Robust Hash Functions for Visual Data: An Experimental Comparison *

Champskud J. Skrepth[1] and Andreas Uhl[1,2]

[1] Carinthia Tech Institute, School of Telematics & Network Engineering
Primoschgasse 8, A-9020 Klagenfurt, Austria
[2] Salzburg University, Department of Scientific Computing
Jakob-Haringerstr.2, A-5020 Salzburg, Austria
uhl@cosy.sbg.ac.at

Abstract. Robust hash functions for visual data need a feature extraction mechanism to rely on. We experimentally compare spatial and transform domain feature extraction techniques and identify the global DCT combined with the cryptographic hash function MD-5 to be suited for visual hashing. This scheme offers robustness against JPEG2000 and JPEG compression and qualitative sensitivity to intentional global and local image alterations.

1 Introduction

The widespread availability of multimedia data in digital form has opened a wide range of possibilities to manipulate visual media. In particular, digital image processing and image manipulation tools offer facilities to intentionally alter image content without leaving perceptual traces. Therefore, it is necessary to provide ways of ensuring integrity other than human vision.

Classical cryptographic tools to check for data integrity like the cryptographic hash functions MD-5 or SHA are designed to be strongly dependent on every single bit of the input data. While this is desirable for a big class of digital data (e.g. executables, compressed data, text), manipulations to visual data that do not affect the visual content are very common and often necessary. This includes lossy compression, image enhancement like filtering, and many more. All these operations do of course change the bits of the data while leaving the image perception unaltered.

To account for this property of visual data new techniques are required which do not assure the integrity of the digital representation of visual data but its visual appearance. In the area of multimedia security two types of approaches have been proposed to satisfy those requirements in recent years: semi-fragile watermarking and robust multimedia hashes (see [2, 3, 6, 4, 8, 10, 11] for some examples for the latter approach).

* This work has been partially supported by the Austrian Science Fund FWF, project no. P15170.

Main advantages of semi-fragile watermarking schemes are that watermarks are inserted into the image and become integral part of it and that image manipulations may be localized in most schemes. The main advantage of hashing schemes is that image data is not altered and not degraded at all.

In this work we focus onto robust visual hash functions to provide a means to protect visual integrity of image data. In particular, we propose to combine the extraction of robust visual features with the application of a classical cryptographic hash function to result in a robust visual hash procedure. In section 2 we first discuss requirements of a robust visual hashing scheme. Subsequently, we introduce several possibilities to extract perceptually relevant visual features in the spatial and transform domain. In section 3, we experimentally evaluate robustness against JPEG 2000 and JPEG compression and sensitivity towards intentional image modification of visual hashing schemes based on the feature extraction techniques proposed in section 2 and the cryptographic hash function MD-5. Section 4 concludes our paper and provides an outlook to future work in this direction.

2 Approaches to Robust Visual Hashing

Similar to cryptographic hash functions, robust hash functions for image authentication should satisfy 4 major requirements [11] (where P denotes probability, H is the hash function, $X, \hat{X}, Y$ are images, α and β are hash values, and $\{0/1\}^L$ represents binary strings of length L):

1. Equal distribution of hash values:
$$P[H(X) = \alpha] \approx \frac{1}{2^L} \ , \forall \alpha \in \{0/1\}^L \ .$$
2. Pairwise independence for visually different images X and Y:
$$P[H(X) = \alpha | H(Y) = \beta] \approx P[H(X) = \alpha] \ , \forall \alpha, \beta \in \{0/1\}^L \ .$$
3. Invariance for visually similar images X and $\hat{X}$:
$$P[H(X) = H(\hat{X})] \approx 1 \ .$$
To fulfil this requirement, most proposed algorithms try to extract image features which are invariant to slight global modifications like compression or filtering.
4. Distinction of visually different images X and Y:
$$P[H(X) = H(Y)] \approx 0 \ .$$
This final requirement also means that given an image X, it is almost impossible to find a visually different image Y with $H(X) = H(Y)$. In other words, it should be impossible to create a forgery which results in the same hash value as the original image. Note that the visual features selected according to requirement 3) are usually publicly known and can therefore be modified. This might threaten security, as the hash value could be adjusted maliciously to match that of another image.

Note that requirements 1), 2), and 4) also apply to cryptographic hash functions, whereas requirement 3) focuses entirely onto the desired robustness property.

In addition to requirements 1) to 4) there is another often requested property. The Hamming distance between the hash values of two images under consideration should serve as a measure of similarity between those images, i.e. should give a quantitative result in the sense of a metric instead of the qualitative result of a cryptographic hash function. Although desirable from the applications viewpoint, this property partially contradicts to requirements 1) and 2) and therefore excludes a cryptographic hash function as a possible component of such a scheme. As a consequence, several visual hash functions with the abovementioned increased functionality but at least questionable security properties have been suggested.

Our approach investigated in this work therefore basically consists of two steps:

- First, features robust to common (non-hostile) image processing operations (we especially focus onto compression) but sensitive to malicious modifications are extracted from the image.
- Subsequently, a classical hash function is applied to those features.

In the following subsections, we introduce the types of feature extraction techniques which are experimentally compared for their robustness against compression and sensitivity towards intentional image modifications in this study. Note that for simplicity we assume 512 × 512 pixels images with 8 bit/pixel (bpp).

2.1 Spatial domain feature extraction techniques

Multiresolution pyramids As a first step we construct a quater-sized version of the image ("approximation") using a 4-pixel average (AV), a 4-pixel median (ME), or subsampling by 2 in each direction (DS). Subsequently, the construction of the approximation is iterated to construct smaller versions. An approximation of specific size is used as feature. Whereas the bitdepth is not influenced by these operations (AV is rounded to integer) we only obtain a limited number of differently sized approximations the hash function may be applied to: 256^2 values for one iteration, 128^2 values after two iterations, …, and $16^2 = 256$ values after five iterations which is the maximal number of iterations we consider.

Bitplanes We consider the 8bpp data in the form of 8 bitplanes, each bitplane associated with a position in the binary representation of the pixels. The feature extraction approach is to consider a subset of the bitplanes only, starting with the bitplane containing the MSB of the pixels. Each possible subset of bitplanes may be chosen as feature, however, it makes sense to stick to the order predefined by the significance of the binary representation. After having chosen a particular subset of bitplanes, the hash function is applied to pixel values which have been computed using the target bitplanes only. Note that the smallest amount of data the hashing may be applied to (i.e. one bitplane) corresponds to 32768 pixels in this case (1/8 of the total number of pixels in the image). Note also that this feature extraction technique BP comes for free from a computational point of view.

2.2 Transform domain feature extraction techniques

In contrast to spatial domain methods the feature extraction operation (i.e. the transform) increases the bitdepth of the data significantly. To obtain comparability to the spatial domain techniques, the range of coefficients is mapped to the interval [0,255] and subsequently rounded to integer values.

DCT The DCT is well known to extract global image characteristics efficiently and is used for watermarking applications for these reasons (see e.g. Cox's scheme [1]). We use the DCT in two flavours: as full frame DCT (DCT1) and as DCT applied to 8×8 pixels blocks (DCT2) due to complexity reasons. Following the zig-zag scan order (compare e.g. JPEG) we apply the hash-function to a certain number of coefficients or a certain number of coefficients from each block, respectively. Given a 512×512 pixels image and using DCT2, the lowest number of coefficients the hash function may be applied to is 4096 (i.e. the DC coefficient is hashed only for each block), whereas the number of coefficients subjected to hashing may be set almost arbitrarily with DCT1.

Wavelet transform In many applications wavelet transforms (WT) compete with and even replace the DCT due to their improved localization properties (e.g., the WT is used in many watermarking schemes [5]). We use the Haar transform due to complexity and sensitivity reasons. Equivalently to the Multiresolution pyramids, the decomposition depth is a parameter for this method, in case of WT the hash function is applied to the approximation subband only. As it is the case for Multiresolution pyramids, we only obtain a limited number of differently sized approximation subbands the hash function may be applied to. Note that the data subject to hashing resulting from applying the WT is equivalent in principle to that obtained by the Multiresolution pyramid AV.

3 Experiments

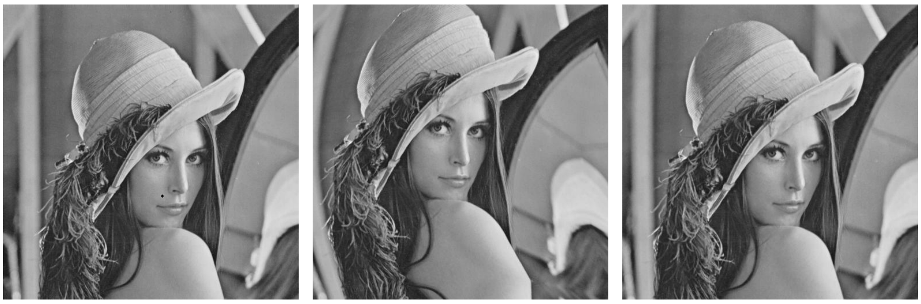
The aim of the experimental section is to investigate whether the introduced visual hashing schemes are

- indeed robust to JPEG and JPEG2000 compression and
- sensitive to intentional image modifications (i.e. attacks).

3.1 Experimental Settings

We use the classical 8bpp, 512 × 512 pixels Lena and Escher grayscale images (see Fig. 2.a for the latter) as testimages. In order to investigate the robustness of the visual hashing schemes, we subject the image Lena to JPEG 2000 (J2K) and JPEG compression with different compression ratios (Cr). The sensitivity to intentional and/or malicious image modifications is assessed by conducting a couple of local and global image alterations:

- Adding a small artificial birthmark to Lenas upper lip (Fig. 1.a, "augmented Lena") - local
- Applying Stirmark [7] attack options b and i (Fig. 1.b and Fig. 1.c) - global
- Increase or decrease the luminance of each pixel of the image Lena by a value of 5 - global
- Addition of an alternating dark/light pattern in a door arch in Eschers painting (Fig. 2.b) - local



(a) Augmented Lena

(b) Stirmark attack b=20

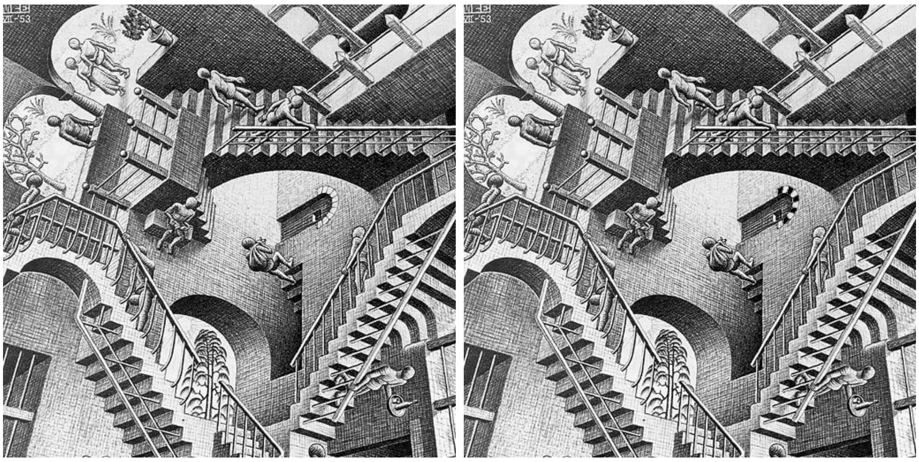
(c) Stirmark attack i=20

Fig. 1. Local and global attacks against Lena.

All feature extraction schemes are implemented using MATLAB®, as hash function we use the well known MD-5 [9] system giving 128 output hashbits. Note that MD-5 may be applied to a certain number of feature values given in full 8bpp precision ("Full") or to feature values with reduced bitdepth by simply ignoring the bits of lower significance (where e.g. 3 BP stands for three bitplanes and MSB for the use of the most significant bitplane only).

3.2 Experimental Results

Using bitplanes as features comes almost for free from the computational point of view. The block-based DCT is the fastest of all other schemes. Following closely



(a) Escher

(b) Escher attacked

Fig. 2. Local attack against a painting by Escher.

are the iterated downsampling and averaging multiresolution pyramids. Note that the latter exhibits equal execution speed as WT if only the approximation subband is generated during the transform. The median multiresolution pyramid is the slowest of these hierarchical schemes due to its inherent sorting procedure. Finally, the global DCT is by far the slowest of all feature extraction techniques considered.

In table 1 we display the minimal number of feature values required to detect global attacks against the Lena image using multiresolution pyramids AV, ME, and DS. Note that the smallest number considered is $16^2 = 256$ which corresponds to 5 iterations of constructing approximations to the image. A larger entry in the table corresponds to higher robustness against the type of attack (desired or not) as indicated in the leftmost column. In this table we consider only the three most significant bitplanes.

We notice robustness to a certain extent against JPEG 2000 and JPEG compression. For example, J2K compression is not detected using 16^2 features up to Cr 6 using 2 bitplanes and up to Cr 14 using the MSB only when employing AV. JPEG compression is not even detected using 32^2 features up to Cr 3 and 16^2 features up to Cr 7.6 even when employing three bitplanes and AV. ME and DS are less robust against compression as compared to AV.

Now let us consider malicious modifications. On the one hand, sensitivity against Stirmark attacks and luminance modifications is high as being desired. For example, choosing AV as multiresolution pyramid and selecting the MSB of 16^2 feature values (i.e. 5 decompositions) is robust against all compression settings considered and reveals all global attacks discussed.

	AV			ME			DS		
Attack	3 BP	2 BP	MSB	3 BP	2 BP	MSB	3 BP	2 BP	MSB
J2K Cr 2	16^2	32^2	64^2	16^2	16^2	64^2	16^2	16^2	64^2
J2K Cr 6	16^2	32^2	32^2	16^2	16^2	32^2	16^2	16^2	16^2
J2K Cr 14	16^2	16^2	32^2	16^2	16^2	16^2	16^2	16^2	16^2
JPEG Cr 1.7	64^2	64^2	64^2	16^2	32^2	32^2	32^2	32^2	32^2
JPEG Cr 2.9	64^2	64^2	64^2	16^2	16^2	16^2	16^2	16^2	16^2
JPEG Cr 7.6	32^2	32^2	32^2	16^2	16^2	16^2	16^2	16^2	16^2
Fig. 1.b b=1	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2
Fig. 1.b b=2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2
Fig. 1.c i=1	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2
Fig. 1.c i=2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2
Lum +5	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2
Lum -5	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2	16^2

Table 1. Minimal number of feature values required to detect the global attacks: Multiresolution pyramids

On the other hand, the situation changes when we investigate the sensitivity against local attacks as displayed in table 2.

Figure	Fig. 1.a			Fig. 2.b		
	AV	ME	DS	AV	ME	DS
Full	16^2	32^2	128^2	16^2	16^2	32^2
4 BP	32^2	256^2	128^2	16^2	16^2	32^2

Table 2. Minimal number of feature values required to detect the local attacks: Multiresolution pyramids

Especially in case of the augmented Lena image we notice extremely low sensitivity against this attack. Choosing again AV as multiresolution pyramid and selecting the MSB of 16^2 feature values there is no way to detect this attack (even when using 4 bitplanes we already require 32^2 feature values to detect it). The situation is even worse regarding ME and DS. In case of the Escher image the result is not that bad but the sensitivity of multiresolution pyramid based hashing is comparable to that against compression which is of course not desirable.

In summary, multiresolution pyramid based hashing can be made robust against compression to some extent by using averaging (AV) as approximation method and employing the most significant bits only. In such a scenario, the sensitivity against the global attacks considered is satisfactory but it is not at all against the local ones.

When turning to bitplanes as a means to feature extraction it turns out immediately that there is no way to make such a scheme robust to compression at all. Even the slightest degradation is propagated to some extent to the MSB information causing the hash function to identify the compressed image as being tampered with. Therefore, it makes no sense to investigate the sensitivity against intentional modifications since the first goal, i.e. robustness to compression, has not been met.

Now we turn to the transform domain. In table 3 we display the results concerning the full frame DCT (DCT1). In contrast to the multiresolution pyramids, the number of feature values may be varied continously. Even when using full 8bpp precision for the feature values we still require 40 values to detect a J2K compression with Cr 14, the same is true for JPEG compression with Cr 13. Consequently we may state that robustness against compression may be achieved.

Attack	Full	7 BP	6 BP	5 BP	4 BP	3 BP	2 BP	MSB
J2K Cr 2	40	40	54	54	>200	>200	>200	>200
J2K Cr 10	40	40	40	40	162	>200	>200	>200
J2K Cr 14	40	40	40	40	79	79	174	>200
JPEG Cr 1.7	55	65	131	>200	>200	>200	>200	>200
JPEG Cr 6.1	54	54	65	65	65	>200	>200	>200
JPEG Cr 13	40	40	40	65	65	174	174	175
Fig. 1.a	40	40	40	43	43	72	**72**	175
Fig. 1.b b=1	4	4	4	4	40	40	40	40
Fig. 1.b b=2	4	4	4	4	4	4	4	4
Fig. 1.c i=1	40	40	40	43	43	43	43	54
Fig. 1.c i=2	40	40	40	41	41	41	41	54
Lum +5	2	2	2	2	2	2	2	2
Lum -5	2	2	2	2	2	2	2	2
Fig. 2.b	4	4	4	4	4	4	4	65

Table 3. Minimal number of feature values required to detect the attacks: DCT1

Sensitivity against intentional attacks, on the other hand, is satisfactory for all types of attacks. Especially for luminance modification, but also for most settings regarding Stirmark attack b and the modified Escher image the alterations are detected using 4 feature values or even less. Also for the remaining attacks sensitivity is always higher as against the strongest compression considered. As a consequence, we may define DCT1 based visual hash functions which are sensitive to all attacks considered but robust to moderate compression. As a concrete example, we could use 2 bitplanes of 80 feature values. In this case the number of feature values to detect J2K and JPEG compression is significantly higher (174 in either case of maximal compression) and therefore this hash function is also robust against even more severe compression. On the other hand, all considered attacks are revealed including the augmented Lena which is detected using 72 feature values (displayed boldface in the table).

Contrasting to the visual hash function based on DCT1, we could not achieve any robustness against compression for DCT2. Therefore, as it is the case for the bitplane approach, it makes no sense to investigate the sensitivity against intentional tampering. Note that both techniques, bitplanes and DCT2, produce a much higher number of feature values (even using their lowest parameter, i.e. MSB or one coefficient per block, respectively) as compared to the other schemes which definitely is the reason for their higher responsiveness.

Finally we focus onto the wavelet transform. Due to the equivalence to the Multiresolution pyramid AV (see previous section), the results are almost identical to this method and are therefore not discussed further.

4 Conclusion and Future Work

We have found that global DCT seems to be the most suitable feature extraction approach to base a robust visual hash function upon if robustness against moderate compression is a prerequisite for such a scheme. Although the computationally most demanding approach, the robustness against JPEG2000 and JPEG compression and the responsiveness to intentional global and local image alterations exhibited by the DCT based system are by far superior as compared to the competing wavelet transform and multiresolution pyramid based schemes. Visual hash functions based on block-based DCT and selective bitplane hashing have failed to provide robustness against compression.

In future work we will **add** to the qualititive approach based on the cryptographic hash function MD-5 ("tampered with or not") a quantitative one tailored to the DCT domain allowing to additionally rate the amount of image alteration in case of detected tampering and we will evaluate the security of the scheme.

References

[1] Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal G. Shamoon. Secure spread spectrum watermarking for multimedia. In *Proceedings of the IEEE International Conference on Image Processing, ICIP '97*, volume 6, pages 1673–1687, Santa Barbara, California, USA, October 1997.

[2] Jiri Fridrich. Visual hash for oblivious watermarking. In Ping Wah Wong and Edward J. Delp, editors, *Proceedings of IS&T/SPIE's 12th Annual Symposium, Electronic Imaging 2000: Security and Watermarking of Multimedia Content II*, volume 3971, San Jose, CA, USA, January 2000.

[3] Jiri Fridrich and Miroslav Goljan. Robust hash functions for digital watermarking. In *Proceedings of the IEEE International Conference on Information Technology: Coding and Computing*, Las Vegas, NV, USA, March 2000.

[4] T. Kalker, J. T. Oostveen, and J. Haitsma. Visual hashing of digital video: applications and techniques. In A.G. Tescher, editor, *Applications of Digital Image Processing XXIV*, volume 4472 of *Proceedings of SPIE*, San Diego, CA, USA, July 2001.

[5] P. Meerwald and A. Uhl. A survey of wavelet-domain watermarking algorithms. In Ping Wah Wong and Edward J. Delp, editors, *Proceedings of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents III*, volume 4314, San Jose, CA, USA, January 2001. SPIE.

[6] M. Kivanc Mihcak and Ramarathnan Venkatesan. A tool for robust audio information hiding: a perceptual audio hashing algorithm. In *Proceedings of the 4th Information Hiding Workshop '01*, Portland, OR, USA, April 2001.

[7] Fabien A. P. Petitcolas, Caroline Fontaine, Jana Dittmann, Martin Steinebach, and Nazim Fatès. Public automated web-based evaluation service for watermarking schemes: Stirmark benchmark. In *Proceedings of SPIE, Security and Watermarking of Multimedia Contents III*, volume 4314, San Jose, CA, USA, January 2001.

[8] R. Radhakrishnan, Z. Xiong, and N. D. Memom. Security of visual hash function. In Ping Wah Wong and Edward J. Delp, editors, *Proceedings of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents V*, volume 5020, Santa Clara, CA, USA, January 2003. SPIE.

[9] B. Schneier. *Applied cryptography (2nd edition): protocols, algorithms and source code in C.* Wiley Publishers, 1996.

[10] Ramarathnam Venkatesan, S.-M. Koon, Mariusz H. Jakubowski, and Pierre Moulin. Robust image hashing. In *Proceedings of the IEEE International Conference on Image Processing, ICIP '00*, Vancouver, Canada, September 2000.

[11] Ramarathnam Venkatesan and M. Kivanc Mihcak. New iterative geometric methods for robust perceptual image hashing. In *Proceedings of the Workshop on Security and Privacy in Digital Rights Management 2001*, Philadelphia, PA, USA, November 2001.