A NEW FINGERPRINTING METHOD FOR DIGITAL IMAGES

V. Fotopoulos and A.N. Skodras

Electronics Laboratory
University of Patras
GR-26110 Patras, Greece

Computer Technology Institute
PO Box 1122
GR-26110 Patras, Greece

Email: vfotop1@ieee.org

ABSTRACT

The wide use of digital media during the past few years, has led to an increase of digital piracy and tampering. To deal with these problems, the concept of watermarking has been introduced. The image is being altered and depending on the method used, these alterations can be from very slight to quite noticeable. We propose here a new scheme that identifies image tampering without altering the original image. To achieve that, we extract some unique information from the image that we use for authentication.

1. INTRODUCTION

Digital media have conquered our lives during the past few years. There are great advantages in the use of digital imagery, video or sound, especially with the rapid growth of the Internet. An artist can make his work available through a website or distribute it easily by electronic means (email, ftp etc.). In parallel with these benefits, we have a severe raise in digital piracy. Digital copies are exact and easily made. So, it is an urgent need to find ways to protect the original artists' intellectual property rights.

A good solution to the problem seems to be digital watermarking. We define *watermark* as "a message intricately and imperceptibly interwoven within a digital medium". Several approaches for watermarking have been proposed. Some of them hide the information into the *spatial domain* [1], while others exploit the visual properties of the *frequency domain*, thus embedding the information by modifying the coefficients [2-3]. Frequency transforms like DCT, DFT, Wavelets etc. are usually employed. Watermarking systems can also be classified as *visible* or *invisible*. A classification of watermarking techniques can be found in [4]. Watermarking techniques have drawn serious attention because of the Internet digital highways. Digimark, Signum and other firms have developed commercial solutions where you can register a watermarking key or ID and then you can use it to watermark your own images. You can check the key at any time through the firm's server, thus verifying the image owner.

Another way to protect intellectual property rights is by using *fingerprints* [5-6]. These are special characteristics that can uniquely identify an object amongst others. Because of this property, they have been used in the past too many times for identification or authentication purposes. There are two main kinds of fingerprints, natural ones such as human fingerprints or the human iris, and the digital ones, such as PGP public keys and audio or video fingerprints for checking piracy. For example, legitimate owners of a digital media can be provided with a set of keys to assure legal ownership. But the creator or the distributor may use fingerprints in the original medium, which combined with the keys, may expose someone that shared his key with someone else. This is also known as the traitor-tracing problem.

Fingerprints can be used in several ways: as additive information to an image, or information that is deleted from them. In the case that they are inherent to an image, they can be used for recognition puprose. Of course, this case is of great importance for our purpose, since image authentication can be performed without altering the original image. Our system is a "recognition" type system, since we're trying to find such types of fingerprints, study and exploit their properties in a digital image authentication system. In Section 2, we discuss the requirements for such fingerprints while in Section 3 we present our algorithm in two steps, the extraction process in paragraph 3.1 and the verification in 3.2. In Section 4 we present our evaluation results followed by the conclusions in Section 5.

2. THE REQUIREMENTS

The problem of additive watermarking or fingerprinting implementations is that those schemes embed some information into the image. This embedding may or may not cause noticeable distortion to the image. A common truth is that the most significant the distortion is, the most robust the scheme is. Not noticeable distortion leads to watermarks or fingerprints that are more fragile.

What we propose here is an algorithm that uses the original image to 'extract' some kind of identity from it and store it for authentication purposes. The choice of the features that can be used for this task is actually the subject of image retrieval over large image databases. The selected features have to obey the following rules:

- uniquely characterize each image
- occupy significantly less storage space than the original image data.

For the selection of the features, in many image databases, the number of regions or texture are being used. But texture cannot uniquely identify an image and the number of regions can be altered as a result of image processing such as filtering, JPEG compression etc. So, since we need these features not only for archiving and retrieval of images but also to understand if a

copyrighted image has been tampered, we will also need a feature that can withstand most of the standard image processing tasks.

3. THE PROPOSED ALGORITHM

3.1 The feature extraction process

A region that possesses a wide frequency range can withstand a lot of changes. For example, if we try to identify fingerprint information in the high frequency content of an image, this could be very difficult after a simple lowpass filtering or during jpeg compression. Now, if we extend this to image blocks, then we have a very good approach to our original goal. We do not expect the location of such blocks in an image to change even after serious image alterations. But how can we find these blocks in an image? For this we propose the following algorithm:

- for each pixel of the image, we calculate the DCT in a 2N+1 sized square around it
- we calculate the variance of the DCT coefficients
- we place this variance into the corresponding pixel's position, into a matrix of the same size as the image, which however contains variances only, and
- we find the local minima of the variance image (using a small window of size N around each element)

The choice of N is an important part of the algorithm. N should be large enough to ensure that the region under investigation possesses a frequency range as wide as possible. On the other hand, large values of N leads to great computational burden and might also lead to overlap between different local minima neighborhoods, thus not allowing to any of them, except the lowest, to appear in the constellation matrix. As a good compromise to it, we selected N=7. The whole process is graphically depicted in Fig.1. The result of this algorithm is a bi-level picture of the same size as the image. It's full of *zeros* except for the points of the local minima in the variance matrix, where we have *ones*. Since this matrix is full of zeros (black points) with the exception of a few aces (white points), it resembles a constellation, so we'll call it a "constellation matrix" and we'll call the ones, "stars". This image is extremely easy to compress and needs only a few bytes to store.

3.2 The verification process

From the image verification point of view, we test the scheme as follows:

i. extract the corresponding constellation matrix from the image under investigation

ii. compare the corresponding matrix with those in the server's database

iii. the result is the number of stars found at the same positions in both images.

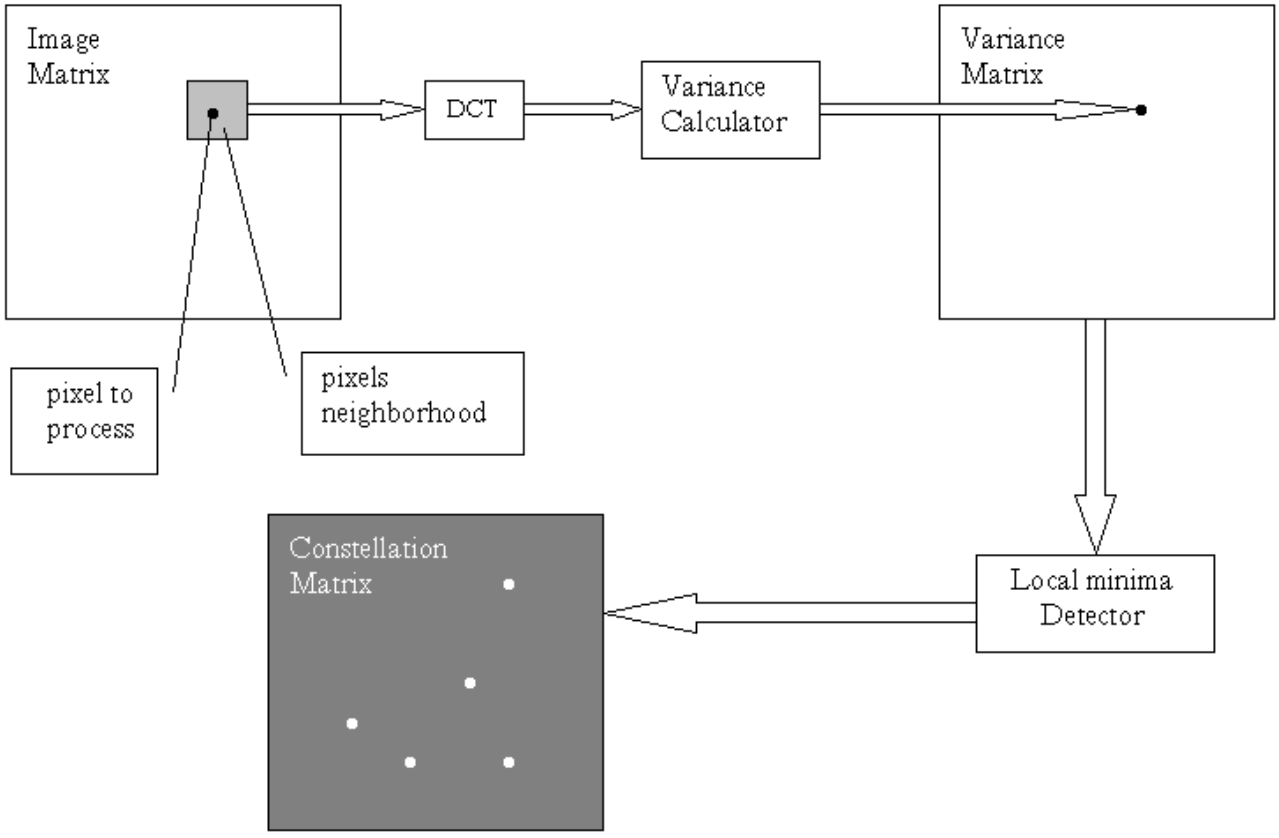

Figure 1 – Extraction of the fingerprint

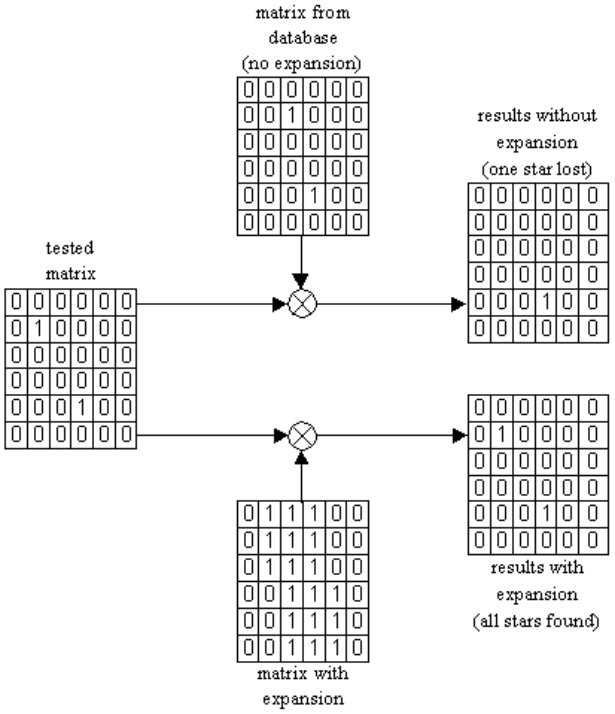

Figure 2 – Identification process

If more than 20% of the stars are found, we consider this as a positive "match". This comparison (step ii above) is easily obtained by means of a simple point to point multiplication (or logical AND) of the two matrices and a subsequent counting of the number of non-zero matrix elements. Any non-zero elements in the matrix under test that do not match with those in the original constellation matrix will be vanished after the multiplication. A problem that might occur is the fact that sometimes, after a possible image manipulation, the position of the stars might slightly change. It has been observed that they might move to a small distance from their original place. To ensure that we will still be able to locate them, we'll need to slightly "expand" the stars in the original matrix so that we may anticipate these small displacements. The expansion area is another point of discussion. Larger expansion could cover larger displacement, but may also lead to wrong identification. In order to prevent that, we expand each star only by ±1 around its initial position (Fig. 2). This has been proved valuable to significantly improve the detection results in tests such as gaussian noise addition or JPEG compression.

4. EVALUATION RESULTS

For the evaluation of the method, we have divided our tests into three parts. Each part has to do with the following requirements:

- unique identification
- compressibility
- robustness against image tampering

4.1 Unique identification

It is of vital importance for our purpose, that the constellation matrices of the images should not be identical. It would be very hard to distinguish between an altered image and a completely different one, if there where some stars at the same position in both matrices. To test the uniqueness of our feature, we have built a small database of images of the same size, and tested them according to the previous described method (without any expansion). The results where quite encouraging. Only in very few comparisons there was just one matching star, namely 1.9% of the cases, while in 98,1% the matrices where completely different. An example of the comparison procedure is depicted in Fig.3. On this diagram we compare an image with 20 others. For each case, the output of the comparison is the number of common stars, this is what we see on the vertical axis. It is clearly shown that there is only one pick in the searching procedure, when we compare the image with itself.

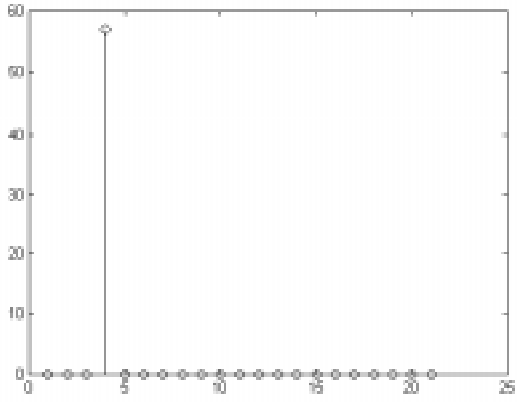

Figure 3 - Results of identity searching into the fingerprint database

4.2 Compressibility

Compression is very important if we want to built a fingerprint database. We tried to compress the constellation matrices for our test images and the resulting file yields a compression ratio of 536:1, for grayscale images. If we consider the same process for color images this ratio will be tripled since we will turn the RGB image to grayscale in order to extract the constellation matrix. We also have to mention that compression has been performed with Winzip, which means that there is additional information to the output file that is program specific. So, the true compression ratio is even higher.

4.3 Robustness against image tampering

We have tested the scheme for four images 'hotel', 'blonde', 'window' and 'caps' (Fig. 4) against five common image processing tasks, namely JPEG compression, blurring, sharpening, histogram equalization and addition of gaussian noise. All tasks have been performed with the default parameters of Adobe PhotoShop 5.0. Results are shown in Table I. Under the name of each image, is the number of the stars in the original

constellation matrix. Column A has the percentage of stars found without the expansion while column B percentages are right after the expansion. It is easily observed that the results for each image and attack, depend on its nature. For example 'hotel' is a complex scene and therefore the number of stars extracted is large enough, so the system performs very well. On the other hand the pale sky in 'caps' and the presence of texture, does not allow the appearance of an adequate number of stars in this image, thus lowering the performance of the system. Also the Gaussian noise addition attack has the worst rankings while blurring and sharpening do not seem to affect the scheme too much.

Table I – Results in various image attacks

	Hotel (109)		Window (64)		Blonde (47)		Caps (32)	
	A	B	A	B	A	B	A	B
JPEG Compression	53%	79%	48%	69%	43%	79%	13%	47%
Blurring	80%	98%	80%	91%	74%	96%	84%	94%
Sharpening	67%	84%	70%	94%	66%	83%	75%	88%
Histogram Equalization	72%	83%	41%	81%	64%	83%	41%	63%
Gaussian Noise	49%	70%	25%	72%	30%	72%	0%	25%

5. CONCLUSIONS

It has been proved that the position of local minima in the DCT domain can be used as an inherent fingerprint of an image. The characteristic is unique and it can't be altered by standard image manipulation procedures such as filtering, compression etc. It is our belief and it can be easily proven that the feature can withstand even more complex tampering such as cropping, rotation or scaling. Thus, it can be used by means of an Internet server for authentication of digital imagery.

6. REFERENCES

[1] M. Kutter, F. Jordan and F. Bossen, "Digital Signature of Color Images using Amplitude Modulation", Proc. of SPIE 3022, pp. 518-526, 1997

[2] D. Kundur and D. Hatzinakos, "Digital Watermarking for Telltale Tamper Proofing and Authentication", Proc. of IEEE, vol. 87, no.7, pp. 1167-1180, July 1999

[3] F. A. P. Petitcolas, R. J. Anderson and M. G. Kuhn, "Information Hiding – A survey", Proc. of IEEE, vol. 87, no.7, pp. 1062-1078, July 1999

[4] I. Cox, J. Kilian, F.T. Leighton, T. Shamoon, "A Secure, Robust Watermark for Multimedia", Workshop on Information hiding, Newton Institute, Univ. of Cambridge, May 1996

[5] V. Fotopoulos, A.N. Skodras, "Digital fingerprints and their applications in image databases", Computer Technology Institute, Technical Report, TR2000-04-05, April 2000

[6] S. Katzenbaisser, F. A. P. Petitcolas, "Information Hiding techniques for steganography and digital watermarking", ISBN 1-58053-035-4, Norwood MA, Artech House, 2000

Figure 4 – Test Images
From top to bottom: hotel, windows, caps, blonde