USING DIGITAL WATERMARKS WITH IMAGE SIGNATURES TO MITIGATE THE THREAT OF THE COPY ATTACK

John Barr
Digimarc Corporation
19801 SW 72nd Ave., Suite 100
Tualatin, OR 97062 USA
e-mail: jbarr@digimarc.com

Brett Bradley
Digimarc Corporation
19801 SW 72nd Ave., Suite 100.
Tualatin, OR 97062 USA
bbradley@digimarc.com

Brett T. Hannigan
Digimarc Corporation
19801 SW 72nd Ave., Suite 100.
Tualatin, OR 97062 USA
bhannigan@digimarc.com

ABSTRACT

In some applications, the utility of an image watermarking system is greatly reduced if an attacker is able to extract a watermark from a marked image and re-embed it into an unmarked image. This threat is known as the copy attack. In this paper, we develop an image signature scheme to be used with digital watermarks to create an image watermarking system that is more resistant to this attack. We describe the image signature algorithm in detail, and how it may be fused with a digital watermark. We then present preliminary results of our system using an image test set of highly correlated images.

1. INTRODUCTION

In the past few years, the use of digital watermarks has been proposed in a variety of applications, from broadcast monitoring to digital image tracking[1]. One proposed threat to these systems is the copy attack [2], which is performed by attempting to isolate the watermark from one piece of media, and inserting this estimated watermark into an un-watermarked piece.

In this paper we develop a system to mitigate the threat posed by a copy attack. We first develop an image signature routine which forms a short, binary signature based upon image characteristics. We then show how to use a digital watermark to ensure the image signature is robust to common image manipulations such as printing/scanning, rotation, and rescaling. Finally, we combine this image signature with a standard watermark to tie the watermark to the original image only.

2. IMAGE SIGNATURE ALGORITHM

2.1 Description of Image Signature Algorithm

The first step in diminishing the threat of this Copy Attack is to create an image signature algorithm. The image signature algorithm aims to distil the essential qualities of an image into a small sequence of bits so that any perceptually similar image will generate the same or close to the same signature. On the other hand, if two images are perceptually different, they are expected to produce very different signatures[3]. Of course the definitions of perceptually similar and different may be vague, but essentially we need an image signature algorithm which will produce similar bit sequences as long as the image examined is the same as the original image. If the image has been changed, then the signatures should vary greatly.

We begin by noting, as Fridrich points out in [4], that modification of the low frequency DCT values of an image typically results in significant visible changes to the original image. Similarly, the low frequency coefficients of the DCT are given the greatest priority in JPEG quantization tables [5]. Because of the importance of the low frequency components of the image, we propose to use the low frequency coefficients of the DCT to form our image signature.

Our image signature algorithm begins by converting the image into grayscale and then dividing the image into blocks of 128x128 pixels. A 128x128 DCT is performed on each block, and all but the lowest 16 x 16 coefficients are then discarded. Because the DC value simply denotes the average luminance value in the block, which may change depending on scanning conditions, this value is zeroed out and the median value of the 16x16 coefficients is found. Using this median value as a threshold, we convert the 256 coefficients into a sequence of bits by replacing a coefficient with a 1 if the value is above our threshold, and with a 0 if it is below the threshold. In this manner we create a 256 bit sequence, half 1's, half 0's, for each block, as shown in figure 1 below. We will refer to this as a block signature, and the collection of block signatures for the entire image as the image signature.

Because the DCT is sensitive to scaling, rotation, and translation, a change in image scale, orientation, or position may produce a dramatically different image signature. The presence of a watermark can help solve this problem. In our system, we embed a watermark that allows for easy geometric synchronization, similar to that described in [6]. This watermark is able to identify and measure geometric manipulations such as rotation, scale, and translation. Using this information, the watermark reader is able to reverse these transformations without knowledge of the original photo. With the image realigned to its proper scale and orientation during embedding, the image signature algorithm should produce a signature very similar to the original image signature.

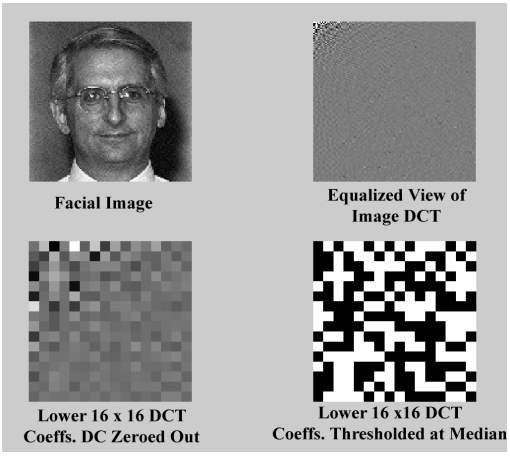

Figure 1. Graphical Depiction of Image Signature Algorithm

2.2 Results of Testing Image Signature Algorithm

In order for our image signature algorithm to successfully protect against the copy attack, image signatures for recaptured images must vary little from the image signature of the original digital image. Likewise, image signatures for images which are different from a given digital image, such as a different face, must vary greatly. To determine whether our proposed algorithm meets these conditions, we tested our algorithm with a subset of facial images from the FERET Facial Image Database test set, distributed by the National Institute of Standards and Technology [7]. We chose this test set because the heads-on facial images should be very highly correlated, since all heads will have a similar oval type boundary, two eyes, a nose, and a mouth.

The FERET Facial Image Database test set contains over 14,000 head shots, with head angles ranging from profile to heads-on. We pruned the image test set by eliminating all non-heads-on images. This forces the images in the test set to have even stronger correlation between images. These correlations are essential in testing the uniqueness of our image signatures, since different photos need to produce very different signature results. Finally, some of the images contained in the database are essentially duplicates, with only a digital change in dress color or a slight expression change. Since our system works on grayscale images, these photos contain almost the same image information of the same subject. We therefore eliminated these image from our test set. Our final test set contained 780 digital images.

After embedding all 780 digital images with a watermark, we then printed 90 of these images using an Atlantek Model 85 dye sublimation printer. Each digital image had pixel dimensions of 256 x 384 and was printed at 300 dpi. We then reacquired each image at 300 dpi using an HP Scanjet 5470c flatbed scanner. Scans with a Cardscan business card scanner, which provides slightly lower quality images, were also obtained. Due to space limitations, these results will not be presented here, but interested parties may contact the authors for further information. Since we used 128 x 128 blocks in calculating our image signatures, each image generated 2 x 3 blocks, for a total of 6 block signatures. These six block signatures form the full image signature. Using the watermark to properly align the images, we generated 6 block signatures for each of the 780 digital images, and each of the 90 scanned images. This gave us a total of 4680 digital block signatures and 540 scanned block signatures. Each block signature generated from a scan was then compared to every block signature from the digital collection. The number of bits that differed between the block signatures, or the Hamming distance, was recorded for each comparison. We then created two histograms: one showing the distribution of Hamming distances between the same digital and scanned blocks, and the other showing the distribution of Hamming distances between different digital and scanned blocks. Because each image signature will contain 128 ones and 128 zeros, the Hamming distance between two image signatures will be even. For simplicity's sake, we remove the zero samples at the odd locations in the distributions in figure 2 below.

Since each block signature consists of 256 bits, if block signatures of different blocks were fairly uncorrelated, we'd expect an average Hamming distance of 128. In practice, this is almost exactly what we see, with a mean Hamming distance of 127.5 (standard deviation of 8.56) between different blocks. Ideally the Hamming distance between the same block, digital version and re-acquired version, would be 0. In practice, variations are introduced in the printing and scanning process which make some differences inevitable. On the HP 5470 scanner, the mean Hamming distance between same blocks was 19.3 (standard deviation of 7.28). From these results it appears that our image signature algorithm is able to separate same blocks from different blocks fairly efficiently. It is important to note that in an actual system each photo is composed of multiple blocks, and therefore the image signature would be composed of many block signatures. The plots, however, show the distribution of examining just one block signature

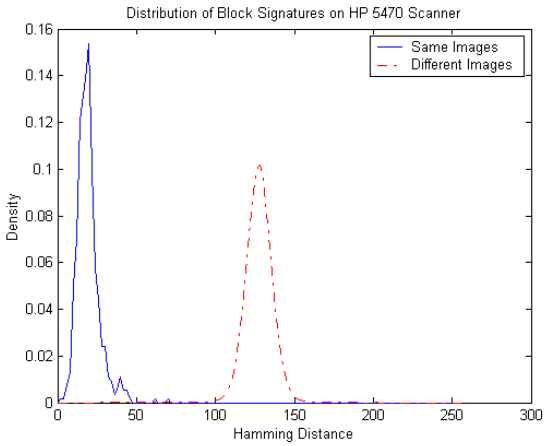

Figure 2. Distribution of Hamming Distances for Images Reacquired with the HP Scanjet 5470

3. COMBINING IMAGE SIGNATURE WITH WATERMARK

3.1 Embedding Block Signatures into Watermark

Creating an image signature algorithm which separates same and different blocks efficiently only solves half of the problem. We also need to be able to transmit the image signature of the original image to an inspector looking for counterfeits, so that he/she can compare it to the image signature of the image being inspected. To accomplish this, our system carries the image signature of the original digital image as part of the watermark, as described in [8]. Potentially, the watermark could use a variety of other methods to embed the information as well [9][10].

Normally, the watermark contains a message of length L bits that is error correction coded to produce a string of a*L bits, where $a \geq 1$ and represents the redundancy introduced by the code. The encoded bit pattern is then spread equally across non-overlapping MxM blocks of the image so that each bit is repeated $N = M^2/(aL)$ times. In our proposed system, we augment the a*L bits by 256 bits, to include our calculated block signature. Clearly in each MxM block there are now fewer repeated bit locations as there are more bits to embed. Instead of repeating the watermark message bits and the block signature bits equally, we choose to use 25% of the MxM bits to carry our block signature information and 75% of the bits to carry the watermark message.

Decreasing the number of bits used to carry our watermark message will of course cause a loss in SNR. Specifically the

Loss in SNR = -10 $\log_{10}$(1-Fraction of Bits Used for Image Signature) or -10 $\log_{10}$(1-.25) = 1.25 dB

If needed, this loss can be overcome by either increasing the watermark strength or decreasing the number of bits the watermark message carries.

By placing the block signature information into a portion of the watermark message bits, we introduce a second source of error. In addition to the differences found between block signatures of digital and the same print/scan blocks, we can also expect errors in faithfully extracting the block signature bits placed in the watermark. Perfect extraction of each block signature bit is very unlikely since there are many of them and each will receive a relatively small number of repetitions compared to one of the coded watermark message bits. We measure this new source of error in SNR by comparing the extracted bit sequence to that originally embedded.

To determine the likely distribution of SNR values, we measured the SNR of one watermark block in each of our scans from the HP scanner. The distribution obtained from these tests can be found in figure 3.

The overall effectiveness of our system is now determined by the interaction of two sources of error: the ability to accurately recalculate the original block signatures, and the ability to accurately extract each block signature embedded into the watermark. To depict this interaction, we define a value C, such that:

$$C = \frac{\sum_{i=1}^{256} Cis_i Eis_i}{|Cis||Eis|}$$

where Cis_i is the i^{th} calculated block signature bit, and Eis_i is the i^{th} extracted block signature bit. One could think of this value C as the normalized correlation between a received signal (extracted block signature) and a known binary signal (calculated block signature) except that some of the bits have been reversed in the recalculation of the block signatures.

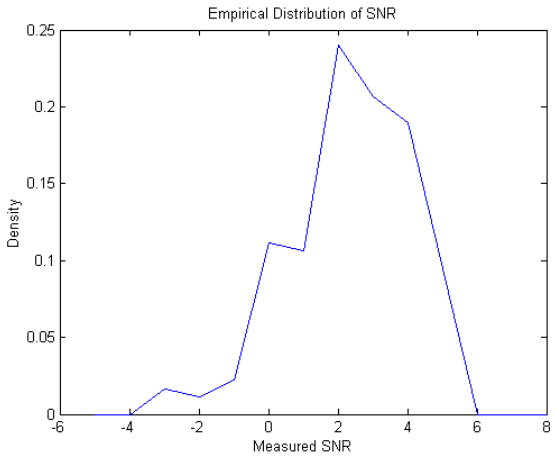

Figure 3. Distribution of SNR on HP Scanner

In order to determine what performance we could expect from our system as a whole, we ran a number of simulations. In these simulations, we first chose a value of SNR for our extracted block signature. Since in practice our extracted bits are approximately IID Gaussian, white Gaussian noise at this SNR level was pseudo-randomly generated and added to the pristine digital signature to simulate the extracted signature. Next, we used the distribution of Hamming distances between same images on the HP scanner calculated in section 3 to create our recalculated image signature, complete with bit errors. We found the normalized correlation between the extracted block signature and the recalculated block signature, and repeated this procedure a number of times. After plotting the distribution of C for the given SNR, we reset the SNR to a new value and repeated this process. Figure 4 has the results for six separate SNR values.

With low SNR between the original embedded block signature and the extracted block signature (values up to about -3 dB), we see that our distributions do overlap somewhat. However, even for the low case of SNR = -9dB (well outside the distribution of SNR given in figure 3), a full 90% of the copy distribution lies outside of 99.9% of the legitimate distribution. This implies that we could detect 90% of the copy with minimal chance of falsely calling a legitimate a counterfeit. As our SNR increases to more likely values as shown in figure 3, so does our ability to separate the two distributions.

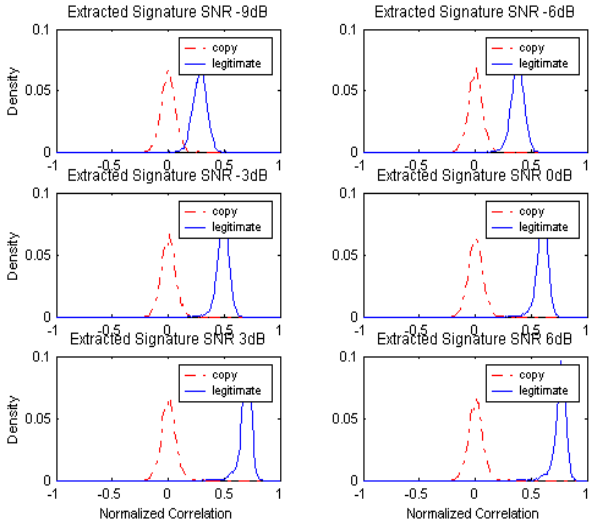

Figure 4. Distribution of C for Various Extracted Signature SNR Values on the HP 5470 Scanner

It should also be pointed out that the normalized correlation behavior is highly dependent upon extracted signature SNR. If one were to set thresholds based upon low SNR values, a high SNR counterfeit would be much more likely to be authenticated by our system. However, we can estimate the SNR of our extracted block signature bits by calculating the SNR of our watermark message. After decoding the watermark message, we calculate the SNR of this signal, and adjust this value to take into account the differences in repetition between the watermark message and block signature.

Once the estimated SNR of the extracted signature is obtained, we simply refer to the characteristic subplot to obtain the proper choice of threshold.

3.2 Combining Multiple Block Signatures into Image Signature

As noted previously, the full image signature is composed of multiple block signatures. By examining multiple block signatures, we should be able to better separate our same and different images. To simulate this, we recalculated the distribution of C assuming we used three of the six block signatures for each image. These results can be found in figure 5 below.

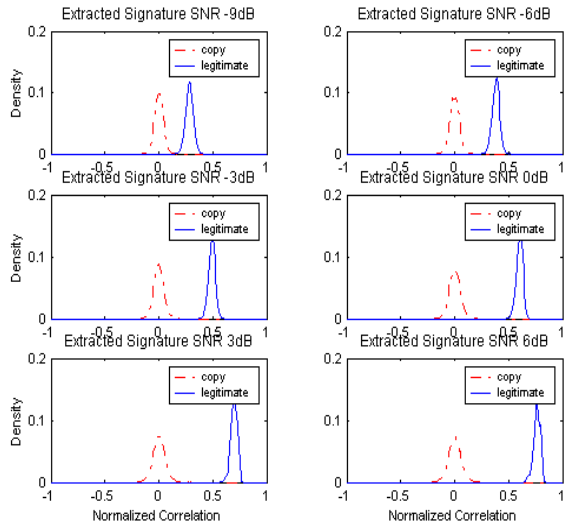

Figure 5. Distribution of C for Three Combined Block Signatures on the HP 5470 Scanner

In these distributions we see even clearer separation between the legitimate and counterfeit image signatures. Looking once again at a low value of SNR = -9dB, we find that with 3 block signatures the complete counterfeit distribution lies outside of 99.9% of the legitimate distribution. Within the limits of the experimental sample size, we should be able to detect 100% of the counterfeits with next to no chance of falsely calling a legitimate a counterfeit.

4. CONCLUSIONS

In this paper we presented a system which will mitigate the threat posed by the copy attack. We first developed an image signature algorithm which uses highly stable low frequency DCT coefficients to uniquely describe the image. This image signature was then combined with a standard image watermark, and embedded into the original image. If an attacker attempts to remove the watermark from this image and insert it into a new image, the image signature embedded in the watermark will not match the re-calculated image signature of the new image. The watermark also enables geometric synchronization, which allows us to automatically restore the image to its proper rotation, scale, and translation. This process, which in other systems must be performed by hand, is necessary to ensure the re-calculated image signature matches the image signature of the original digital image.

We also described how the image signature detector can be thought of as the normalized correlation between a received signal and a known binary signal. To determine whether the watermark contained in an image truly belongs to that image, we calculate the normalized correlation between the extracted image signature from the watermark and the re-calculated image signature. If this correlation value is above a certain value, we call the image legitimate, otherwise we label it a copy. The choice of our threshold value is directly dependent on the SNR of our extracted image signature. We illustrated that by calculating the SNR of our extracted watermark, we can estimate the SNR of our extracted image signature, and thus dynamically set the correlation threshold value despite varying environmental conditions.

ACKNOWLEGEMENTS

Portions of the research in this paper use the FERET database of facial images collected under the FERET program [7].

REFERENCES

[1] I. Cox, M. Miller, J. Bloom. *Digital Watermarking*. Morgan Kaufmann Publishers. 2002.

[2] M. Kutter, S. Voloshynovskiy, and A. Herrigel. The Watermark Copy Attack. *Proceedings of the SPIE, Security and Watermarking of Multimedia Contents II,* Volume 3971, pages 371-379. San Jose, CA, 2000.

[3] L. O'Gorman and I. Rabinovich. Secure Identification Documents Via Pattern Recognition and Public-Key Cryptography. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 20, No. 10. pages 1097 – 1102. October 1998.

[4] J. Fridrich. Robust Bit Extraction from Images. *Proceedings of ICMCS '99,* Volume 2, pages 536-540, Florence, Italy, 1999.

[5] K. Sayood. *Introduction to Data Compression*. Pages 392-394. Morgan Kaufmann Publishers, 2000.

[6] Digimarc's U.S. Patent Nos. 6,408,082, 5,862,260 and 5,636,292

[7] P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms," Image and Vision Computing J, Vol. 16, No. 5, pp 295-306, 1998.

[8] A. Alattar. "Smart images using Digimarc's watermarking technology." *Proceedings of the IS&T/SPIE's 12th International Symposium on Electronic Imaging.* Vol. 3971 No. 25. pp 264-273. January 2000,

[9] Barni M, Bartolini F, Cappellini V, Piva A. "A DCT-domain system for robust image watermarking." *Signal Processing*, Vol. 66, No. 3, pp 357-372, May 1998.

[10] I. Cox, J. Kilian, F.T. Leighton, T. Shamoon. "Secure spread spectrum watermarking for multimedia." *IEEE Transactions on Image Processing*. Vol. 6, No. 12. pp1673-1687, December 1997.