

Wavelet Based Watermarking Algorithms for Digital Images

Amir Schricker
amirs@bme.jhu.edu

Erica Richstad
erichstad@netscape.net

EE 520.646: Wavelets and Filter Banks

Final Project

The Johns Hopkins University

Spring 2002

1 Introduction

With the increasing importance of the Internet in information acquisition, the protection of copyrighted material has become a big challenge. The recent court battles between Napster and record companies have shown that artists and authors are more committed to protecting their work than ever before. The problem occurs in text, audio, video, and even still images. In response to this challenge, watermarking, a method of hiding data in the original signal, has gained popularity. The owner of the copyrighted material can thus place a personal stamp on his property and, furthermore, can detect illegal copies of his work.

Many watermarking algorithms have been proposed in the last decade, and in this paper we will examine the use of the multi-resolution discrete wavelet transform (DWT) as a method of hiding the watermarks. The advantage of this transform is that it provides the ability to zoom in on the energies of the transform coefficients to select the optimal subband for embedding the watermark.

A good watermarking system will fulfill several requirements: first, the watermark must be imperceptible after it has been embedded in the source. Secondly, since some individuals will certainly try to crack the watermark in order to obtain illegal copies, the watermark must be robust to basic types of signal processing operations, including compression, scaling, filtering, and the addition of noise.

In this paper we concentrate on the watermarking of still, digital images. The rest of the paper is organized as follows: Section 2 reviews some of the established watermarking methods. We present our algorithm in Section 3. Experimental results and analysis are presented in Section 4 to demonstrate the performance of the proposed algorithm. Discussions and conclusions follow in Sections 5 and 6.

2 Review of Previous Work

2.1 Method 1: Overview

The first method of watermarking that we investigated involved embedding a watermark by simply adding the watermark coefficients to the image subbands with the most energy. The embedding technique is implemented in the following manner: first, a wavelet decomposition is performed on the original image. Then based on the energies of the various subbands, an area is selected in

which to embed the watermark. Next, a qualified significant wavelet tree (QSWT) of the selected subband(s) is found. The QSWT involves identifying coefficients in the parent subband that are above a certain threshold¹, and then organizing them with their children coefficients in successive subbands. The QSWT and watermark coefficients are then sorted in descending order, and the same indices of the watermark and the QSWT are added together. This allows the watermark coefficients with the greatest amount of energy to be added to the subband coefficients with the most energy. Finally, an inverse discrete wavelet transform is performed to reconstruct the image.

The extraction procedure performs the steps of the embedding algorithm in reverse order. First, the original image and the watermarked image are decomposed into their wavelet coefficients and are subtracted from one another. Ideally, only the watermark coefficients will be left. We then used the original indices of the watermark coefficients to rearrange the watermark into its original order. Next, we found the correlation coefficient between the original watermark and the extracted watermark to determine the success of our algorithm.

2.2 Method 1: Results and Discussion

While this is a valid method of watermarking, there are several potential problems. Most importantly, the quality of the reconstructed image is visibly poor since there are noticeable artifacts. Fig. 1 displays the original watermark and the watermarked image and Table 1 lists some quantitative measures of this algorithm’s performance in an example. Next, much information was needed for reconstruction. Aside from the decomposition scheme, filter banks, and original watermark, all of which are generally needed for every watermark extraction technique, this method required the original image and knowledge of the embedded position of the watermark. Finally, there were too many variables to optimize; not only could the QSWT threshold be varied, but the strength of the watermark could also change, as could the amount of watermark that was added to the original image.



Figure 1: Demonstration of Method 1 algorithm. (a) Original binary watermark pattern (not to scale), (b) Original boat image, and (c) watermarked boat. Note the visible presence of watermark.

¹This threshold is variable and should be chosen for a given image and watermark. We chose the median coefficient value of the subband as the threshold for that subband.

Table 1: Performance of Method 1. Scaling of watermark pattern before addition to the original image, PSNR of the watermarked image, and the correlation coefficient ρ between the extracted watermark with the original.

Scaling	PSNR	ρ
1	27.7dB	0.74
0.1	29.3dB	0.62
0.01	29.3dB	0.60

2.3 Method 2: Overview

We felt that a good watermarking algorithm should create an image with a quality nearly equal to that of the original image and further, the extraction process should not require the original image or the embedded positions. As an alternative to the method described above, we investigated a second technique by embedding a watermark by completely replacing the coefficients in the selected subband with the watermark pattern.

Specifically, we first performed a multi-level wavelet decomposition on the original image and based on the contents of the various subbands, it was decided in which subband to embed the watermark. The watermarked pattern itself was next decomposed using a one-level decomposition, and the coefficients of the selected subband were then replaced by the coefficients of the watermark. The reason for embedding the DWT of the watermark instead of the watermark pattern itself was for security. This step acted as a further deterrent to detection, so that after the watermark was embedded, a hacker who might have accidentally decomposed the watermarked image would not be able to readily recognize the watermark. This step was thus a way of scrambling the watermark before embedding it.

Finally, an inverse DWT was performed to reconstruct the image. As before, the extraction process was the reverse of the embedding process. We peeled off the layers by first decomposing the watermarked image into its wavelet subbands. Then, the watermark was extracted from the subband and reconstructed. Finally, the reconstructed watermark was correlated with the original watermark to determine the success of the algorithm.

2.4 Method 2: Results and Discussion

Using the same watermark pattern and test image, we found that this method produced high image quality in the resulting image, yielding PSNR's of over 40dB. Detailed results of the watermarked images and the correlation coefficients of the watermarks are shown in Table 2. Furthermore, this method was partially robust to several commonly used image processing techniques. Though robust to JPEG compression and cropping, we also found that the addition of various types of noise to the watermarked image caused the reconstructed watermark to break down and to yield much lower correlation coefficients. Table 2 also summarizes the robustness results of this algorithm. Details of these image processing operations are discussed in Section 4.

3 Our Method

We believe that a good watermarking algorithm should be more robust to the addition of noise and be more secure against possible attempts to extract the watermark without the proper information.

Table 2: Performance of Method 2. PSNR of watermarked image and correlation coefficient of extracted watermark before and after various types of processing.

	PSNR	ρ	ρ_{JPEG}	ρ_{noise1}	ρ_{noise2}	ρ_{crop}
Boat	50.8dB	0.85	0.60	0.40	0.58	0.80
Lena	41.8dB	0.87	0.73	0.48	0.68	0.81

Therefore, our method is a variation on the second algorithm described above, with the following changes. Instead of using the same set of filters for every level of the discrete wavelet decomposition, we used different filters, and we redundantly embedded the watermark in another subband.

This algorithm was implemented as follows: for each level of wavelet decomposition, a set of filters was randomly selected from an existing database that contained many sets of filter banks. A specific set was (randomly) chosen, and the corresponding $H_0(z)$ and $H_1(z)$ filters were chosen for each level of decomposition. The corresponding synthesis filters, $F_0(z)$ and $F_1(z)$ were also chosen, and all four were stored in an array to be used during the extraction process. This scheme of selecting filters randomly was implemented in order to increase the security of the watermarking system: i.e. a hacker is less likely to be able to accurately locate and remove a watermark if many, random sets of filters are used to decompose/reconstruct the image.

In our implementation, the only filters we used in our database were the orthogonal Daubechies solutions. Our database consisted of the two D_4 solutions (minimum phase and maximum phase), four D_8 solutions (minimum phase, maximum phase, and the two mixed-phase solutions), and four D_{10} solutions (minimum phase, maximum phase, and two arbitrary mixed-phase solution).

The image was then decomposed with the DWT using these chosen filters. As before, a one-level wavelet transform was performed on the watermark to safeguard against watermark detection by simple inspection. Based on the energy levels of the different subbands, two or more areas were selected in which to embed the watermark, and the coefficients of these subbands were replaced with the coefficients of the watermark. If a subband was of a larger size than the watermark, the coefficients of the watermark were spread evenly throughout the subband rather than being embedded intact. The image was finally reconstructed with the inverse wavelet transform.

As usual, to extract the watermark, we performed the steps of the embedding process in reverse order, starting with the inverse DWT. The watermark was extracted from the proper subbands, reconstructed, and finally correlated with the original watermark. With this algorithm, extraction required the following information: the original watermark, subband position, and filter banks. Since different filters are used for each initial decomposition level, it is important to store this information to properly extract the watermark.

4 Results

To test the performance of our watermarking algorithm, we embedded the same watermark into several images using a variety of different parameters. Because this algorithm involves the random selection of filter banks, we first evaluated the performance of the individual filters by running the algorithm using only one set of filters for all levels of decomposition and reconstruction. Each class of filters, the D_4 , D_8 , D_{10} solutions, performed similarly with respect to the PSNR of the watermarked image, and the correlation of the recovered watermark with the original after various image processing techniques. To evaluate the overall performance of our algorithm, we let the

algorithm randomly select a class of filters and then, within that class, randomly select five sets of filter banks, one for each level of the DWT.

Each of the Daubechies solutions yielded watermarked images with PSNR's consistently over 40dB; Fig. 2 shows one of our watermarked images; the presence of the watermark should be minimal. Simply removing the watermarked image without further image processing provided recovered watermarks with correlations, ρ , greater than 0.90. The results of the overall algorithm (with the randomized filter selection being implemented) are listed in Table 3, including the results of robustness testing. The algorithm was run five times for each class of solution (Daubechies 4, 8, and 10) and five times with the randomized filter selection, and the mean values are shown. Fig. 3 shows the recovered watermark after no processing and after compression of the watermarked image.



Figure 2: Demonstration of images using Method 2 algorithm. (a) Original Lena image, (b) water-marked Lena. PSNR = 43.4dB



Figure 3: Recovered watermarks using our algorithm. (a) Watermark extracted from unprocessed image and (b) watermark extracted from JPEG compressed image.

We tested the robustness against several common image processing techniques by performing one of the following on the watermarked image. The image was compressed with JPEG using the default level of compression in Matlab. Robustness to noise was tested with two different kinds of noise: zero-mean Gaussian noise with a variance of 0.01, and salt and pepper noise with 0.01 density. To crop the images, we selected the upper 480x480 pixels and eliminated the remaining pixels.

To further examine the robustness of our algorithm against image cropping, we cropped off various 480x480 pixel blocks of the watermarked image, with the upper-left corner of each cropped image always lying along the diagonal of the original image. In Fig. 4, correlations of the extracted watermark are plotted as a function of the location of the cropped image relative to the uncropped one.

Table 3: Performance of our algorithm, using randomly selected filters. PSNR of watermarked image and correlation coefficient of extracted watermark before and after various types of processing. Tests were performed five times, and the averages are shown. Data are reported as *mean* \pm *SD*.

	PSNR	ρ	ρ_{JPEG}	ρ_{noise1}	ρ_{noise2}	ρ_{crop}
Boat	42.9 \pm 3.3dB	0.97 \pm 0.01	0.66 \pm 0.15	0.55 \pm 0.08	0.78 \pm 0.03	0.96 \pm 0.01
Lena	43.4 \pm 2.3dB	0.95 \pm 0.04	0.74 \pm 0.17	0.61 \pm 0.05	0.82 \pm 0.04	0.95 \pm 0.04

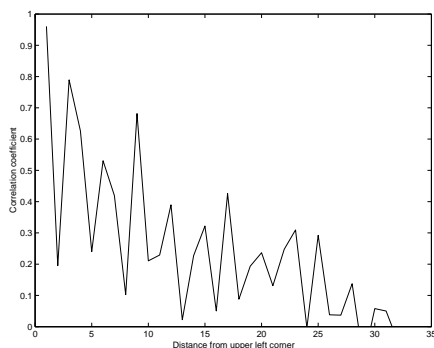


Figure 4: Correlation coefficient of the watermark extracted from the cropped image as a function of location of the cropping. All cropped images were 480x480 pixels. The location measures the distance from the upper-left corner of the crop to the upper-left corner of the uncropped image.

5 Discussion

As we had hoped, redundantly embedding the watermark greatly improved robustness to noise and also improves the overall performance of the algorithm. The image quality of the resulting watermarked images were very high, all of them achieving PSNR’s of over 40dB. This result was qualitatively verified by visual inspection of the watermarked images, since the watermarks’ presence were barely, if at all, noticeable.

More importantly, the watermarked images generated by our algorithm proved very robust against compression, noise, and cropping; the correlations were consistently higher than any of the other two methods mentioned earlier in this paper. Using a threshold correlation coefficient of $\rho = 0.40$, a reasonable threshold² determined by Wang *et al* [2], all of these recovered watermarks would qualify as being detected. Considering that the most prevalent image processing operation is likely that of JPEG compression, this algorithm would be an effective one to use in practical situations.

However, there were several limitations with this algorithm. After extensive test watermarks were embedded in images with a wide variety of filters for the decomposition/reconstruction processes, it appeared that when the filter lengths varied from one level to the next, the quality of the extracted watermark was very poor. The image quality decreased from approximately a PSNR = 40dB, using same-length filters, to approximately a PSNR = 28dB, using filters of mixed lengths. It was for this

²The authors of [2] analytically calculated that with a threshold correlation of $\rho = 0.40$, the probability of an arbitrary pattern passing the detection process, for a given set of filter banks, is on the order of 10^{-11} .

reason that we first selected a specific class of filters, all of which had the same lengths, and then randomly selected the individual filters from within that class. Further investigation will be needed to receive successful results with an algorithm that uses a completely randomized choice of filters.

Another limitation was the ability to get satisfactory correlation coefficients when the watermarked image was cropped far away from the upper-left corner. Because of the decomposition of the DWT and the subband location in which we embedded our watermark patterns, the upper-left portion of the images would contain more of the watermark. Thus, increasing the distance of the cropped image from that corner of the image resulted in dramatically decreased watermark correlations; this fact is evidenced in Fig. 4. Investigating this undesirable effect should be another area of future attention with this algorithm.

6 Conclusion

Several watermarking algorithms based on the discrete wavelet transform (DWT) have been described for watermarking images. While Methods 1 and 2 seemed to work in general, they both had minor and significant shortcomings. We have presented an algorithm that we believe is superior to those two methods, as an effective method of watermarking still images. It successfully both embedded and extracted watermarks and also satisfied the requirements set out in the introduction, namely those of robustness and imperceptibility. Also, given the randomized nature of filter selection, this method of watermarking should provide adequate security against attempts to extract or remove the watermark without prior knowledge of the details of the algorithm.

With proper attention given to the issues mentioned in this paper, watermarking might prove to be a valuable technique in copyright protection and other situations where proof of ownership might be hard to determine otherwise. We look forward to seeing their widespread use.

References

- [1] M. Hsieh, D. Tseng, Y. Huang, "Hiding Digital Watermarks Using Multiresolution Wavelet Transform," *IEEE Trans. Industrial Electron.*, vol 48, pp. 875-882, 2001.
- [2] Y. Wang, J. Doherty, R. Van Dyck, "A Wavelet-Based Watermarking Algorithm for Ownership Verification of Digital Images," *IEEE Trans. Image Processing*, vol 11, pp. 77-88, 2002.
- [3] G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Wellesley: Wellesley-Cambridge Press, 1996.

