CONTENT-BASED WATERMARKING ROBUST TO BOTH AFFINE AND JPEG DISTORTIONS

Chad Coats and Xiaojun Qi

Computer Science Department, Utah State University, Logan, UT

Abstract - The problem of reliably detecting watermarks in digital images has been given much examination and experimentation. Many approaches to date have been focused on repelling attacks along one of two main branches: either JPEG compression or affine distortions. This paper presents a blind dual-approach to combat both types of attacks. In order to maintain and determine watermark presence against JPEG compression, a binary watermark string is interleaved and embedded in the discrete wavelet domain. A second watermark is used to ally the effects of geometric (i.e., rotation, scale) attacks. Significant feature points are extracted from the image using a controlled Harris corner detector and subsequently used to compute the Delaunay tessellation. The watermark is then embedded separately into each of the triangles produced by this tessellation. The success of the extraction process is based on correlation values calculated from each of the triangles. If the image is distorted via JPEG compression to the extent that detection in the triangles is not possible, we then attempt to retrieve the watermark from the wavelet domain.

1. INTRODUCTION

The field of digital image watermarking has developed in response to several stimuli. Advances in computer hardware have made the personal computer an increasingly affordable feature in many homes. The explosion of internet technology has provided the world with a vast virtual marketplace of products and ideas. Digital content such as images and video is easily shared amongst a potentially huge number of people. Almost just as easily that content can be edited or modified. Once altered, that media itself may likewise be utilized or shared by any number of users. For copyright protection and content authentication purposes, the goal of a watermarking scheme is to embed visually innocuous data (the watermark) into the media. Additionally, because of the relative ease with which images may be altered, these schemes must also be able to extract the mark even after changes have been made.

Of the many previously presented schemes, one way we may classify a watermarking algorithm is according to its dependence on the presence of the original image for extraction. An algorithm that does not rely on the properties of the original image is said to be blind. Contrarily, an approach that does require the original image is not. Our approach implements a blind watermarking scheme. Furthermore, we will consider two basic types of distortions that can be introduced to images. One is compression. These attacks are the result of using a compression algorithm, such as JPEG, to alter the image. Because of the variety of settings that can be modified with such algorithms, the resultant images may be significantly different from the original. The second is geometric. These attacks distort the image spatially, and include such operations as translation, rotation, and scaling.

Our approach attempts to embed a watermark that is robust to both of these types of attacks, and to blindly extract the mark from the image. As an overview, the embedding process is as follows. First, we embed a key-generated binary watermark into the approximation coefficients of the wavelet transform of an image to achieve robustness against compression attacks [1]. Secondly, we obtain through the use of a Harris corner detector points in the image which are likely to remain present under most geometric attacks [3][4][5]. We obtain a set of triangles by computing the Delaunay tessellation from these points. A triangular watermark is then embedded into each of these triangles by warping the watermark via affine transformations into the shape of each of these triangles. Watermark detection is accomplished by retrieving from the image the same triangles obtained in the embedding process, then warping each of these triangles back to the original watermark shape. If this is not possible, for example if the image has been compressed to the extent that the watermark cannot be detected, we then attempt to extract the watermark that was embedded into the wavelet transform.
The remainder of the paper will proceed as follows. In section 2 we present the details of the compression-resistant technique and the results achieved in applying it. Likewise, in section 3 the technique robust to geometric distortion is explained along with its results in isolation. We then combine the two approaches, which is presented in section 4 with our the experimental results from our dual approach. Finally, we give our concluding remarks in section 5.

### Table 1: Experimental Results for Compression Approach

<table>
<thead>
<tr>
<th>Attack</th>
<th>Lena</th>
<th>Baboo</th>
<th>Beach</th>
<th>Tank</th>
</tr>
</thead>
<tbody>
<tr>
<td>JPEG 90</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
</tr>
<tr>
<td>JPEG 80</td>
<td>0.99, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
</tr>
<tr>
<td>JPEG 70</td>
<td>0.99, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
</tr>
<tr>
<td>JPEG 60</td>
<td>0.99, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
</tr>
<tr>
<td>JPEG 50</td>
<td>0.99, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
</tr>
<tr>
<td>JPEG 40</td>
<td>0.99, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
</tr>
<tr>
<td>JPEG 30</td>
<td>0.99, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
<td>1.00, 0</td>
</tr>
<tr>
<td>JPEG 20</td>
<td>0.99, 4</td>
<td>1.00, 5</td>
<td>1.00, 5</td>
<td>1.00, 3</td>
</tr>
<tr>
<td>JPEG 10</td>
<td>0.99, 31</td>
<td>1.00, 27</td>
<td>1.00, 39</td>
<td>1.00, 20</td>
</tr>
<tr>
<td>Med. 2x2</td>
<td>1.00, 19</td>
<td>1.00, 17</td>
<td>1.00, 1</td>
<td>1.00, 5</td>
</tr>
<tr>
<td>Med. 3x3</td>
<td>1.00, 0</td>
<td>1.00, 19</td>
<td>1.00, 1</td>
<td>1.00, 2</td>
</tr>
<tr>
<td>Med. 4x4</td>
<td>1.00, 21</td>
<td>1.00, 31</td>
<td>1.00, 0</td>
<td>1.00, 9</td>
</tr>
<tr>
<td>Med. 5x5</td>
<td>1.00, 8</td>
<td>1.00, 28</td>
<td>1.00, 1</td>
<td>1.00, 7</td>
</tr>
<tr>
<td>Med. 6x6</td>
<td>1.00, 20</td>
<td>1.00, 33</td>
<td>1.00, 1</td>
<td>1.00, 15</td>
</tr>
<tr>
<td>Med. 7x7</td>
<td>1.00, 17</td>
<td>0.99, 32</td>
<td>1.00, 1</td>
<td>1.00, 16</td>
</tr>
<tr>
<td>Med. 2x2</td>
<td>1.00, 17</td>
<td>1.00, 18</td>
<td>1.00, 0</td>
<td>1.00, 8</td>
</tr>
<tr>
<td>Mean. 3x3</td>
<td>1.00, 13</td>
<td>1.00, 11</td>
<td>1.00, 2</td>
<td>1.00, 13</td>
</tr>
<tr>
<td>Mean. 4x4</td>
<td>1.00, 26</td>
<td>1.00, 22</td>
<td>1.00, 10</td>
<td>1.00, 20</td>
</tr>
<tr>
<td>Mean. 5x5</td>
<td>1.00, 9</td>
<td>0.99, 10</td>
<td>0.99, 12</td>
<td>0.98, 13</td>
</tr>
<tr>
<td>Mean. 6x6</td>
<td>1.00, 27</td>
<td>0.99, 20</td>
<td>0.99, 3</td>
<td>0.99, 18</td>
</tr>
<tr>
<td>Mean. 7x7</td>
<td>0.99, 23</td>
<td>0.98, 12</td>
<td>0.97, 8</td>
<td>0.96, 12</td>
</tr>
</tbody>
</table>

The remainder of the paper will proceed as follows. In section 2 we present the details of the compression-resistant technique and the results achieved in applying it. Likewise, in section 3 the technique robust to geometric distortion is explained along with its results in isolation. We then combine the two approaches, which is presented in section 4 with our the experimental results from our dual approach. Finally, we give our concluding remarks in section 5.

### 2. COMPRESSION-RESISTANT WATERMARK

In order to achieve robustness against JPEG compression, we implement the following embedding scheme, originally presented in [1]. A key-generated watermark $m$ consisting of $n$ bits is dually encoded using first BCH then Direct-Sequence Spread Spectrum (DSSS) encoding schemes, used for their error-correction capabilities, resulting in the final watermark $m' = \{ m'_i \mid i = 1..\ell, m'_i \in \{-1, 1\} \}$. In our experiments, we use $n = 60$ and $BCH(60, 72)$, resulting in $\ell = 72$.

The image to be marked is transformed via a 4-level DWT using the Daubechies 9/7 filter, obtaining detail-coefficient sub-blocks $\{H_i, V_i, D_i \mid i = 1..4\}$ and approximation coefficient sub-block $A$, The $\ell$ bits of $m'$ are reshaped to a rectangular pattern and placed in the upper-left corner of a new matrix $B$. The successive-packing interleaving (SPI) scheme described in [2] is then applied to $B$ (fig. 1). Once the data has been interleaved, it can be embedded into $A$ according to [1], where $\alpha$ is the watermark embedding strength (chosen to be 90 for our experiments). The resulting matrix, $A'$, replaces the original matrix $A$ in the transformed image, and the inverse DWT is applied to the image to conclude the embedding process.

Detection and extraction of the watermark is achieved by first applying the DWT and retrieving the same approximation-coefficient matrix $A$ used in the embedding phase. $A$ is then de-interleaved using the inverse SPI operation to restore the watermarked data to the upper-left corner of the matrix. We use the following function to reverse the calculations of the embedding function.

$$m_i(i) = \begin{cases} 1, & \text{if } A(i) \text{ mod } \alpha > \frac{\alpha}{2}, \\ -1, & \text{if } A(i) \text{ mod } \alpha \leq \frac{\alpha}{2} \end{cases},$$

To determine if the watermark was successfully detected, we compare each of the bits in $m_i$ to the bits in $m'$. If there are fewer than 5 bits that differ between the
two sequences—5 being the maximum number of recoverable errors allowed by our encoding scheme—we designate the image as having been watermarked and that the watermark has been successfully detected. Table 1 illustrates the success of this approach.

3. GEOMETRIC-RESISTENT WATERMARK

Our second approach offers a defense against distortions introduced to an image via geometric attacks. The embedding process, originally set forth in [3], follows these procedures.

As shown in [3], a Harris corner detector outperforms other corner detection schemes such as Archard-Roquet and SUSAN. Refinements made by [5] reduce the number of feature extraction points by considering the local relative strength of the points. Similarly in [4], the local strength of feature points is considered when evaluating which points to retain. It is this approach that we implement in our algorithm. A Delaunay tessellation is then applied to these points to obtain a set of triangles whose vertexes are among the set of feature points.

A secret key is used to generate a watermark $T_w$ in the shape of an isosceles-right triangle where the two off-hypotenuse edges are of length $\ell$ (in our work, this is chosen to be 64). The watermark is then warped to fit each triangle produced by the Delaunay tessellation by applying a set of affine transformations, defined by the matrix

$$T = \begin{pmatrix} a & b & e \\ c & d & f \\ 0 & 0 & 1 \end{pmatrix}$$

(3)

where $a...e$ are defined in terms of translation, rotation about the x axis, rotation about the z axis, shearing, and scale, and are obtained by extracting the following parameters for a given feature-point triangle $T_d$ (where $T_w$ is located at the origin).

First, the points of each triangle are labeled $a$, $b$, $c$ according to the relative size of their associated angles (largest to smallest). In the case of the watermark (which has 2 identical angles), the top point is considered to be $b$ and the right point $c$. Additionally, the unit vectors between each of the points is calculated.

$$\begin{align*}
\bar{x} &= (b_i - a_i) \\
\bar{y} &= (c_i - b_i) \\
\bar{z} &= (a_i - c_i)
\end{align*}$$

(4)

Our translation factor $\mu$ is then determined to be the distance needed to move $a_d$ to the origin, with the signs reversed. Rotation about the z axis ($\theta_z$) is determined by calculating the angle between the vectors $\bar{z}_w$ and $\bar{z}_d$. By applying the reverse transformations calculated thus far to $T_d$ we obtain a new triangle $T_d'$ where $a_{T_d'}$ is located at the origin and $c_{T_d'}$ is located along the x axis. Because $T_d'$ can be one of two possible triangles, we use the y component of $b_{T_d'}$ to determine the angle of rotation about the x axis $\theta_x$ as either 0 or $\pi$. Shearing angle $\theta_s$ is determined to be the magnitude of the angle associated with point $a_{T_d'}$. Finally, we calculate the scaling factors $s_x, s_y$, considered along each of the axes independently, as the ratio of the widths and heights of the two triangles, respectively.

Now that all the parameters needed to transform $T_w$ have been calculated, we create a 3x3 matrix for each transformation and concatenate the matrices as described in [6] to obtain $T$. In order to apply the transformations, each pixel in $T_w$ is transformed to form a new triangle $T_{w'}$, which is then interpolated using a cubic-spline

- Figure 3. Watermark data during transformation process. a) original, b) after scaling, c) after shearing, d) after x rotation, e) after z rotation, and f) after translation.
interpolation scheme to obtain the final triangle $T_f$.

Since noise introduced to the extremes of the grayscale are less noticeable visually than in the central portion [7], we associate with each grayscale value a luminance correction value defined as

$$\omega = \{\alpha | i = 0...255, \alpha_i = (128-i)^2/128^2\} \quad (5)$$

The transformed, interpolated triangle is then embedded into the image with an additive scheme

$$im(i, j) = im(i, j) + \alpha \omega(\alpha)(i, j)T_f(i, j) \quad (6)$$

where $\alpha$ is the embedding strength and $im$ is the image.

The procedure for the detection of a watermark in an image is similar to the embedding process. Using our Harris corner detector we extract the feature-points, which are then employed to calculate the Delaunay tessellation. For each resultant triangle $T_d$, a set of affine transformations is calculated to transform the triangle back to the original watermark shape, which is then interpolated using the same scheme utilized by the embedding scheme to obtain triangle $T_f$. To extract the watermark data, a Wiener filtering is performed on the triangle as described in [3], resulting in the triangle $T_k$. The correlation value $corr(T_w, T_k)$ is calculated and stored for each triangle.

A positive detection of the watermark is determined in two parts. First, each local correlation value is compared with a threshold $\tau_l = 0.0174$, which was determined empirically by generating 1000 watermarks with different keys and then computing the mean correlation value among 100 randomly generated triangles per watermark.

If 70% of the local correlation values are greater than this threshold, the watermark is determined to be locally detected. Second, a new metric

$$S = \sum_1^N corr(T_w, T_k) \quad (7)$$

is computed to determine whether or not the mark is detected globally. A second threshold value $\tau_g$ based on the probability of a false positive $P_{fp}$ (taken to be 12.85 as described in [3]) is calculated as

$$\tau_g = \frac{P_{fp}}{\sqrt{N}}, \quad (8)$$

where $N$ is the number of Delaunay triangles considered in the image. If $S \geq \tau_g$, then the watermark is considered to have been detected globally. If both of these conditions are met, then we have successfully detected the presence of the watermark in the image.

4. DUAL APPROACH

The embedding process in our dual approach is accomplished by applying the DWT watermarking scheme set forth in section 2 and then subsequently embedding watermark data into triangular regions of the spatial domain as described by the feature-point approach in section 3. For watermark detection, we attempt to extract the feature-point watermark. If this approach does not yield a successful result, we then proceed to extract the mark embedded into the DWT. If this approach succeeds, we conclude that the watermark is present in the image. Otherwise we consider any possible watermark in the image to be undetectable.

With this approach, we expect the results to conform to those presented in [3], with a marked improvement in regards to robustness against JPEG compression.

5. CONCLUSIONS

We have presented in the preceding remarks two separate blind watermarking approaches. The first is a technique that utilizes the DWT domain to achieve a robustness to distortion introduced by compression algorithms in particular. The second is a feature-based procedure, resistant to geometric attacks such as rotation and shearing.
With our dual embedding scheme we expect to achieve results that exceed the previously mentioned techniques by improving the amount of JPEG compression the watermarks can withstand. Because of the sparse results we have accumulated, future work will be to expand the robustness of the algorithm itself. In addition, it may be noted that successful or not, the correlation values given in Table 1 may signify a possible improvement to the detection scheme that produced them.

6. REFERENCES


