Transform Domain Watermarking: Adaptive Selection of the Watermark’s Position and Length‡

V. Fotopoulos a,b and A. N. Skodras b,c

a Electronics Laboratory, University of Patras, GR-26110 Patras, Greece
b Research Academic Computer Technology Institute, GR-26221 Patras, Greece
c Hellenic Open University, School of Sciences & Technology, GR-26222 Patras, Greece

ABSTRACT

Transform domain methods have dominated the watermarking field from its early stages. In these methods some coefficients are selected and modified according to certain rules. The two most important numbers in this process are the length and the position of the watermark. These are usually heuristically chosen. In order to handle this problem, an adaptive scheme for the selection of the proper coefficients is analysed in the present communication.

1. INTRODUCTION

The work of Cox et al 1 has signalled the opening of the watermarking field. Many people have followed the same trails using frequency domain watermarking with slight variations. Methods, which are compatible to this scheme, in general include the following steps:

- Determine a frequency transform
- Perform transform
- Select transformed coefficients
- Alter selected coefficients according to some rule
- Inverse transform
- Save watermarked image

A variety of transforms have been used in such applications like DCT 2,3, DFT 4, DWT 5,6 or others 7,8,9. The application of the transform could be full frame or in smaller image blocks. The transformed coefficients are then reordered in vector form and in order of frequency. First coefficients represent low frequencies while the last correspond to high frequencies. The reordering scheme has to do with the transform used. For the DCT case, zig-zag scanning is performed and this is quite understandable since the low frequency (and highest energy coefficients) are concentrated at the upper left corner of the transformed block. Other transforms concentrate energy in different regions of the transform block and thus scanning should be different. This scheme performs generally well but there are two parameters that are heuristically chosen. Usually from the ordered coefficients vector, first \( M \) are skipped and next \( L \) are selected for watermarking. The selection of these parameters until now was based on the following assumptions:

1. Parameter \( M \) should not be too low because changing the first coefficients of the vector (that correspond to low frequencies) would heavily degrade the image.
2. \( M \) should not be too large because then we would be using high frequencies. Altering these coefficients would produce visible artefacts and would be vulnerable to low pass filtering and similar processes.
3. The \( L \) parameter should be sufficiently large (because of statistical hypothesis implied) but not too large in order to avoid image degradation.

There is no sufficient theoretic background on the selection of these coefficients. It is observed that even now, almost a decade after the watermarking boom, the selections are the same as those used in the first two years. In this

‡ This work has been funded by a GSRT grant (ΠΑΒΕ Τ 2000 – 00ΒΕ 363)
communication, we are not going to present a theoretic investigation, but rather an efficient calculation algorithm that can be used to maximize the efficiency of the detection.

2. ADAPTATION OF L AND M

In watermarking literature, common values of L and M are usually drawn in the range of 5000-10000. Using a large value form L has mainly to do with statistical reasons but at the same time produces more distortion. As M is concerned, the number of skipped coefficients, the selection depends on the desired frequency band for the watermark embedding. If low frequencies are used, then M is usually selected in the range 1000-2000. In that case, very good robustness is achieved in exchange for stronger image degradation compared to the case of middle frequencies where M is selected in the range 5000-10000. All these numbers correspond to 512x512 images which produce 262144 transform coefficients (full frame transform). For different image sizes, these numbers are proportionally changed.

To perform the selection of these parameters in an adaptive way, some decision rule is needed. This rule is the maximization of the similarity function used in the detection process. This function depends on L and M and that’s why, from now on we are going to refer to it as \( n_{sim}(M,L) \). To perform parametric analysis, for different pairs of M and L, we are going to watermark the selected coefficients and test the output of the detector. The output of the detection function will be visualized as a 3D graph in which the main task is the localization of the maximum value for an optimum pair of L and M for watermarking the selected image.

If an image is of size \( R \) rows by \( C \) columns, the number of the available transform coefficients will be \( N=R \times C \). The initial values for M and L are chosen to be 0.004N and 0.002N respectively. The calculations are performed in increments

![Diagram](image)

**Figure 1** M and L selection algorithm
of 64 for each parameter, for profound computational reasons and continue until $M=0.016N$ and $L=0.02N$. Figure 1 depicts the algorithm used.

To draw some early conclusions, the method was applied on the grayscale image “cameraman” of size 256x256. The 3D mesh produced, is presented in Figure 2, where $M$ runs on the left axis and $L$ on the right. It is easily observed that the detection decays while $L$ is increasing. Especially for $L>10000$ the decay is too fast. As $M$ is concerned (keeping $L$ constant), we have some distinct peaks. Which one should be chosen? Being strict, the largest value corresponds to the lower value of $M$ but by choosing the larger value, although the detection would be only a bit lower, one would benefit from better PSNR and visual result.

To test the scheme in another way, we used it on four test images with two different frequency transforms, DCT and Hartley. The reason behind this is to examine the relationship between the transform and the detection obtained. It is obvious that the scheme does not depend on the transformed selection; it can be applied with no problem in any frequency domain method. It is a question though to examine how different transforms affect the overall scheme’s performance. Two transforms are checked here, the DCT and Hartley respectively. The results for the $L$ and $M$ pairs that produce the maximum detection are given in Table 1, while the 3D meshes are shown in Figures 3 and 4. Results show that $M$ should be as low as possible. The value of $M$ that produced the maximum output was the starting one (1024), for all combinations of images and transforms. Contrary to $M$, the $L$ parameter varies significantly. It is interesting to note that for the “house” image $L$ is less than half compared to the other images. The “Girl” image requires the most coefficients in the case of DCT but if Hartley transform is used, it is “Lena” that requires the most. In general it is observed that for DCT, more coefficients are needed compared to Hartley.

<table>
<thead>
<tr>
<th>Image</th>
<th>DCT</th>
<th></th>
<th>Hartley</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>1024</td>
<td>1984</td>
<td>1024</td>
<td>1600</td>
</tr>
<tr>
<td>House</td>
<td>1024</td>
<td>832</td>
<td>1024</td>
<td>576</td>
</tr>
<tr>
<td>Girl</td>
<td>1024</td>
<td>3840</td>
<td>1024</td>
<td>1024</td>
</tr>
<tr>
<td>Baboon</td>
<td>1024</td>
<td>3648</td>
<td>1024</td>
<td>1280</td>
</tr>
</tbody>
</table>

3. DETECTION IMPROVEMENT IN PRACTICE

After those preliminary tests, the next logical step towards the evaluation of the scheme is to incorporate the selection algorithm into a common transform domain watermarking scheme. The scheme selected was based on the Hartley transform. The reason behind this was to exhibit the improvement achieved with the adaptation of $L$ and $M$ for a...
transform that does not perform as good as others like DCT or Wavelets. Corresponding results are given in Table 2. The image used is “Lena” and the attacks are the non-geometric attacks of the “CheckMark” software \(^{10}\), a very efficient benchmarking tool for watermarking applications.

**Figure 3**  Selecting parameters for DCT

**Figure 4**  Selecting parameters for Hartley
Table 2  Hartley transform based watermarking (with/without adaptation)

<table>
<thead>
<tr>
<th>Attack (number)</th>
<th>Without Adaptation (heuristically chosen parameters)</th>
<th>With Adaptation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Number of detections</td>
<td>Percentage</td>
</tr>
<tr>
<td>reSample(1)</td>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td>Filtering(3)</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>ColorReduce(2)</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td>JPEG(12)</td>
<td>11</td>
<td>92%</td>
</tr>
<tr>
<td>Wavelet(10)</td>
<td>9</td>
<td>90%</td>
</tr>
<tr>
<td>MAP(6)</td>
<td>5</td>
<td>83%</td>
</tr>
<tr>
<td>ML(7)</td>
<td>3</td>
<td>43%</td>
</tr>
<tr>
<td>Remodulation(4)</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Copy(1)</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td><strong>Total number of attacks: 46</strong></td>
<td><strong>Successful detections: 34</strong></td>
<td><strong>Total success percentage: 74%</strong></td>
</tr>
</tbody>
</table>

The performed tests, although they exclude geometrical attacks, are extremely hard. Filtering includes various lowpass gaussian-windowed, sharpening, median and more. Compression with JPEG goes down to a quality factor of 10 while wavelet compression can go down to 0.1 bpp. Denoising with Wiener filtering, hard and soft thresholding, denoising combined with remodulation are also included. The “Copy” attack involves estimating the watermark by means of Wiener filtering and copying onto another one. For more information on the nature and parameters of the attacks, one can refer to 11. It is observed from the table that without the adaptive scheme assistance, the Hartley transform based scheme fails for the highest compression rates (both JPEG and Wavelets), the Remodulation and Copy attacks. It fails even worst in the maximum likelihood estimation attacks (ML) where it misses 4 out of 7. From this point of view, it seems that the adaptive selection of parameters produced great improvement, as the scheme that originally performed successful in only 74% of the cases, now succeeds always. There’s not even one case that the enhanced method fails to identify the watermark. Although results are taken for one image only, the improvement is clearly evident.

4. CONCLUSIONS – FUTURE DIRECTIONS

The Hartley transform has been selected intentionally to show how, the results of a method based on a transform not known to perform well in watermarking applications, can be boosted significantly by using the proposed algorithm. Not only Hartley but all schemes that perform watermarking this way (frequency transform-selection of coefficients-multiplication by pseudo-random noise sequence) can be benefited from the proposed technique, the scheme is independent of the transform used. Some questions though remain to be answered.

One is the possibility that the false positive errors are increased. The answer to this question cannot easily be given because it has to do with the detection threshold selection (fixed, dynamic or other). In any case, a lot more experiments in a larger image set are needed to quantify this percentage. Another important issue has to do with the robustness vs. quality trade off. We’ve seen that there might be two or more easily distinguishable peaks in the 3D surface. These correspond to local maxima of the nsim function. Until now we’ve used the total maximum point that happened to correspond to the lowest pair of values. This was the best solution in terms of detectors reliability-robustness but perhaps it would be hard to sacrifice a bit of robustness to select a lower peak that performs much better in terms of quality (larger PSNR values). A quality threshold could be included into the scheme, something like the just noticeable difference (JND), mean absolute
error (MAE), mean squared error (MSE), root mean square error (RMS) or other. The algorithm then should be changed as follows:

- Transform input image
- Produce 3D mesh for the nsim function
- For each pair of L,M values, find the corresponding quality metric
- Divide each value of the nsim function with the corresponding quality metric
- Find total maximum of the product function

**Figure 5** Scheme improvement by quality metric inclusion

Additionally, weights $w_1$ and $w_2$ could be introduced for the division of the nsim and the metric (Figure 5). Another method could probably perform the inverse transform for each pair of L, M values in order to compute the JND between the original and watermarked image. Of course such an addition would result in significant computational cost.

A query of “adaptive watermarking” in the Google search engine yields almost 6,500 hits. Interesting ideas have been proposed, some of them coming from different research fields like genetic algorithms for example\textsuperscript{12}. None of them though addresses the adaptation problem in the same way to the proposed method. The algorithm presented in this communication is brute force and suboptimal because it does not account for all possible L,M pairs. For a typical 512x512 sized image, 3,700 different pairs are examined. The performance boost achieved is impressive. Of course, the computational cost is discouraging for real time applications. On the other hand, if robustness is the main issue, like in a digital rights management (DRM) system, such improvement is highly desirable. This fact, along with the transform selection independence makes this algorithm a useful add on for many watermarking applications. Of course there is still space for improvement.

**REFERENCES**