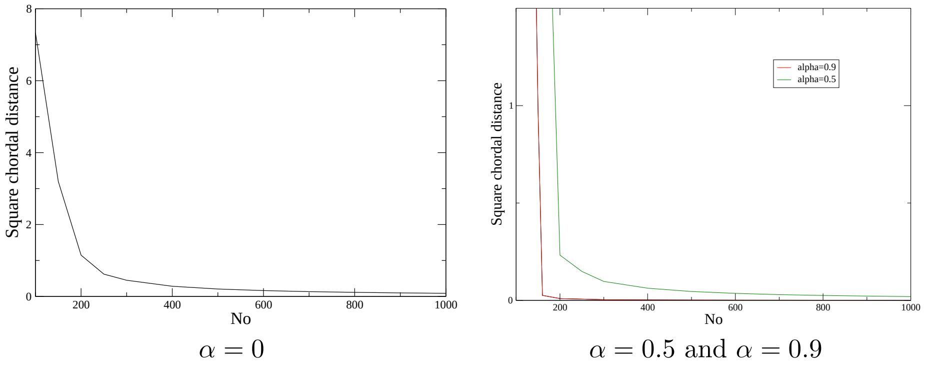
One possible attack is to notice that all the contents having the same hash will belong to the same quantization cell of the lattice, which means that the variance on the observations along the directions of the secret subspace will be smaller than on the other directions. It is consequently possible to perform Minor Component Analysis (MCA) in order to estimate the secret subspace. As Principal Component Analysis, this method relies on the eigenvalue decomposition of the covariance matrix $\mathbf{C}$ given by:

$$\mathbf{C} = E[(\mathbf{y} - E[\mathbf{y}])(\mathbf{y} - E[\mathbf{y}])^T].$$

The K minor components are given by the K eigenvectors associated with the K smallest eigenvalues, they are arranged as the columns of $\hat{\mathbf{A}}$.

We first investigate the behaviour of MCA in the BHA setup. Fig. 10 shows the evolution of the square chordal distance between the estimated and the secret subspaces according to the number of observations N_o for different compensation values ($\alpha = 0$, $\alpha = 0.5$ and $\alpha = 0.9$). We can see that the robustness of the hash system (the more α close to 1, the more robust) impacts directly the security of the system and that even with the less robust setup ($\alpha = 0$) the security of the system is weak: an accurate estimation of the subspace (e.g. a square chordal distance below 0.1) is possible if 900 contents are available to the adversary. When $\alpha = 0.9$, only 160 contents are needed, this is due to the fact that only 128 orthogonal vectors are theoretically needed to provide a basis of the whole space.

The performances of MCA for the Group Hash Attack setup are depicted on Fig. 11 for $N_o = 1000$ and different number of hashes. We can observe that the number of distinct hashes available for the adversary impairs the accuracy of the estimation. This result can be explained by the fact that by increasing the number of hashes, the variance of observations on the secret subspace becomes more and more important (which reduces the efficiency



$\alpha = 0$ $\alpha = 0.5$ and $\alpha = 0.9$

Figure 10. Batch Hash Attack setup: Performances of the MCA for different distortion parameters.

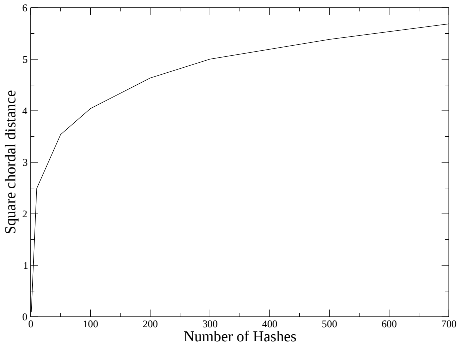


Figure 11. Group Hash Attack setup: Performances of the MCA for different number of hashes ($\alpha = 0$, $N_{=}1000$).

6. PRACTICAL ATTACKS FOR CBH

In this case, the whole security of the system relies on the possibility to devise a tampering attack.[28] Since the hash construction is based on public content-based features, the problem is: can the adversary devise an attack in order to alter the hash in such a way that the identification system will be fooled? This problem is known in watermarking as finding the optimal tampering attack and the performance of this attack will depend of the distortion introduced in order to modify the hash and the computational cost of the optimisation problem.

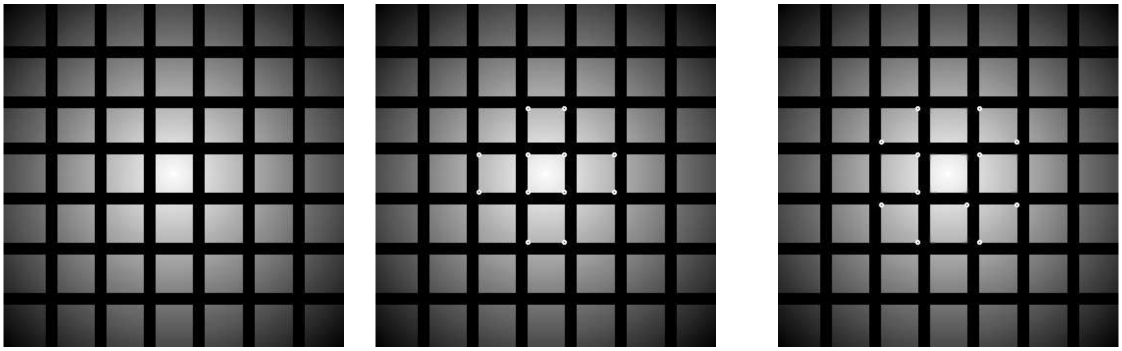
To the best of our knowledge, Fridrich[3] was the first who included a security analysis of a robust hashing method which is based on public features. The author suppose that the adversary may know the basis that is used to extract the robust features. The distortion that is induced by the modification of a set of features is evaluated: on average 13 of 50 bits can be flipped with such a strategy while inducing an imperceptible distortion for the presented hashing system.

In this section, we focus on the public feature extraction scheme presented in Section 2.2 to present two classes of attacks. The adversary might draw malicious attacks by altering the image in such a way that the output function will afterward lure either the third *competition step* or the fourth *ranking step*. Both these last two processes are part of the final decision procedure and the modification of $R(u, v)$ has to be done according to the decision criterion of either the third or the fourth step.

One simple way to alter the Harris criterion $R(u, v)$ at a given location (u_i, v_j) is to modify the variance of a block centered on (u_i, v_j). This property is due to the fact that the Harris detector is somewhat similar to a local cross-correlation function of the derivate of the signal (or a low-pass version of the signal). Consequently increasing (resp. reducing) the variance of a block centered on (u_i, v_j) enables to increase (res. decrease) the output of $R(u_i, v_j)$. In practice, the size of the block has to be the same than the size n of the blur filter M_n that is used as a pre-processing step.

6.1 Attack based on the local competition

The illustrative image *toy1*, is an image that contains lot of corners, additionally the function $R(u, v)$ is maximum on the center of the image and is a decreasing function of the distance from the center of the image. Fig. 12 depicts the image *toy1* and its associated detected feature points. We can see that, due to the local competition, only one on the four nearest corner is selected. It is always the corner that is next to the center of the image, this is due to the fact that this point is a local maximum in this case. A successful attack has been performed by locally blurring several corners to decrease $R(u, v)$, this enables to totally change the set of feature points that is used. We can also notice that the distortion introduced by this attack is not important, because we have only decreased the variance of the blocks centered on the attacked feature points.

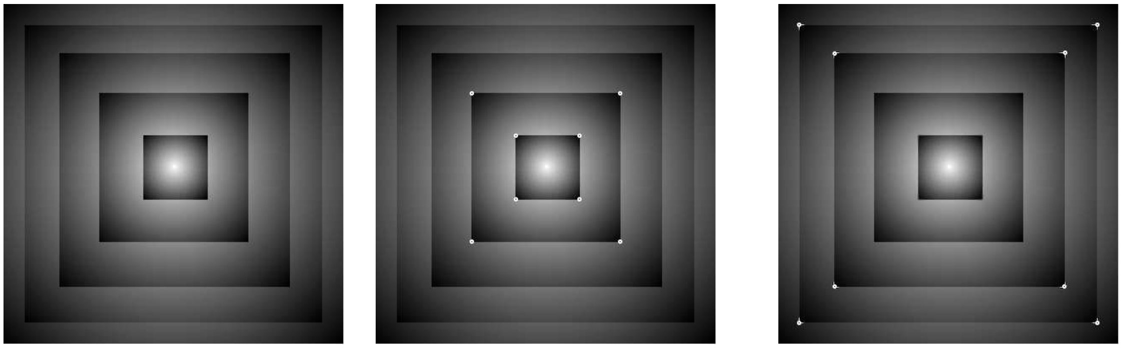


Toy1 image — Image and feature points — Attacked image and feature points

Figure 12. Example of the attack based on the local competition. The PSNR of the attack equals 44.2dB.

6.2 Attack based on the final ranking

The example image *toy2* has been built in such a way that no *competition attack* is possible because the corners are too far from each other. Consequently, only the *ranking attack* is possible. Height feature points have been extracted by the algorithm (see Fig. 13) and the image has been synthesized in such a way that $R(u, v)$ is a decreasing function of the distance from the center of the image. The attack has been performed here by increasing the variance of the other corners in such a way that at these locations, the function $R(u, v)$ becomes greater than in the previous locations. This is done by increasing the variance of the new positions up to have a more important output than for the initial feature point. It is important to notice that such an operation may require an important distortion because each new feature point has to have a more important response than the feature point that is erased.



Toy2 image

Image and feature points

Attacked image and feature points

Figure 13. Exemple of the attack based on the final ranking. The PSNR of the attack equals 35.8dB.

7. CONCLUSION AND PERSPECTIVES

In this paper, we have presented two classes of robust-hashing techniques named Random-Based Hashing or Content-Based Hashing systems and we have performed a security analysis for each class. The security of each class is very different: the security of RBH systems relies on secret transforms like secret projections and the security of CBH systems relies on sensitivity of the content descriptor to tampering attacks. However, the definition of security setups enables to foresee potential information leakage of the secret key. Our study has been conducted for basic RBH systems (linear projection followed by quantization) and basics CBH systems (feature point detector on simple images) in order to show that security issues can arise. Future work should be conducted on more sophisticated content-based hashing schemes which use a more important number of descriptors[29] and on practical implementations of random-based hashing systems.[3]

8. ACKNOWLEDGEMENTS

O. Koval and S. Voloshynovskiy thank the SNF Professeur Boursier grant 114613, SNF projects 20020-121635 and 200021-119770 and Swiss IM2 project.

P. Bas and F. Cayre thank the National French projects ACI-SI Nebianno, RIAM Estivale and ARA TSAR for their financial support. The authors thanks also Hervé Jégou for his highlights on content-based retrieval techniques.

REFERENCES

[1] Villán, R., Voloshynovskiy, S., Koval, O., Deguillaume, F., and Pun, T., "Tamper-proofing of electronic and printed text documents via robust hashing and data-hiding," in [*Proceedings of SPIE-IS&T Electronic Imaging 2007, Security, Steganography, and Watermarking of Multimedia Contents IX*], (28 Jan. – 1 Feb. 2007).

[2] Steinert-Threlkeld, T., "Zdnet undercover: Youtube's video id system: Is 75 percent accuracy good enough?," tech. rep. (November 2008).

[3] Fridrich, J., "Robust hash functions for digital watermarking," in [*ITCC '00: Proceedings of the The International Conference on Information Technology: Coding and Computing (ITCC'00)*], 178, IEEE Computer Society, Washington, DC, USA (2000).

[4] Kerckhoffs, "La cryptographie militaire," in [*Journal des Sciences Militaires*], *9* **IX**, 5–38 (Janvier 1883).

[5] Cayre, F., Fontaine, C., and Furon, T., "Watermarking security: theory and practice," *IEEE Trans. Signal Processing* **53** (oct 2005).

[6] Comesaña, P., Pérez-Freire, L., and Pérez-González, F., "Fundamentals of data hiding security and their application to spread-spectrum analysis," in [*7th Information Hiding Workshop, IH05*], *Lecture Notes in Computer Science*, Springer Verlag, Barcelona, Spain (June 2005).

[7] Lowe, D., "Object recognition from local scale-invariant features," in [*Proceedings of the International Conference on Computer Vision*], **2**, 1150–1157.

[8] Mikolajczyk, K. and Schmid, C., "Indexing based on scale invariant interest points," *IEEE International Conference on Computer Vision* **1**, 525 (2001).

[9] Bas, P., Chassery, J.-M., and Macq, B., "Geometrically invariant watermarking using feature points," *IEEE Trans. on Image Processing* **11**(9), 1014–1028 (2002).

[10] Delannay, D. and Macq, B., "Method for hiding synchronization marks in scale and rotation resilient watermarking schemes," in [*Proceedings of SPIE, Security, Steganography and Watermarking of Multimedia Contents IV*], (Jan. 2002).

[11] Koval, O., Voloshynovskiy, S., Beekhof, F., and Pun, T., "Analysis of physical unclonable identification based on reference list decoding," in [*Proceedings of SPIE-IS&T Electronic Imaging 2008, Security, Steganography, and Watermarking of Multimedia Contents X*], (28–31 Jan. 2008).

[12] Duda, R., Hart, P., and Stork, D., [*Pattern Classification*], Wiley and Sons, New York (2001).

[13] Linnartz, J. P. and van DIjk, M., "Analysis of the sensitivity attack against electronic watermarks in images," in [*International Information Hiding Workshop*], (April 1998).

[14] Choubassi, M. E. and Moulin, P., "Noniterative algorithms for sensitivity analysis attacks," *IEEE Trans. Information Forensics and Security* **2**, 113–126 (June 2007).

[15] Cayre, F., Fontaine, C., and Furon, T., "Watermarking security part I: Theory," in [*Proceedings of SPIE, Security, Steganography and Watermarking of Multimedia Contents VII*], **5681** (Jan. 2005).

[16] Pérez-Freire, L., Pérez-González, F., Furon, T., and Comesaña, P., "Security of lattice-based data hiding against the Known Message Attack," *IEEE Transactions on Information Forensics and Security* **1**, 421–439 (December 2006).

[17] Koval, O., Voloshynovskiy, S., Beekhof, F., and Pun, T., "Security analysis of robust perceptual hashing," in [*IS&T/SPIE Electronic Imaging'7, Session: Security and Watermarking of Multimedia Contents*], (January 2008).

[18] Shannon, C. E., "Communication theory of secrecy systems," *Bell System Technical Journal* **28**, 656–715 (1949).

[19] Cover, T. and Thomas, J., [*Elements of Information Theory.*], Wiley and Sons, New York (1991).

[20] Fridrich, J., "Robust bit extraction from images," in [*Proceedings ICMCS'99*], **2**, 536–540 (June 1999).

[21] Swaminathan, A., Mao, Y., and Wu, M., "Robust and secure hashing for images," *IEEE Transactions on Information Forensics and Security* **1**, 215–230 (June 2006).

[22] Berger, T., [*Rate Distortion Theory: A Mathematical Basis for Data Compression*], Prentice-Hall, Englewood Cliffs, NJ (1971).

[23] Chen, B. and Wornell, G. W., "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," **47**(4), 1423–1443 (2001).

[24] Conway, J. H. and Sloane, N. J. A., [*Sphere Packings, Lattices and Groups*], Springer-Verlag, New York (1998).

[25] Jegou, H., Amsaleg, L., Schmid, C., and Gros, P., "Query-adaptative locality sensitive hashing," in [*International Conference on Acoustics, Speech, and Signal Processing*], IEEE (apr 2008).

[26] Knyazev, A. V. and Argentati, M. E., "Principal angles between subspaces in an a-based scalar product," in [*SIAM, J. Sci. Comput.*], **23**, 2009–2041, Society for Industrial and Applied Mathematics (apr 2002).

[27] Pérez-Freire, L. and Pérez-González, F., "Spread spectrum watermarking security," *IEEE Transactions on Information Forensics and Security* (To appear 2008).

[28] Bas, P. and Doërr, G., "Evaluation of an optimal watermark tampering attack against dirty paper trellis schemes," in [*ACM Multimedia and Security Workshop*], (Sept 2008).

[29] Sivic, J. and Zisserman, A., "Video google: A text retrieval approach to object matching in videos," in [*ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*], IEEE Computer Society, Washington, DC, USA (2003).