A Secret Image Sharing Method Using Integer-to-Integer Wavelet Transform

Chin-Pan Huang\textsuperscript{1}, Ching-Chung Li\textsuperscript{2}

\textsuperscript{1}Dept. of Computer & Communication Engineering, Ming Chuan Univ., Taoyuan, Taiwan, ROC

\textsuperscript{2}Dept. of Electrical & Computer Engineering, Univ. of Pittsburgh, Pittsburgh, PA, USA

hcptw@mcu.edu.tw, ccl@engr.pitt.edu

Abstract

We present a new image sharing method based on the integer-to-integer (ITI) wavelet transform and Shamir’s \((r, m)\) threshold scheme that provide highly compact shadows for real time progressive transmission. Our approach, working in the wavelet domain, processes the transform coefficients in each subband, divides each of the resulting combination coefficient into \(m\) shadows for sharing, and recovers the complete secret image using any \(r\) or more shadows\((r \leq m)\). We take advantage of the properties of the wavelet multi-resolution representation of an image, such as coefficient magnitude decay and excellent energy compaction, to design combination procedures for transform coefficients and processing sequences in wavelet decomposition subbands such that compact shadows for real time progressive transmission are obtained.

1. Introduction

Recently, secret image sharing has become a key technology used to keep confidentiality in the field of information security and protection. Shamir \cite{1} first proposed a concept of secret data sharing called a \((r, m)\) threshold scheme. Thien and Lin \cite{2,3} developed a secret image sharing method based on this \((r, m)\) threshold scheme. Their method permutes a secret image first to de-correlate pixels and then incorporates the \((r, m)\) threshold scheme to process the image pixel-wise or pattern-wise in the spatial domain sequentially, hence, it may not be suitable for real time progressive transmission.

A reversible integer-to-integer (ITI) wavelet transform maps an integer-valued image to integer-valued transform coefficients and provides the exact (lossless) reconstruction of the original image \cite{4 - 8}. This multi-resolution representation can be fast computed with only integer addition and bit-shift operations. Most of the signal energy is concentrated in the low frequency bands and the transform coefficients therein are expected to be better magnitude-ordered as we move downward in the multi-resolution pyramid in the same spatial orientation \cite{4,5,8}. The smooth (scaling) coefficients have the same range of values as that of the input image and the detail (wavelet) coefficients have smaller absolute values than the input image. Instead of using permutation to de-correlate pixels \cite{2} prior to applying the \((r, m)\) threshold scheme, we first apply ITI wavelet transform and then process transform coefficients with a combination procedure to de-correlate pixels (coefficients) and increase security, enabling the real time progressive transmission.

2. Review of Related Works

According to Shamir’s \((r, m)\) threshold scheme \cite{1}, the secret \(D\) is divided into \(m\) shadows \((D_1, D_2, \ldots, D_m)\) and any \(r\) or more shadows can be used to reconstruct it. To split \(D\) into \(m\) pieces, a prime \(p\), which is bigger than both \(D\) and \(m\), is randomly selected and a \((r-1)\)th degree polynomial is picked,

\[ q(x) = (a_0 + a_1x + \cdots + a_{r-1}x^{r-1}) \mod p. \]  \hspace{1cm} (1)

In (1), \(a_0 = D\), and \(a_1, a_2, \ldots, a_{r-1}\) are random numbers selected from numbers \(0 \sim (p-1)\). The pieces are obtained by evaluating

\[ D_i = q(i), \ldots, D_i = q(i), \ldots, D_m = q(m). \]  \hspace{1cm} (2)

Note that \(D_i\) is a shadow. Given any \(r\) pairs from these \(m\) pairs \(\{(i, D_j)\}, j = 1, 2, \ldots, m\) , the coefficients \(a_0, a_1, a_2, \ldots, a_{r-1}\) can be solved using Lagrange’s interpolation, and hence the secret data \(D\) can be revealed. In Thien and Lin’s work, they took \(a_0, a_1, a_2, \ldots, a_{r-1}\) as the gray levels of \(r\) pixels in the permuted secret image to generate \(m\) shadows.
3. The Proposed Image Sharing Method

In our proposed method described below, we take $a_0, a_1, a_2, \cdots, a_{r-1}$ as values of $r$ processed transform coefficients to generate $m$ shadows. A secret image is ITU-wavelet transformed down to a selected scale level $j$ to form its multiresolution hierarchical representation. Combination procedures for transform coefficients in individual subbands are developed first based on the strong intra-band correlation and small absolute values of the coefficients in the detail bands. Thus, we expect to have small values of differences between neighboring coefficients in the smooth subband and small coefficients in the detail subbands. These are used in the combination processes in the respective subbands to produce combination coefficients for use in the $(r, m)$ threshold scheme. The sequence of the combination process starts from the smooth subband and follows a zigzag path to the detail subbands in a hierarchical tree [8] such that the progressive transmission can be readily achieved.

3.1 Combination procedures

Since the numbers (in images) suitable for the $(r, m)$ threshold scheme are from 0 to 255 [2], we need to take care of this requirement. Thus, combination procedures are designed by concatenating neighboring transform coefficients (or coefficients differences in the smooth subband) into one byte in case they are small enough or else scaling their values into the appropriate range. Then the size of the resulting combination coefficients is reduced and its range is adjusted.

Let us consider the smooth subband with coefficients $S = \{s_{m, n}\}$ and coefficient differences $DS = \{ds_{m, n}\}$. At location $(m, n)$, coefficient difference is given by

$$ds_{m, n} = \begin{cases} s_{m, n} & \text{if } m = 0, n = 0, \\ s_{m, n} - s_{m-1, n} & \text{if } m \neq 0, n = 0, \\ s_{m, n} - s_{m, n-1} & \text{otherwise}. \end{cases}$$ (3)

Combination numbers $C_{com}$ are generated, referring to coefficients $S$ and differences $DS$, with the following steps.

1. Divide the coefficients $S$ and differences $DS$ into nonoverlapping blocks, each block contains $2 \times 2$ neighboring coefficients or differences. In each block, the four coefficient differences are designated as $ds = \{ds_{m, n}, ds_{m, (n+1)}, ds_{(m+1), n}, ds_{(m+1), (n+1)}\}$ and their corresponding coefficients are $s = \{s_{m, n}, s_{m, (n+1)}, s_{(m+1), n}, s_{(m+1), (n+1)}\}$.

2. Process each block from left to right and top to bottom.

3. The coefficient differences are combined as follows:
   (i) If the values of four differences are all not less than -2 and not greater than 1 ($-2 \leq ds \leq 1$), then these four differences are processed together by adding 2 (i.e. $0 \leq ds + 2 \leq 3$) to each difference and concatenating them into a new byte $c_{com} = ds_{m, n} | ds_{m, (n+1)} | ds_{(m+1), n} | ds_{(m+1), (n+1)}$, where $|$ denotes bitshift and bitor and will be used in the sequel. (ii) If the values of the following two differences, either $(ds_{m, n}, ds_{m, (n+1)})$ or $(ds_{(m+1), n}, ds_{(m+1), (n+1)})$, are both less than -4 and not greater than 3 ($-4 \leq ds \leq 3$), then these two differences are processed together by adding 4 (i.e. $0 \leq ds + 4 \leq 7$) to each difference and concatenating them into a new byte $c_{com} = ds_{m, n} | ds_{m, (n+1)}$ or $c_{com} = ds_{(m+1), n} | ds_{(m+1), (n+1)}$. (iii) If the values of four differences do not satisfy the condition (i) or (ii), then each coefficient $s$ in $(s_{m, n}, s_{m, (n+1)}, s_{(m+1), n}, s_{(m+1), (n+1)})$ is processed separately by checking its sign and forming a new byte $c_{com} = sgn(s) * s$ where $sgn(s) = \begin{cases} +1, & s \geq 0, \\ -1, & \text{otherwise}. \end{cases}$

4. The new byte $c_{com}$ generated in step 3 is assigned sequentially in combination numbers $C_{com}$. Note that the value of $c_{com}$ is between 0 and 255.

5. Use two bits to record the type of a new byte in step 3 as follows: 00 and 01 for concatenation of four and two differences, respectively; 10 and 11 for a positive and a negative valued byte, respectively. Every four consecutive two record bits are combined to form a byte called $t_{com}$. Note that the value of $t_{com}$ is between 0 and 255.

6. The byte $t_{com}$ generated in step 5 is recorded sequentially in type combination numbers $T_{com}$.

The inverse combination can be easily done following the reverse steps. The combination procedure for coefficients in detail subbands is done similarly.

3.2 The sharing phase

The combination numbers $T_{com}$ and $C_{com}$ are each divided into nonoverlapping sharing blocks, each containing a sequence of $r$ numbers. For each sharing block $b$, an $(r-1)$th degree polynomial is used as in [2] except here $p=257$. 

0-7695-2521-0/06/$20.00$ (c) 2006 IEEE
\[ q_b(x) = (a_0 + a_1x + \ldots + a_{r-1}x^{r-1}) \mod 257, \]  
where \( a_0, a_1, a_2, \ldots, a_{r-1} \) are \( r \) numbers of the sharing section. Evaluate

\[ D_i = q_b(i) \quad i = 0,\ldots, m, \]  
The \( m \) output numbers \( q_b(1), \ldots, q_b(i), \ldots, q_b(m) \) of this sharing block \( b \) are placed sequentially in the \( m \) shadow coefficients. In this case, the possible values of the output are \( 0 \leq q_b(i) \leq 256, i = 1,\ldots, m \). The problem is that the value of a byte in our coefficient is in the range from 0 to 255. If the output values are 255 and 256, this problem can be dealt with by associating 255 with an extra bit of 0 and 1 (for 255 and 256, respectively) stored in the following byte. In order to have an embedded progressive transmission and establish a traceable set of coefficient combination numbers, \( C_{\text{com}} \), the type combination numbers \( T_{\text{com}} \) and the byte for extra bit are stored as an overhead. Note that \( r \) type combination numbers \( t_{\text{com}} \) are associated with \( 4r \) coefficient combination numbers, \( c_{\text{com}} \). The sharing process is described below.

1. Apply the combination procedures to get the combination numbers, \( T_{\text{com}} \) and \( C_{\text{com}} \).
2. Pick \( r \) consecutive numbers from the \( T_{\text{com}} \) and \( 4r \) consecutive numbers from the \( C_{\text{com}} \) to form five sharing blocks each containing \( r \) numbers.
3. Apply the sharing equations (4) and (5) to the picked sharing block to generate \( m \) output shares for the \( m \) shadows. If the output values are less than 255, store the generated output shares in the shadows. If an output value is 255 or 256, then store the coefficient 255 in the shadow coefficient and an extra bit 0 for 255 and 1 for 256 is stored in a list that follows.
4. Go to step 2 until all combination numbers are processed.

### 3.3 The reveal phase

The combination coefficients can be revealed by any \( r \) out of \( m \) shadows via the following steps.

1. Take one pixel (coefficient) from each of the \( r \) shadows to form a shadow block sequentially from left to right and top to bottom.
2. Use these \( r \) numbers and apply Lagrange’s interpolation to solve for the values of \( a_0, a_1, a_2, \ldots, a_{r-1} \) in equation (5).
3. Steps 1 and 2 are processed for every 5 shadow blocks with one type combination block and 4 coefficient combination blocks. In case any value of \( q_b(i) \) is 255 in these 5 blocks, the following 6th shadow block is examined for the corresponding extra bit (0 or 1) to be added back.
4. Repeat steps 1 to 3 until all coefficients of the \( r \) shadows are processed. The combination coefficients are reconstructed.

### 4. Experimental Results

Four images—Lena, Jet, Monkey and Peppers, each has \( 512 \times 512 \) pixels with 8 bits per pixel, were used in our experiment. The ITI wavelet derived from Daubechies’ 5/3 biorthogonal wavelet, 6 level decomposition, and the \((r, m)\) threshold scheme with \( r=4 \) and \( m=6 \) were used. The smaller size compact shadows produced by our method (Proposed) are shown in Table 1 in comparison to those obtained by Thien and Lin’s (TL’s) method [2]. The real time progressive transmission and reconstruction performance using four of the six shadows with different percentages of shadow coefficients from these four shadows is shown in Table 2. Note that using whole shadow coefficients from these four shadows results in lossless (L.L.) reconstruction. The performance of our method on Lena image is shown in Figures 1 and 2. Figure 1(a) shows the ITI wavelet transform of Lena image. The share phase applied to the combination numbers gave the result shown in Figure 1(b) with blank lower part showing the size reduction after the coefficient combination process. In Figure 1(c), six shadows were obtained. The secret image was revealed in real time progressively using four of the six shadows with different percentages of coefficients as shown in Figure 2.

### 5. Security analysis

We have done a security analysis, similar to what was done in ref. [2], to ascertain that our method has the security property that “Any \( r-1 \) or less shadows cannot get sufficient information to reveal the secret image”. Note that our method utilizes ITI wavelet transform representation of the image and combines the wavelet coefficients prior to the sharing process. For a \( 512 \times 512 \) secret image, with the use of \( r \) pixels (combination coefficients) in a section employed in the sharing phase, there are \( 512^2 \times 512^2 \) polynomials involved where \( \alpha \) is a compact shadow factor. The range of \( \alpha \) is around 0.70 to 0.95 as obtained in our experiment. In this case, the probability of guessing the right combined coefficients in our scheme is

\[
\left( \frac{1}{257} \right)^{512 \times 512 \alpha / r}
\]

which is an extremely small value. An intruder has only this near zero probability to get the
correct processed coefficients shown in Figure 1 (b),
not to mention the difficulty to reconstruct the original image.

6. Conclusions

In the paper a new method based on reversible ITI
wavelet transform to share a secret image has been
presented. Taking advantages of transform coefficient
magnitude decay and excellent energy compaction in
wavelet representation, a combination process for
transform coefficients is developed for use in the \((r, m)\)
threshold scheme to generate shadows for image
sharing. It results in shadow size reduction and
capability for progressive transmission. The
effectiveness of our method is illustrated by
experimental results on test images.

7. References

[1] A. Shamir, “How to share a secret”, Communication of
ACM, vol. 22(11), 1979, pp. 612-613.


with user-friendly shadow images”, IEEE Trans. on CSVT,
vol. 13(12), 2003, pp. 1161-1169.

compression using biorthogonal wavelet transforms with
multiplierless operations”, IEEE Trans. on circuit and
systems-II: analog and digital signal processing, vol. 45(8),

Compression with reversible embedded wavelets”, Proc. 5th
212-221.

Yeo, “Wavelet transforms that map integers to integers”,
Applied and Computational Harmonic Analysis, vol. 5, 1998,
pp. 332-369.

compatible reversible integer-to-integer wavelet transforms”,
2624-2636.

image codec based on set partition in hierarchical trees”,

<table>
<thead>
<tr>
<th>Method</th>
<th>Shadow size comparison (Bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image</td>
<td>Proposed</td>
</tr>
<tr>
<td>Lena</td>
<td>382 × 128</td>
</tr>
<tr>
<td>Jet</td>
<td>329 × 128</td>
</tr>
<tr>
<td>Monkey</td>
<td>486 × 128</td>
</tr>
<tr>
<td>Peppers</td>
<td>330 × 128</td>
</tr>
</tbody>
</table>

Table 2. Progressive transmission and reconstruction
(PSNR, dB)

<table>
<thead>
<tr>
<th>Shadow coefficients</th>
<th>Image</th>
<th>1%</th>
<th>25%</th>
<th>50%</th>
<th>75%</th>
<th>100%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>23.29</td>
<td>31.72</td>
<td>34.42</td>
<td>37.82</td>
<td>L.L.</td>
<td></td>
</tr>
<tr>
<td>Jet</td>
<td>22.26</td>
<td>30.14</td>
<td>32.05</td>
<td>35.89</td>
<td>L.L.</td>
<td></td>
</tr>
<tr>
<td>Monkey</td>
<td>20.65</td>
<td>24.50</td>
<td>27.81</td>
<td>30.64</td>
<td>L.L.</td>
<td></td>
</tr>
<tr>
<td>Peppers</td>
<td>22.68</td>
<td>31.35</td>
<td>35.45</td>
<td>37.82</td>
<td>L.L.</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. (a) ITI wavelet transform of Lena image to 6
scale levels, (b) The result of combination process, (c)
Shadows generated by our method with \(r=4\) and \(m=6\)

Figure 2. Progressive reconstruction using any 4 out of
6 shadows and with the following percentages of
coefficients, and the resulting PSNR: (a) 1%, 23.29dB,
(b) 5%, 27.35dB, (c) 15%, 29.73dB, (d) 25%, 31.72dB,
(e) 50%, 34.42dB, (f) 100%, lossless

0-7695-2521-0/06/$20.00 (c) 2006 IEEE