# Confusion/Diffusion Capabilities of Some Robust Hash Functions

Baris Coskun
Department of Electrical and Computer Engineering
Polytechnic University
Brooklyn, NY 11214
Email: baris@isis.poly.edu

Nasir Memon
Department of Computer and Information Science
Polytechnic University
Brooklyn, NY 11214
Email: memon@poly.edu

***Abstract*— Perceptual hash functions have been recently proposed as cryptographic primitives for multimedia security applications. However, many of these hash functions have been designed with signal processing robustness issues and have not addressed the key issues of confusion and diffusion that are central to the security of conventional hash functions. In this paper we give a definition for confusion and diffusion for perceptual hash functions and show how many common perceptual hash functions do not display desirable confusion/diffusion properties.**

## I. Introduction

Data integrity is one of the core requirements of secure systems. In the context of cryptography, the integrity or authenticity of data is provided by a cryptographic hash function using which the data is mapped to a short bit string called the hash value or a message digest. The authenticity of the data is then verified by simply recalculating the hash value from the data and comparing it to the attached hash value. In order to prevent tampering of the data, the hash value is protected by either signing the hash (resulting in a digital signature) or by using a secret key to compute or encrypt the hash (resulting in a message authentication code). In this work we focus on message authentication codes. A cryptographic hash function $h$ which is a member of MAC family generates a hash value $H$ from an arbitrary input $X$ and a secret key $K$. That is,

$$H = h(X, K)$$

Since the hash value $H$ itself is protected by the secrecy of a key, an adversary who would like to change the data needs to do it either in a way the hash value still remains the same, or guess the new valid hash value without knowledge of the secret key that was used in its computation. If either of these can be done, the receiver would regard the data as authentic, although it is not.

In order for a message authentication code to to be regarded as secure, it must be very hard to find the hash value $H$ without knowing the secret key $K$ and it must be very hard to find the secret key $K$ or the hash value of a new input $H' = h(X', K)$ even if very large set of input-hash $\{X_i, H_i = h(X_i, K)\}$ pairs are given. A hash function typically achieves these properties by its confusion/diffusion capabilities which are explained in detail in Section II. More detailed information about cryptographic hash functions and their security issues can be found in [1], [2], and [3].

The recent proliferation of multimedia content in digital form has led to the need for integrity mechanisms for such data. Traditional cryptographic hash function based mechanisms have been found lacking for this purpose due to the peculiar nature of multimedia data. Namely, with multimedia data, the same content can have many different digital representations. For example, an image can be represented in different formats and would be perceptually be the same although the two digital files would be entirely different. In view of the above problem researchers in the signal processing community have proposed the notion of *robust hash functions.* Robust hash functions are designed to produce the same hash value as long the input has not been perceptually modified. Whereas cryptographic hash functions are designed to generate a totally different hash value even if the input is changed by a single bit, robust hash functions are expected to change the hash value only if the input is perceptually changed. This property is often known as *robustness*. Although robust hash functions have been designed for different types of multimedia data, in this paper we restrict our attention to robust image hash functions. Specifically we present a new notion of confusion/diffusion for robust image hash functions. We show that some of the best known robust hash functions in the literature have poor confusion/diuffusion properties and cannot be considered secure for data integrity applications.

The rest of the paper is organized as follows: in Section II definitions of confusion/diffusion and their modifications for robust hash functions are presented. In order to clarify the perceptual difference concept, the notion of *'perceptual unit'* is introduced in Section III. In Section IV we evaluate the confusion/diffuison capabilities of three image hash functions and finally we expose the vulnerability of these functions against forgeries in Section V.

## II. Confusion/Diffusion And Robust Hash Functions

Since *confusion* and *diffusion* were first proposed by Shannon [4] in 1949, they have been extensively used to evaluate

the security of cryptographic systems. Confusion is basically defined as the concealment of the relation between the secret key and the cipher text. On the other hand, diffusion is regarded as the complexity of the relationship between the plain text and the cipher text. Although they were initially defined for encryption systems, they have also become the primary engineering design principle for cryptographic hash functions.

### A. Confusion/Diffusion for Cryptographic Hash Functions

In the context of hashing, *confusion* is *the complexity of the relation between the key and the hash value.* In other words for a hash function having good confusion property, given $X$, $K$ and $H = h(X, K)$, it is highly impractical to reveal the relation between $H = h(X, K)$ and $H' = h(X, K')$ where $K$ and $K'$ differ by even only a single bit. A hash function with good confusion capability generates completely different (statistically independent) hash values when the key is changed. Ideally, when the key is changed, each bit of the hash value either flips or remains same with probability of $\frac{1}{2}$. Hence when the key is changed even by a single bit, one should expect to observe that approximately half of the hash bits are flipped and the locations of the these flipped bits are also randomly distributed.

For hash functions which have relatively weak confusion capabilities, once can expect similar hash values for the same input when the key is slightly changed. More formally:

$$NHD\{h(X, K), h(X, K')\} < \epsilon$$

$$while, |K - K'| < \delta$$

where $NHD\{\}$ is the *Normalized Hamming Distance*, and $\epsilon$ , $\delta$ are some small numbers. That is to say, neighboring keys in the key-space produce very similar hash values, which makes the key-space virtually narrower and the hash function susceptible to brute-force (exhaustive search) type of attacks.

For an encryption function, *diffusion* is defined as the complexity of the relation between plain-text and cipher-text. However, for hash functions it can be altered to represent *the statistically irrelevance between the input bits and the hash value.* More formally, a hash function is said to have strong diffusion capability, if given $X, K, X'$ and $H = h(X, K)$, $H' = h(X', K)$, it is highly impractical to reveal the relation between $H$ and $H'$ where $X$ and $X'$ may differ by even only a single bit. For cryptographic hash functions, strong diffusion capability can be achieved by making each bit of the input affect each bit of the hash value. Thereby, any single bit change in the input would cause a drastic change in the hash value. This is often referred as the *avalanche effect* in the literature. Ideally one should expect approximately half of the hash bits having random locations are flipped when the input is changed even by a single bit. This is because the change in the input affects each bit of the hash value in the sense that each hash bit either flips or remains same with probability of $\frac{1}{2}$. In the case where the hash function lacks strong diffusion capabilities, an adversary could create collisions very easily since he could predict the response of the hash function to alterations in the input.

### B. Modified Confusion/Diffusion for Robust Hash Functions

Since the definition of robust hash functions is similar but not exactly the same as the cryptographic hash functions, a slightly modified confusion/diffusion concept is required. In robust hash functions, unlike the bitwise difference for cryptographic hash functions, the multimedia input is regarded as changed only if the underlying perceptual information is changed. For instance, similar or even the same hash values are expected after applying a robust hash function to an uncompressed image and its slightly compressed version whose bit representations are entirely different but the perceptual information is the same. Therefore one should expect a totally different hash only when the perceptual information is changed.

As mentioned in Section II-A, the difference of the input is related to diffusion only. Confusion is involved with the secret key, which has exactly the same definition as in the context of cryptographic hash functions. Therefore when a robust hash function is in question, only the definition of diffusion has to be modified. For a robust hash function we define diffusion to be *the irrelevance or complex relationship between the perceptual information of the input and the hash value.*

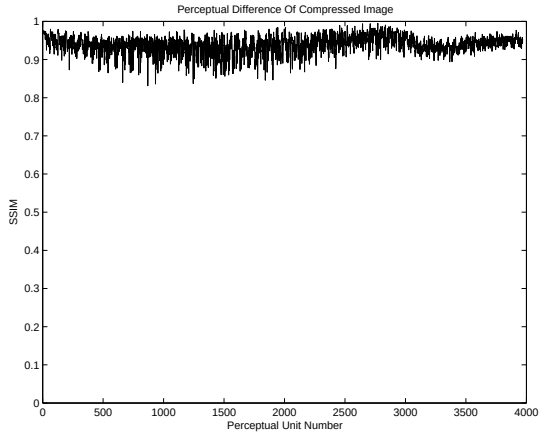
In order to identify perceptual change, the input can be regarded as a collection of perceptual units and the corresponding perceptual units are compared when comparing two different inputs. Particularly in the case of robust hash functions for images, if we neglect the geometrical alterations such as scaling and rotation, a perceptual unit can be defined as a small image block whose size is carefully decided to be sure that no significant perceptual change could take place without changing at least one perceptual unit. Since any change in one of the perceptual units could potentially alter the whole semantic information, any two images should be declared as perceptually same only if all corresponding pairs of the perceptual units are decided to be the same. For instance in a car image, if the digit '3' is transformed into the digit '8' on the plate, probably only a single perceptual unit will be different where the semantic information will be completely changed and the new image should be regarded as a different image. Therefore, any two same sized images can be perceptually compared by means of comparing corresponding perceptual units.

## III. Perceptual Unit and Perceptual Difference For Images

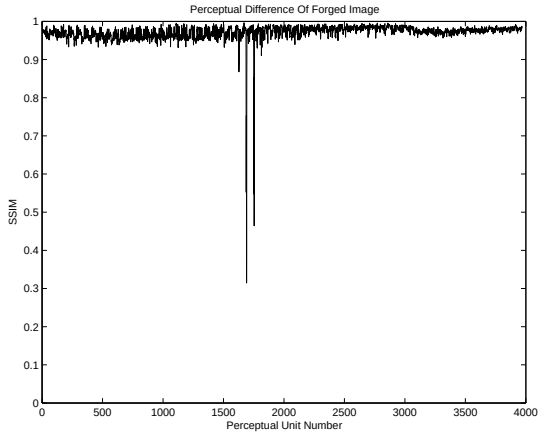
As mentioned in Section II-B, tiny perceptual differences could cause drastic semantic changes. Therefore perceptual similarity of two images should be analyzed block by block. If the perceptual difference is measured by comparing the images entirely at once, perceptually small but semantically significant changes probably will not be noticed by the comparison algorithm since significant portions of the images are perceptually identical. However, with carefully determination

(a) Compressed Image with JPEG-30

(b) Forged and Slightly Compressed Image




(c) Perceptual Difference Of Compressed Image




(d) Perceptual Difference Of Forged Image

Fig. 1. Illustration of perceptual comparison. An SSIM value for each perceptual unit pair is calculated. In 1(c) and 1(d) SSIM values between corresponding perceptual units of original image and modified images are plotted.

of the block size, it can be guaranteed that any perceptual difference will affect the significant portion of at least one block which will be declared as perceptually different. Hence, perceptual difference between two same sized images can be determined by the number of perceptual unit pairs which have the same location on two images but have been identified as different.

Perceptual units have to be overlapping blocks in order to eliminate the boundary problems and to ensure that small perceptual differences can be fully encapsulated within a single block. Otherwise, there would be a possibility that tiny perceptual differences located around the block boundaries might be shared by neighboring blocks causing block by block comparison algorithm to ignore those partial dissimilarities even if the whole difference is indeed much larger.

Deciding whether two perceptual units are similar or different can be done with the help of perceptual image quality measurement algorithms. In this work, we adopt Structural Similarity (SSIM) Index of Wang *et al.* [5], where a distance value is produced regarding human visual system (HVS). In SSIM, the perceptual similarity is calculated from cross correlations of luminance and contrast measurements which are obtained from statistical models. SSIM is bounded by 1 indicating perceptually identical blocks and goes to 0 as the perceptual information differs.

In the experiments as the perceptual units of 512x512 images, we choose 16x16 blocks which are overlapped with ratio of $\frac{1}{2}$ in both horizontal and vertical directions. We observe that 16x16 blocks are large enough to contain significant perceptual information and small enough to be affected by even tiny perceptual changes. In Figure 1 an illustration of perceptual difference measurement is presented. In order to observe the perceptual difference, two different modifications were applied on the original 'boat' image. First it is compressed by JPEG to a quality factor of 30. Although some visual distortions occur, it is expected that no perceptual difference would be

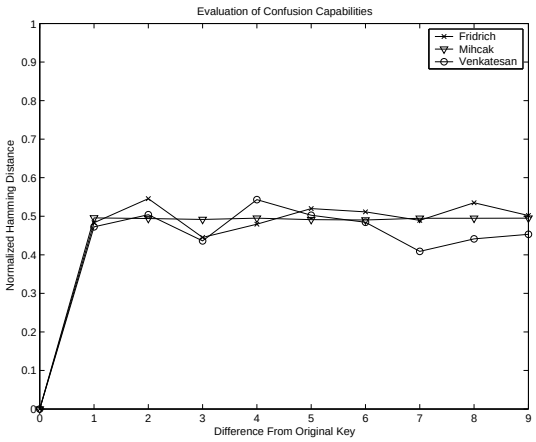



Fig. 2. Evaluation of confusion capabilities of robust hash functions.

observed. Then the last few letters of the script on the back of the boat were changed. Also the forged image was slightly compressed in order to observe the interference of forgery and compression. The compressed and forged images are shown at Figure 1(a) and Figure 1(b). The perceptual units of each figure were extracted and compared with the corresponding perceptual unit of the original image via SSIM. As expected the perceptual units of the compressed image did not differ too much from those of original image as can be seen in Figure 1(c). However, it is observed in Figure 1(d) that the SSIM value drastically drops at the perceptual units where the forgery takes place.

From the above example and the others that are not shown here we can chose a SSIM threshold around 0.8. That is, SSIM values below this threshold indicate perceptual difference. After deciding the threshold value for the example in Figure 1 we can say that there are no different perceptual units between the original and the compressed image whereas 9 out of 3969 perceptual units are different between the original and the forged image.

## IV. Evaluating Modified Confusion/Diffusion

In this section, we evaluate the *confusion* and *diffusion* capabilities of three well-known robust image hash functions. The first one is Fridrich's well known visual hash method [6] in which, 64x64 image blocks are projected onto pseudo-randomly generated smooth basis functions. The final hash value is a 1 bit quantization of these projection values where the threshold is determined carefully so that the number of "1"s and the number of "0"s are approximately equal. In our experiments we employed 50 random bases onto which each 64x64 image block was projected. Hence, at the end we generated 3200 bits of hash for each 512x512 image.

The second robust image hash function we investigated was Mihcak's robust hash [7], where binary representations of the images are produced from iterative geometric filters. These filters are designed to enhance the geometrically significant components by means of region growing. In Mihcak's method, first an iterative geometric filter is applied to a set of pseudo-randomly selected regions (can be overlapping) of the coarse subband of the image and then the bit representations of each region is pseudo-randomly permuted and concatenated to produce the final hash. In our experiments we pseudo-randomly selected 100 rectangles from each 512x512 image.

Finally we investigated the robust hash of Venkatesan *et al.* [8], where the hash is calculated from the statistics of wavelet coefficients. In this method, first the subbands are pseudo-randomly tiled into small subsections, and the mean and variance of coarse subband and detail subbands respectively are collected. Then a random quantization is applied to those statistics in order to obtain the final hash.

### A. Evaluation of Confusion

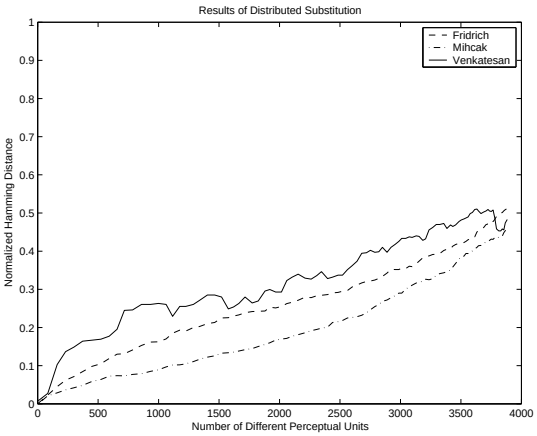
As we previously mentioned in Section II, *confusion* is related to the relation between the key and the hash value. Basically the hash function with strong confusion capability is expected to produce a statistically irrelevant hash value when the key value is changed even by a single bit. In order to investigate the confusion capabilities of robust hash functions, one should observe the change in the hash value along with the slightly changing key. The normalized hamming distance between the initial hash value and the hash value obtained by slightly changing the key is expected to be around 0.5, which roughly means the hash values are irrelevant. Results of such experiment is presented in Figure 2, where normalized hamming distances are recorded as the key values are slightly increased. It is observed that all three robust image hash functions achieve their maximum normalized hamming distance value,which is around 0.5, even right after a single bit is changed. Since the normalized hamming distance of 0.5 roughly represents statistical irrelevance, we can conclude that both hash functions have sufficient confusion capabilities.

### B. Evaluation of Diffusion

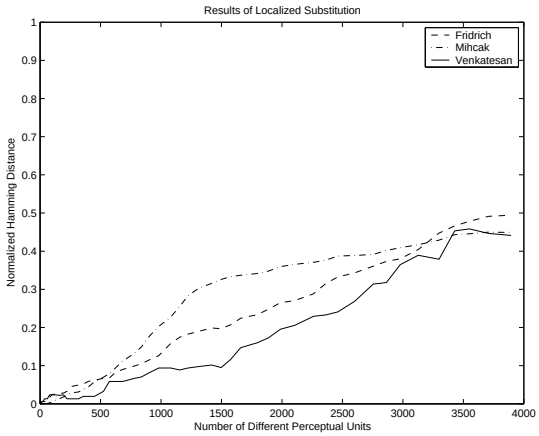
Since the notion of diffusion is based on the relationship between the input and the hash value, it can be evaluated by observing changes in the hash value as the input is being slightly changed. For cryptographical hash functions the input could be changed bit by bit, however in the case of robust hashing, slightly changing the input means changing the perceptual units one at a time. In order to change a perceptual unit of an image, we replace that unit with the corresponding unit of another image. Hence, as the number of changed perceptual units increased, the original image begins to look like another photographic image rather than a meaningless visual data. Since the robust image hash functions may use a relationship between neighboring pixels, we evaluate diffusion capabilities in two different schemes. In the first scheme the replaced perceptual units are selected randomly of which an example can be seen in Figure 3(a). An example of the second scheme is shown in Figure 3(b) where the replaced perceptual units are localized to a specific neighborhood. But in both schemes as the number of replaced perceptual units are increased, the Lena image begins to look like the Baboon image.

(a) Image obtained from Distributed Substitution. 689 of 3969 perceptual units are found to be different from the original Lena image

(b) Image obtained from Local Substitution. 262 of 3969 perceptual units are found to be different from the original Lena image




(c) Normalized Hamming Distances of Hash values under Distributed Substitution




(d) Normalized Hamming Distances of Hash values under Local Substitution

Fig. 3. Evaluation of diffusion capabilities of robust hash functions.

As mentioned in Section II-B and observed in Figure 4, even a change of a single perceptual unit could be a very significant semantic deceit. Therefore a reliable robust hash function should produce a statistically irrelevant hash value whenever the input is changed even by a single perceptual unit. Unfortunately, all of the hash functions reach the statistically irrelevance which corresponds to the normalized hamming distance of 0.5, only when all of the perceptual units are changed regardless of the replacement scheme. Hence we can conclude that all three hash functions have very weak diffusion capabilities under both localized and random replacement schemes.

Slowly increasing hamming distance for these robust hash functions is not surprising because they all focus on the significant perceptual information over the entire image and naturally cannot notice the tiny but dangerous modifications. Therefore, they cannot be used to prove the authenticity of images.

## V. Response of Robust Hash Functions Against Forgeries

The major problem of the hash functions lacking strong diffusion capabilities is that an adversary can easily generate collisions by carefully forging the input. Moreover in the context of robust hashing it is much easier to generate collisions because unlike cryptographic hash functions, robust hash functions are designed to tolerate some small modifications in order to be robust. Therefore, it is very likely that a careful forgery causing tiny perceptual change but very significant semantic change will not be noticed by robust hash functions. Two examples of such modifications are shown in Figure 4, where the script on the ”Boat” and the right eye of the ”Lena” are modified. In either of forged images no more than 4 perceptual units has been changed where there are total of

Fig. 4. Original (left) and forged (right) images.

3969 perceptual units in each image. Regarding the diffusion evaluation results summarized in Figure 3, these forgeries are expected to be unnoticed by the robust hash functions. Unfortunately this kind of behavior immediately suggests that, using robust hash functions having weak diffusion capability in authentication applications is very dangerous. As can be seen in Table I where the normalized hamming distances between the hash of original images and the hashes of forged and compressed images are presented, if any of these three robust hash functions were used in an authentication application, the forged images would be declared as more authentic than the JPEG-40 compressed images which have no different perceptual unit from the original images.

TABLE I
FORGERY VS. COMPRESSION

| Image | Fridrich Hash | Mihcak Hash | Venk. Hash |
|---|---|---|---|
| **Lena Forged** | **0.007** | **0.016** | **0.011** |
| Lena Compr. | 0.008 | 0.019 | 0.036 |
| **Boat Forged** | **0.004** | **0.014** | **0.021** |
| Boat Compr. | 0.013 | 0.021 | 0.016 |

## VI. CONCLUSION

We have presented a new definition of confusion/diffusion that can be used to measure the security of robust hash functions. Our definition is based on the notion of perceptual difference. We have evaluated the confusion/diffusion capabilities of three well-known robust image hash function and found them to be significantly lacking. We observed that all of the three robust hashing methods have excellent confusion capabilities. That is to say, if the secret key is changed even by a single bit, the resulting hash value will be completely different. This property makes the hash function more robust against exhaustive search for the secret key. However, all three robust hash functions we investigated do not have satisfactory diffusion capabilities meaning that the hash value remains similar as the perceptual information is slowly changed. Since an adversary can change the semantic information drastically even by changing few perceptual units, this weak diffusion property is very undesirable in authentication applications. In fact, we have created such perceptual changes and shown that all of the hash functions regard semantically changed images more authentic then their compressed versions.

## REFERENCES


[1] B. V. Rompay, "Analysis and design of cryptographic hash functions, mac algorithms and block ciphers," Ph.D. dissertation, Katholieke Universiteit Leuven, Faculteit Toegepaste Wetenschappen Departement Elektrotechniek, 2004.

[2] I. Damgard, "A design principle for hash functions," in *Crypto '89*, vol. 435, 1989, pp. 416–427.

[3] S. Lucks, "Design principles for iterated hash functions," 2004, lucks, Design Principles for Iterated Hash Functions, IACR preprint archive, http://eprint.iacr.org/2004/253.pdf, 2004.

[4] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, pp. 656–715, October 1949.

[5] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error measurement to structural similarity," *IEEE Transactions On Image Processing*, vol. 13, 2004.

[6] J. Fridrich, "Robust bit extraction from images," in *ICMCS '99, Florence, Italy*, June 1999.

[7] M. K. Mihcak and R. Venkatesan, "New iterative geometric methods for robust perceptual image hashing," in *Proceedings gf the Digital Rights Management Workshop*, November 2001.

[8] R. Venkatesan, S. Koon, M. Jakubowski, and P. Moulin, "Robust image hashing," in *Proc. IEEE Int. Conf. Image Processing*, 2000.

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

[10] R. Radhakrishnan, Z. Xiong, and N. D. Memon, "On the security of visual hash function," in *Proceedings of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents V, Santa Clara, CA, USA*, vol. 5020, January 2003.