<table>
<thead>
<tr>
<th>Title</th>
<th>A New Digital Watermarking Scheme Applying Locally the Wavelet Transform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author(s)</td>
<td>Kuribayashi, Minoru / Tanaka, Hatsukazu</td>
</tr>
<tr>
<td>Citation</td>
<td>IEICE transactions on fundamentals of electronics, communications and computer sciences, E84-A(10): 2500-2507</td>
</tr>
<tr>
<td>Issue date</td>
<td>2001-10-01</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Journal Article / 学術雑誌論文</td>
</tr>
<tr>
<td>Resource Version</td>
<td>publisher</td>
</tr>
<tr>
<td>Rights</td>
<td>Copyright (c) 2001 IEICE</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://www.lib.kobe-u.ac.jp/handle_kernel/90001348">http://www.lib.kobe-u.ac.jp/handle_kernel/90001348</a></td>
</tr>
</tbody>
</table>
A New Digital Watermarking Scheme Applying Locally the Wavelet Transform

Minoru KURIBAYASHI*), Student Member and Hatsukazu TANAKA††, Regular Member

SUMMARY Generally, the wavelet transform is applied for watermarking, where a whole image is transformed to the frequency domain. However, many schemes which adopt the wavelet transform are weak for StirMark attack designed to affect locally such as rotation, extension, reduction, etc. In this paper, we propose a new scheme which applies the wavelet transform locally and has a tolerance for StirMark attack. How to maintain the quality of the watermarked image is also shown.

key words: wavelet transform, watermark, StirMark attack, searching protocol

1. Introduction

The recent advance of information technology in the World Wide Web, digital networks and computers make it possible to communicate a lot of digital contents as their values increase, and it becomes possible for everyone to get them from the net very easily. The cost of transmission is much cheaper than the conventional one. However, the authors wonder if the copyright of their contents is protected in the network as digital contents can be easily copied without any degradation. Therefore the copyright protection of the digital contents is the important problem in this field. Digital watermarking is one of the effective schemes to protect the right of authors and to prevent the production of illegal copy by embedding some information in the digital contents without being perceived. And the embedded signal can be extracted by only the authorized party. Therefore the author can claim his copyright if the embedded signal indicates his proper information. Here, if an individual user knows how to embed and extract a watermark, unauthorized parties may forge the embedded information. To prevent such a situation, we establish a trusted center which governs a watermarking system, and each author must register his contents to the center in order to protect the copyright. Usually many schemes are designed without requiring original image, since the access to it is difficult in many applications [1]. However, in our proposed scheme the existence of trusted center is assumed and hence a data base is available in extraction process. Generally, the watermarking schemes referring the original data are robust against attacks. Therefore in our scheme a watermark is extracted using the original data.

There are many schemes for watermarking techniques [2], [3]. Some schemes embed a watermark into the spatial domain, and others embed it into the transformed domain such as DCT, wavelet transform, DFT, etc. Generally, a signal embedded into the frequency domain has a tolerance for attacks because the embedded signal is spread over the whole image. Here, if the signal is embedded into only a few frequency components, the distortion may be perceptible as it forms a certain pattern in the image. So we apply the wavelet transform and DCT to make the watermark spread over the frequency domain. Furthermore, PN sequence is multiplied to make more spread effects. The procedure is given as follows. First, the wavelet transform is performed to the sub-image which consists of some local blocks in order to endow the robustness against geometrical attacks, and then the coefficients multiplied by PN sequence are transformed to the DCT domain. Since the attacks distort the shape of the image, we may not get the correct frequency coefficients from the attacked image, but the synchronization of the orthogonal axis can be recovered easily by matching each local block with a template.

2. Basic Technique

StirMark attack* [4], [5] is widely applied for a benchmark test of a watermarking scheme. Because it combines many attacks such as JPEG compression, additive noise, rotation, extension, reduction, etc. Considering the characteristic of StirMark attack, we propose a new watermarking scheme based on the following basic concepts.

2.1 Tolerance for the Geometric Distortions

There are two types of attack for the watermarking of an image. One is the signal processing such as filtering, noninvertible compression, additive noise and so on, and the other is the geometric transformations which make an image rotated, stretched, clipped, sheared, etc.

*StirMark3.1: http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/
Because the tolerance for the distortions caused by geometric transformations is much lower than that of signal processing in the almost scheme. Since geometric distortions change the coordinate axis of orthogonal transform, the frequency components cannot be recovered correctly from the distorted image.

Some schemes are concerned with geometric distortions due to rotation, scale and translation (RST). To be robust against these distortions, they apply the RST invariant modulus like Fourier-Mellin transform [6], [7]. As the results, they can immobilize the distortions caused by RST. However, the distortions caused by StirMark are different from that of RST because it gives a minor unnoticeable geometric distortions such as random slight stretch, shift, rotation, etc. It is wonder if RST invariant scheme can be robust against random geometric transformations.

In our scheme, considering the geometric transformations, a watermark is embedded into small blocks, and the searching protocol [8] is applied to search for the amount of shift and rotation to recover the orthogonal axis. As the results, the watermarked blocks can be synchronized with the original one, and hence the watermark can be extracted correctly.

2.2 Wavelet Transform

Generally, the Haar base is applied for the wavelet transform (WT) of images, and the transform maps a whole image to the frequency domain. In the transformation process, an image is first divided into four sub-bands \((LL_1, LH_1, HL_1, HH_1)\), where \(LL_1\) is called multi-resolution approximation (MRA) and the others multi-resolution representation (MRR). The former represents a half scaled image, which can be transformed similarly and be divided into four sub-bands again. The latter is the element of difference in vertical, horizontal and oblique components and the coefficients are high frequency components of the image.

As an embedded signal into the wavelet coefficients is spread over the whole image by the inverse wavelet transform, the perceptual degradation becomes very small. However, attacks for the local area in the image cause serious effects to the wavelet coefficients. If you consider the effects caused by local attacks, the wavelet transform must be performed in the local area, for example divided blocks. Therefore, we make a sub-image which consists of some local blocks in an image and then embed a few watermark bits into the sub-image. Here, embedding a watermark in high frequency components may be useful as the perceptual degradation caused by the embedding is very small. Then, from the fact that much high frequency elements are vulnerable for some signal processing, a watermark is embedded into the elements of \(LH_2\) and \(HL_2\) in our proposed scheme.

2.3 Spread Spectrum

In spread spectrum communication, the signal-to-noise ratio in every frequency band will be small although the power of the signal to be embedded can be large. And even if parts of the signal could be removed in several frequency bands, enough information should be present in the other bands to recover the signal. Spread spectrum techniques also protect the watermark privacy using a secret key to control a pseudo-random noise generator. Therefore many schemes are proposed applying the technique [1].

In our scheme a watermark is embedded in the DCT coefficients which are obtained from the wavelet coefficients \(LH_2, HL_2\). If a watermark is embedded directly in the DCT coefficients, a certain pattern may appear in the image. Therefore, spread spectrum technique is applied to spread the watermark over the whole frequency elements of the DCT coefficients. As the results, the embedded signal is spread over the wavelet coefficients \(LH_2, HL_2\) and it makes a noisy pattern which becomes less perceptible.

3. Proposed Scheme 1

A watermark is embedded into the sub-image which is picked out from an image because of the following reasons. The sub-image can be derived from a distorted image by calculating the amount of rotation and translation although they may change the coordinate of the image. It means that an operation can be performed before extracting the watermark from the image. The operation is called a searching protocol which is illustrated in this section. The important point is how to pick out the sub-image from an image, where the sub-image consists of four blocks selected randomly.

3.1 Embedding

In our scheme a watermark is embedded into a sub-image which is consisted of four blocks selected locally from an image. Here, if the sub-image is chosen from only one block, the security may decrease. On the contrary, a watermark embedded in many small pieces of blocks is vulnerable for geometric distortions. Considering this property, a watermark is embedded into the sub-image which consists of four small blocks selected randomly from the image. Then the block size is fixed \(16 \times 16\) because of the property of searching protocol [8].

Figure 1 illustrates how to embed a watermark and the procedure to embed 2 bits is given as follows.

Step 1. Select four blocks \(A_i\) \((1 \leq i \leq 4)\) of which size is \(16 \times 16\) from an image using a secret key, and construct \(32 \times 32\) sub-image of those four blocks.
Step 2. Perform the hierarchical wavelet transform to the sub-image, and obtain the two elements \( LH_2 \) and \( HL_2 \).

Step 3. Multiply PN sequence to the elements \( LH_2 \) and \( HL_2 \), and then perform DCT.

Step 4. Embed a watermark information bit in each domain as follows. If watermark bit is equal to 0, then \( m \) is added to a special DCT coefficient, else \( m \) is subtracted from it, where the special DCT coefficients are determined by a secret key and \( m \) is the embedding intensity.

Step 5. Multiply the same PN sequence to each element after performing IDCT and transform it to the spatial domain by the inverse wavelet transform (IWT). Finally divide the watermarked sub-image into four blocks and bring them back to the original positions.

When IDCT is performed and PN sequence is multiplied in step 5, the embedded signal is spread over the frequency components of the sub-image. And the spread signal is spread further over the sub-image by the following inverse wavelet transform. By repeating such a procedure \( n \) times, \( 2^n \)-bit watermark information can be embedded into the image, where in step 1 each sub-block has never been selected overlapped.

3.2 Searching Protocol

A watermarked image may be distorted by the attacks such as rotation and shift. Without the synchronization of orthogonal axis, the watermark can not be extracted correctly. As the watermark is embedded into four blocks \( A_i \), it is possible to recover the orthogonal axis by performing the following operation to each block. The amount of rotation and shift is calculated by the comparison with the original block in the searching protocol. The search is performed by checking the MSE (Mean Square Error) between the original block and the distorted one, where the distorted block is picked out from a searching domain block \( K \) shown in Fig. 2. \( d \) is called a searching distance, and the distorted shapes \( T_j (0 \leq j \leq 12) \) are indicated in the Fig. 3. Here, the block size of \( A_i \) is \( 16 \times 16 \) because of the computational complexity and the efficiency of the search. If the size is bigger than it, it takes more computation time to search the distorted pattern and its position though they can be recovered with high quality. On the other hand, they can be recovered with less computation and lower quality in the case of the smaller size. And the number of the distorted shapes \( T_j \) is determined by the following reason. If we would evaluate the MSE using more shapes, it might need more computation time and more memory to estimate the MSE of all possible rotated blocks. If the number of distorted shapes would be a few, the accuracy of the search might decrease. In order to compromise these property, 13 candidates of the shapes are designed to search the block, where \( T_0 \) is the original shape and \( T_1 \) to \( T_{12} \) are the shapes which can be obtained by the rotation of \( T_0 \).

Let \( I' \) be the image which may be copied illegally from the watermarked image \( I \). Then each rotated and shifted block \( A_i' \) can be searched using the original image as follows.

Step 1. Perform the following three operations.

1. Pick out the block of shape \( T_0 \) from the upper left side in the searching domain block \( K^* \), calculate the MSE of the block, and then store the value of MSE and the block.

2. Calculate the MSE of the block picked out from the position shifted one pixel. If the MSE is less than the stored one, then replace the former MSE and its block with new ones.

3. Repeat the operation (2) until checking all the blocks of possible position in \( K^* \).

Step 2. Perform the step 1 for the shapes \( T_1 \) through \( T_{12} \) repeatedly, where the block picked out from \( K^* \) is reformed to the square shape like \( T_0 \), and then its MSE is calculated.

Step 3. Output the block \( A_i' \) of \( 16 \times 16 \) which is finally left in the storage.
3.3 Extracting

The watermark can be extracted from the four blocks \( A_i \) which are found out from the distorted image and reformed to the square shape. Then the original image and the secret key is necessary to extract the watermark. As the result applied the searching protocol, the synchronization loss can be recovered correctly. Therefore, the watermark will be extracted by comparing the recovered sub-image with the original one.

The procedure to extract 2-bit informations from four blocks \( A_i \) can be shown as follows.

**Step 1.** Construct a sub-image from four blocks \( A_i \).

**Step 2.** Perform the hierarchical wavelet transform to the sub-image and perform OCT to the elements \( L_{H2} \) and \( H_{L2} \) which is multiplied the PN sequence.

**Step 3.** Perform the following operation to each element. Calculate the difference between the original OCT coefficient and distorted one, where the OCT coefficient is specified by the secret key. If the difference is positive, then the watermark bit is regarded as 0, else 1.

The embedded 2n-bit informations can be extracted by repeating the above procedure \( n \) times.

4. Proposed Scheme II

Generally, a watermark may be imperceptible if an image includes much high frequency elements, and the maximum amount of embedding information depends on the character of the image. So it is efficient to determine some parameters by it [9]. In this section, the selection of blocks for embedding a watermark is based on the amount of high frequency components. The procedure is given as follows. First, a block is selected by the method similar to the case of proposed scheme I. Then the variances are calculated for several directions illustrated in Fig. 4. Finally, if its variances are larger than a certain criterion, the block is applied for watermarking. As the result, only the blocks can be selected which include much high frequency elements, and a large signal can be embedded as a watermark without serious perceptual degradation in the original image.

In the extraction process the same blocks which is applied for embedding should be selected. If the selected positions are stored, then it is easy task but it requires more memory. As the original image can be referred in our extraction process, the blocks can be selected again by the same operation as used in the embedding process. Therefore the position of the blocks can be determined by the secret key and the original image.

5. Computer Simulated Results

The following results are obtained by embedding a length 72-bit vector into images in the proposed scheme I. In our simulation some standard images such as “lena,” “girl,” “baboon” and “peppers” are applied, where each image has RGB 256 level color scale with size of 256 x 256. An image is transformed to the luminance element where the watermark is embedded and extracted.

5.1 Degradation

The degradation of the watermarked image depends on the amount of the embedding information and the embedding intensity. At first, we research how much the watermarked image is degraded by changing the intensity \( m \), where the size of watermark is random 72 bits. Figure 5 shows the PSNR versus embedding intensity \( m \) for the image “lena.” The results of other images are the same as that of “lena.” As a watermark is embedded by adding \( \pm m \) to a DCT coefficient, the power of embedded signal like a noise depends on the embedding intensity \( m \). Therefore, the value of PSNR is independent of the image in the proposed scheme I. The embedding intensity \( m \) should be designed to retain a strong tolerance for attacks. However, the quality of the image decreases if the embedding intensity is increased. Therefore, considering the trade-off between the quality and robustness, the embedding intensity is set to the range \( 8 \leq m \leq 12 \) in the proposed scheme I.

In the proposed scheme II, the possible amount of embedding information depends on the image and embedding blocks are selected by the amount of the
variances. Therefore, we calculate the variances of the blocks which are selected by applying the proposed scheme I, and determine some of them to embed a watermark. It is necessary to calculate the variances of 144 blocks as the whole amount of embedded information is 72 bits and in each sub-image of four blocks 2 bits are embedded. Figure 6 shows the possible amount of embedding information for each variance and it indicates the following features. The possible amount of information is different for each variance, and the image "baboon" can embed more information than the other images because it contains a lot of high frequency components compared with others.

The degradation of the image is shown in the following images. Figure 7 is original image and Fig. 8 is obtained by embedding a watermark in the original image for \( m = 12 \). There is no obvious visually difference between the original image and the watermarked one. It seems that \( m \) can be raised from the point of the PSNR, but the distortions become obvious in the region where the high frequency elements are very small. Figure 9 is obtained by embedding a watermark for \( m = 20 \). The distortions in a flat region become obvious compared with a noisy region which has a lot of high frequency elements. Because the distortions caused by embedding are blurred in the latter region. Since such regions are selected in the proposed scheme II, the embedding intensity can be set to \( m = 20 \). Figure 10 is obtained by embedding a watermark when \( m = 20 \) and the variance is 50. The obvious distortions are disappeared in this image and the perceptual quality is maintained.

5.2 Searching Distance

If the StirMark attack is performed, the blocks which embed a watermark will be slightly rotated and shifted. In the searching protocol the distorted block is found out from the attacked image. As the performance depends on the searching distance \( d \), we examine the rate of correct extraction for each distance \( (2 \leq d \leq 16) \), where the "correct" means that a watermark can be extracted with no error. The results are shown in Fig. 11. When the searching distance \( d \) is over 12, the improvement of the rate is very few. Therefore StirMark attack
KURI8AYASHI and TANAKA: A NEW DIGITAL WATERMARKING SCHEME APPLYING LOCALLY THE WAVELET TRANSFORM

Fig. 10 Watermarked image \((m = 20, \text{variance } = 50)\).

Fig. 11 The rate of correct extraction versus the searching distance \(d\).

will rotate and shift the block \(L\) in the domain which size is \(28 \times 28\). In the following simulation the searching distance is set to \(d = 12\).

5.3 StirMark Attack

In this section, StirMark attack is performed to examine the robustness under the following condition. A random number of 72 bits is embedded using the proposed scheme I, where the simulation time is \(10^5\). The rates of correct extraction are shown in Table 1, where the version of StirMark is 3.1 which parameters are defaults. From this table, the tolerance for StirMark attack is not enough even when the embedding intensity is \(m = 12\). However, the number of errors is a very few and the distribution is limited to small bits. For example, the error distribution rates for image “lena” are shown in Fig. 12. It indicates that the number of errors is almost less than 3. Therefore, some error correcting codes are available for the 72-bit watermark information. The results for other images are shown numerically in Table 2, where the embedding intensity is fixed for \(m = 12\). They show the similar result of “lena.” So the number of errors is independent of the character of images. Hence the proposed scheme I can immunize the StirMark attack if error correction is applied. And it is clear that the number of errors must be decreased if the embedding intensity is increased. Table 1 also shows the result for \(m = 20\). From these results, it is evident that the watermark will remain in the attacked image with extremely high probability in the proposed scheme II as well as the proposed scheme I.

As a watermark is embedded in a DCT coefficient determined by a secret key, the tolerance of each coefficient should be examined. Table 3 shows the rates of correct extraction for each coefficient, where \((*,*)\) is the coordinate of DCT coefficient. From Table 3, the robustness is almost independent of the embedding coefficient. Since PN sequence is multiplied, a watermark embedded in one DCT coefficient is spread over the whole frequency elements. The degradation for each coefficient is shown in Table 4. As well as the robustness, the PSNR is almost independent of the embedding coefficients.

<table>
<thead>
<tr>
<th>(m)</th>
<th>lena [%]</th>
<th>girl [%]</th>
<th>baboon [%]</th>
<th>peppers [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>37.1</td>
<td>68.8</td>
<td>14.9</td>
<td>37.4</td>
</tr>
<tr>
<td>10</td>
<td>70.5</td>
<td>87.4</td>
<td>47.8</td>
<td>69.1</td>
</tr>
<tr>
<td>12</td>
<td>88.4</td>
<td>95.3</td>
<td>75.7</td>
<td>86.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>99.8</td>
<td>99.8</td>
<td>99.3</td>
<td>99.5</td>
</tr>
</tbody>
</table>

Table 2 Error distribution rate \((m = 12)\).

<table>
<thead>
<tr>
<th>image</th>
<th>0 bit [%]</th>
<th>1 bit [%]</th>
<th>2 bits [%]</th>
<th>3 bits [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>girl</td>
<td>95.3</td>
<td>4.6</td>
<td>0.2</td>
<td>0.0</td>
</tr>
<tr>
<td>baboon</td>
<td>75.7</td>
<td>20.8</td>
<td>3.1</td>
<td>0.4</td>
</tr>
<tr>
<td>peppers</td>
<td>86.6</td>
<td>12.5</td>
<td>0.9</td>
<td>0.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(m)</th>
<th>lena</th>
<th>girl</th>
<th>baboon</th>
<th>peppers</th>
</tr>
</thead>
<tbody>
<tr>
<td>((0,0))</td>
<td>90.9</td>
<td>90.7</td>
<td>91.9</td>
<td></td>
</tr>
<tr>
<td>((3,3))</td>
<td>96.4</td>
<td>96.4</td>
<td>96.7</td>
<td></td>
</tr>
<tr>
<td>((7,7))</td>
<td>77.6</td>
<td>76.2</td>
<td>75.0</td>
<td></td>
</tr>
<tr>
<td>peppers</td>
<td>88.0</td>
<td>89.9</td>
<td>88.4</td>
<td></td>
</tr>
</tbody>
</table>

Table 3 Robustness for each embedding coefficient \((m = 12)\).
Table 4  PSNR for each embedding coefficient (m = 12).

<table>
<thead>
<tr>
<th>quality</th>
<th>(0,0)</th>
<th>(3,3)</th>
<th>(7,7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR [dB]</td>
<td>44.1</td>
<td>44.0</td>
<td>44.0</td>
</tr>
</tbody>
</table>

Table 5  Tolerance for JPEG compression “lena.”

<table>
<thead>
<tr>
<th>m</th>
<th>25%</th>
<th>30%</th>
<th>35%</th>
<th>40%</th>
<th>45%</th>
<th>50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>11.1</td>
<td>35.4</td>
<td>74.4</td>
<td>88.5</td>
<td>94.0</td>
<td>99.3</td>
</tr>
<tr>
<td>10</td>
<td>43.4</td>
<td>79.9</td>
<td>95.0</td>
<td>99.0</td>
<td>99.4</td>
<td>100.0</td>
</tr>
<tr>
<td>12</td>
<td>80.6</td>
<td>98.6</td>
<td>99.7</td>
<td>99.9</td>
<td>100.0</td>
<td>100.0</td>
</tr>
</tbody>
</table>

Table 6  Error distribution rate after high compression “lena,” (m = 8).

<table>
<thead>
<tr>
<th>quality [%]</th>
<th>0 bit</th>
<th>1 bit</th>
<th>2 bits</th>
<th>3 bits</th>
<th>other</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>11.1</td>
<td>22.9</td>
<td>24.1</td>
<td>17.5</td>
<td>24.4</td>
</tr>
<tr>
<td>30</td>
<td>38.4</td>
<td>33.8</td>
<td>18.7</td>
<td>6.4</td>
<td>2.7</td>
</tr>
<tr>
<td>35</td>
<td>74.4</td>
<td>21.2</td>
<td>3.8</td>
<td>0.4</td>
<td>0.2</td>
</tr>
<tr>
<td>40</td>
<td>88.8</td>
<td>10.1</td>
<td>1.0</td>
<td>0.1</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Table 7  Error distribution rate after high compression (m = 8, quality 30%).

<table>
<thead>
<tr>
<th>image</th>
<th>0 bit</th>
<th>1 bit</th>
<th>2 bits</th>
<th>3 bits</th>
<th>other</th>
</tr>
</thead>
<tbody>
<tr>
<td>girl</td>
<td>34.0</td>
<td>31.5</td>
<td>21.3</td>
<td>8.1</td>
<td>5.1</td>
</tr>
<tr>
<td>baboon</td>
<td>43.7</td>
<td>35.6</td>
<td>14.4</td>
<td>3.1</td>
<td>1.2</td>
</tr>
<tr>
<td>peppers</td>
<td>39.5</td>
<td>33.1</td>
<td>17.2</td>
<td>7.0</td>
<td>3.2</td>
</tr>
</tbody>
</table>

5.4 JPEG Compression

This section examines the robustness against JPEG compression for the same condition as the previous experiment except the simulation time $10^3$. Table 5 shows the rates of correct extraction for some different embedding intensity m. When the embedding intensity is $m = 12$, a strong tolerance can be obtained for higher compression. The results of other images are also shown in Fig. 13. Here we examine the number of errors for $m = 8$, the results are shown in Table 6, Table 7. From those tables the number of errors is almost less than 3 bits when quality is over 30%. Therefore the watermark can be detected correctly from the high compressed image with error correction even when $m = 8$.

Table 8 shows the result in the proposed scheme I for $m = 20$ and it indicates that the proposed scheme I retain a strong tolerance for JPEG compression. As the results, it is evident that the proposed scheme II is also robust against JPEG compression.

5.5 Consideration.

To compare our scheme with others [10], [11], our scheme shows the following features. The possible amount of embedding information is more than them. Our proposed scheme I is 72 bits which has already encoded, but 16 bits in [10], 32 bits in [11]. And the value of PSNR is higher than them. The merit of our scheme is that the original image is required to extract a watermark. However we establish a trusted center which can refer the original image in the extraction process.

A watermark should retain the features such that it must remain in the image no matter what attacks are performed and the size of it should be enough to express the copyright information. In the proposed scheme I the size is large enough and independent of the character of an image though the embedding intensity cannot be raised. Therefore, an error correcting code must be applied to improve the robustness against attacks. In the proposed scheme II the size of a watermark is different in each image, but the watermark will be robust against attacks without applying an error correcting code. From this fact, the watermarking scheme can be made efficient if the scheme is selected adaptively based on the character of the image. For example, the proposed scheme I should be applied for the image which has a few high frequency components such as “lena,” “girl” and “peppers,” but the scheme II should be applied for the image like “baboon.”

The size of a watermark should be independent of the image, the watermark should retain a strong tolerance for attacks and the embedding intensity can be determined based on the blocks for the following reasons. A noisy region which has much high frequency elements is vulnerable for attacks, but the distortions caused by embedding are blurred in it. And a watermark embedded into a flat region will be obvious compared with a noisy region. So the embedding intensity in each block should be determined from the point of perceptual feature and the watermark size, and then the robustness can be improved by error correction. As the results, the proposed scheme will become to retain the preferable feature for watermarking.
6. Conclusion

We have proposed a new watermarking scheme applying locally the wavelet transform. A watermark is embedded into four blocks selected randomly in the image, where the concept of spread spectrum is applied for the wavelet coefficients. Hence it is difficult to remove the watermark without serious degradation to the image. If you want to forge the embedded information, you need to know the position of the selected blocks and the PN sequence used for spread spectrum. Considering the distortions caused by geometric transformation, it is possible to recover the synchronization by the searching protocol. As the result, our scheme immunizes the StirMark attack as well as high compression by JPEG, which can be certified by our simulated results on the tolerance of the attacks.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments.

References


Minoru Kuribayashi received the B.E. degree in 1999, and M.E. degree in 2001, both from Kobe University, Kobe, Japan. He is currently a student of the Graduate School of Science and Technology, Kobe University. His research interests are in digital watermark, information security and cryptography.

Hatsukazu Tanaka was born in Hyogo, Japan, on September 30, 1941. He received the B.E. degree from Kobe University, Kobe, Japan in 1964, the M.E. degree in 1966, and the D.E. degree in 1969, both from Osaka University, Osaka, Japan. He was appointed as a Research Associate in the Faculty of Engineering, University of Osaka Prefecture in 1969. From 1972 through 1987 he was an Associate Professor in the Department of Electrical Engineering, Kobe University. Since 1988 he has been a Professor in the Department of Electrical and Electronics Engineering, Kobe University. From 1980 through 1981 he was a member of the Communication Group of the University of Toronto, Toronto, Ontario, Canada, as a Visiting Scientist. His main work is on the basic theory of Information Engineering such as Information Theory, Coding Theory, Cryptography and Information Security, Image Processing, etc. Dr. Tanaka is a Fellow member of IEEE and a member of IACR.