International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.3, Issue.1, Jan-Feb. 2013 pp-139-148 ISSN: 2249-6645

A User-Friendly Image Sharing Scheme Using JPEG-LS Prediction and LSB Matching Function

Hung P. VO
Department of Engineering and Technology, Tra VinhUniversity, No. 126 National Road 53, Tra Vinh City, Tra Vinh Province, Viet Nam

Abstract: *User-friendly secret sharing encrypts a secret image into n meaningful image shares or shadows first proposed by Thien and Lin in 2003. Their scheme can achieve the goal of secret image sharing with the additional capabilities of simple visual-management of shadows owing to representing of shadow images as shrunken version of the original image. After that, Yang et al. proposed an improved scheme in 2007 by using different primes for various blocks in Shamir's polynomial function. In this article, we propose a novel version of (t, n)-threshold user-friendly image sharing scheme which attempts to indicate prime number for encoding a t-pixel block based on JPEG-LS prediction technique and utilize LSB matching function to hide indicators of the prime numbers. The simulator presents that the proposed scheme attains the properties of user-friendly image sharing. In particular, reconstructed image is higher than that of the previous schemes in terms of the quality of image.*

Keywords*: image sharing, secret sharing, secret image sharing, user-friendly shadow*

I. INTRODUCTION

With the advent of high-speed network infrastructure as well as low-cost computing and networking hardware, handling and processing digital information by computers and sharing them over internet has become more and more popular. Since communication via internet is considered insecurely communicated environment, demands for shielding communicated information is compulsory and necessary. So, security of digital data is an important issue in the design of communication systems. Data hiding techniques and secret sharing are used to secure confidentiality information while it is transmitted through unsecured communication channels. Unlike data hiding technique which provides a way of security for digital image data by embedding secret message in preselected meaningful images, called camouflage images [1-3].

However, the common weak point of them is that secret image is protected in a single information carrier. Once the carrier is damaged or destroyed, the secret image is lost. Differing from aforementioned methods, secret image sharing method provides protocol in which a dealer distributes secret data among a set of participants and only the number of the participants is important for recovering the secret information in the reconstructed phase. This mechanism is not only fault tolerance, but also fast transmission [6, 7]; such scheme is called a threshold secret sharing scheme.

The concept of secret sharing scheme was introduced in 1979 by Shamir [4], called the (t, n)-threshold scheme. Shamir's scheme is based on polynomial interpolation and the polynomial sharing function of degree t–1 is defined as $S(x) = s + a_1 x + \cdots + a_{t-1} x^{t-1} \bmod p$ to encode a secret number s into n shadows (denoted s_1, s_2, …, s_n), where p is the large prime number selected randomly and coefficients a_1, a_2, …, a_{t-1} are random numbers within [0, p–1].

1.1. The shares s_1, s_2… s_n are produced as $s_i = S(x_i)$, for all $1 \le i \le n$, where x_1, x_2, …, x_n are pair distinct random values.

1.2. Given any t pairs $(x_i, s_i)_{i=1}^{t}$, the secret s can be derived using Lagrange's interpolation formula as

$$S(x) = \sum_{i=1}^{t} (s_i \times \prod_{j=1, j \ne i}^{t} \frac{x - x_i}{x_i - x_j}). \quad (1)$$

In 2002, Thien and Lin [6] extended the secret sharing scheme from number system to image system, and this is called (t, n)-threshold secret image sharing scheme. In the scheme, coefficients (s, a_1… a_{t-1}) of the Shamir's polynomial are considered as pixel values and the modulus p used in the function is set to 251. A secret image is encoded into n meaningless shadow images and distributed to n involved participants. The secret image may be only reconstructed without cooperation of at least a group of t authorized participants. Any number of shadows less than t cannot be used to reveal the secret image.

However, from the view point of security, the system may still expose the secret information to attackers since it is known that the shares are the noise-like shadow images which may attract malicious user's attention. To conquer this shortcoming, recently, many proposed secret image sharing with steganography schemes have been introduced which construct n pairs of secret shadows. The produced shadows then are embedded into cover images to generate n camouflage images [8-13]. These camouflage images look like cover images and do not draw attacker's attention while transmitted to involved participants via Internet. The constraint of above-mentioned schemes is that the size of cover image is more lager than that of shadow, so it is commonly not suitable for saved-bandwidth network applications. Besides, although the shadow images are no longer noise-like but it is still difficult to manage secret image from the view of image management.

Therefore, some recent literature has been devised to overcome the above weaknesses [14, 15]. In 2002, Thien and Lin [14] proposed an image-sharing mechanism with user-friendly shadow images. Consequently, the generated shadow images are like the reduction version of the original secret image. In 2007, Yang *et al.* [15] enhanced and extended the scheme based on (t–1)-degree polynomial functions with different primes in encoding blocks. However, quality of the reconstructed images is still inferior, making them infeasible for medical, military, or artistic applications [11, 16-17].

This article presents a novel version of (t, n) user-friendly sharing scheme, capable of producing portrait-looking shadow images yet still having properties of a secretimage sharing. Particularly, each shadow image contains a slight amount of information from the original image, in which the contents of the shadow play the identification rather than restore the original secret image. Herein, the concept of Yang *et al.*'s image sharing using polynomial with different primes is used to ensure that each shadow image is a user-friendly share. However, the proposed scheme sets different prime numbers for encoding different blocks by using the JPEG-LS prediction technique [18-21]. Moreover, to prevent the original image degradation, LSB matching function [22] is utilized to hide indicators of the prime numbers.

The rest of this paper is organized as follows. Section 2 briefly reviews pertinent literature. Section 3 then describes the proposed scheme in detail. Section 4 summarizes the experimental results and discussions. Conclusions are finally drawn in Section 5.

II. RELATED WORKS

In this section, we briefly review the related techniques including the concept of JPEG-LS prediction, using LSB matching function for hiding indicators of the prime numbers and Yang *et al.*'s scheme in Subsections 2.1, 2.2 and 2.3, respectively.

2.1 JPEG-LS Prediction Technique

JPEG-LS technique [18-21] is the latest standard for lossless and near lossless image compression, abbreviated form of Joint Photographic Experts Group-Lossless Standard. The core algorithm behind JPEG-LS was developed by Hewlett Packard in 1990s, called Low Complexity Lossless Compression for Images (LOCO-I) [18]. The detailed design of the algorithm can be summarized in Equation (2) depending on the predictive template shown in Figure 1.

$$X = \begin{cases} \min(A,B) & \text{if } C \geq \max(A,B) \\ \max(A,B) & \text{if } C \leq \min(A,B) \\ A+B-C & \text{otherwise} \end{cases}, \quad (2)$$

where min(A,B) and max(A,B) stand for the minimum and maximum values among pixels A and B, respectively.

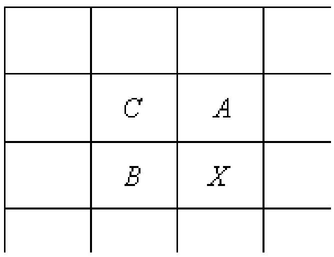

Figure 1. JPEG-LS predictive template

According to the Equation (2), predicted value X in Figure 1 will be A (B) when there is a vertical (horizontal) edge detection of the current pixel location X shown in Figure 2; or X will be A + B – C when the neighboring pixels have comparable values.

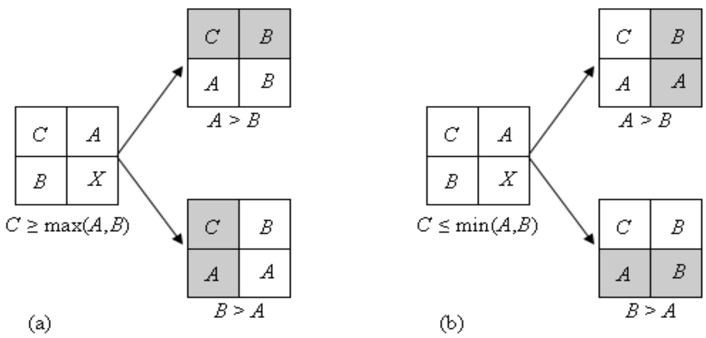

Figure 2. Edge detection flowcharts (a) $C \geq$ max (A,B); (b) $C \leq$ min(A,B)

2.2 LSB Matching Function for data hiding

Unlike conventional LSB substitution, where secret message is embedded into cover image by replacing the LSBs of the cover image with message bits, LSB matching [22] does not simply replace the LSBs of the cover image with secret message bits. In 2009, Chan proposed LSB matching function for data hiding [23]. By adopting a one-dimension approach instead of two-dimension approach of the image, the LSBs of a sequence of image pixels are computed by XOR function (XF) defined in Equation (3) to be compatible with the message bits.

$$XF(y_i) = LSB\left(\left\lfloor \frac{y_{i-1}}{2} \right\rfloor\right) \oplus LSB(y_i), \quad (3)$$

Where y_i, y_{i-1} are pixel values at the location i and $i-1$, respectively. The operator $\oplus$ is an exclusive OR operator and LSB (.) is the least-significant-bit function. The LSBs compatible algorithm of sequential pixels with message bits is presented in Figure 3.

Input: Assume y_i, y_{i+1} are the values at the position i, $i+1$ in the sequential pixels and s_i, s_{i+1} are the secret message bits, respectively.

Output: y_i', y_{i+1}' are stego pixels at the position i and $i+1$, respectively.

if $LSB(y_i) \neq s_i$
{
  $y_{i+1}' = y_{i+1}$
  if $XF(y_{i+1}') = s_{i+1}$
    $y_i' = y_i - LSB(y_i) + \overline{LSB(y_i)}$
  else
    if $LSB\left(\left\lfloor \frac{y_i - 1}{2} \right\rfloor\right) = LSB\left(\left\lfloor \frac{y_i}{2} \right\rfloor\right)$
      $y_i' = y_i + 1$
    else
      $y_i' = y_i - 1$
}

Figure 3. Matching algorithm for pixel and message bit

2.3 Yang et al.'s user-friendly image sharing scheme

In their basic (2, n) user-friendly image sharing using polynomials with different primes [15], the secret image is first divided into 2-pixel non-overlapping blocks. Then the dealer D choose a set of four primes $\{p_1, p_2, p_3, p_4\}$ satisfied by $p_1 < p_2 < p_3 < p_4 \leq 251$. In the sharing phase, each block of the secret image is encoded into shadows based on Shamir's polynomial function with different prime numbers. The prime number used to encode a block is determined according to the maximum distance in pixels of block with the last pixel in the previous block. For example, (P_{n-2}, P_{n-1}) and (P_n, P_{n+1}) are the previous and current two-pixel blocks, respectively. The pixel $P_{\max}$ leading to the maximum distance from the current block to the previous block is defined as in (4):

$$P_{\max} = \begin{cases} P_n & \text{if } |P_n - P_{n-1}| > |P_{n+1} - P_{n-1}| \\ P_{n+1} & \text{otherwise} \end{cases}. \quad (4)$$

The prime number is then determined according to $P_{\max}$. Additionally, the prime number for encoding the block needs to remember which is used to decode the block in the recovering phase. In each two-pixel block, the LSBs are adjusted as (00), (01), (10) or (11) to indicate the prime number p_1, p_2, p_3, or p_4, respectively, used in decoding the next block. The LSB of each pixel in the previous block is modified as follows:

$$\begin{cases} LSB(P_{n-2}) = 0,\ LSB(P_{n-1}) = 0 \text{ for } |P_{\max} - P_{n-1}| \leq (p_1 - 1)/2 \\ LSB(P_{n-2}) = 0,\ LSB(P_{n-1}) = 1 \text{ for } (p_1 - 1)/2 < |P_{\max} - P_{n-1}| \leq (p_2 - 1)/2 \\ LSB(P_{n-2}) = 1,\ LSB(P_{n-1}) = 0 \text{ for } (p_2 - 1)/2 < |P_{\max} - P_{n-1}| \leq (p_3 - 1)/2 \\ LSB(P_{n-2}) = 1,\ LSB(P_{n-1}) = 1 \text{ for } (p_3 - 1)/2 < |P_{\max} - P_{n-1}| \leq 250 \end{cases} \quad (5)$$

Subsequently, Shamir's polynomial function rewritten as in Equation (6) is applied on the current block.

$$S(x) = (a_0 + x \times a_1) \bmod p_j \quad \textit{for } j = 1, 2, 3, 4, \quad (6)$$

Where the prime number p_j is determined by Equation (5), and the coefficients a_0 and a_1 are computed asfollows:

$$\begin{cases} a_0 = (P_n - P_{n-1}) + (p_j - 1)/2 \\ a_1 = (P_{n+1} - P_{n-1}) + (p_j - 1)/2 \end{cases} \quad \textit{for } j = 1, 2, 3, \quad (7)$$

$$\begin{cases} a_0 = \lfloor (P_n - P_{n-1})/2 \rfloor + (p_j - 1)/2 \\ a_1 = \lfloor (P_{n+1} - P_{n-1})/2 \rfloor + (p_j - 1)/2 \end{cases} \quad \textit{for } j = 3 \cdot \quad (8)$$

Finally, shadow pixels $\hat{P}_i$ ($1 \le i \le n$) can be found by locating the value that are closest to the average value of pixels in the current block, but also satisfy the constraint $\hat{P}_i \bmod p_j = s_i$.

The recovering procedure starts by determining the prime number from the previously recovered block and shadow pixels. Then, using Lagrange interpolation recovers coefficients a_0 and a_1. After that, the original pixels P_n and P_{n+1} are restored through Equation (7) and (8).

III. PROPOSED SCHEME

Details of the proposed scheme are given in this section. There are two phases: Sharing and retrieving phase.

1.3. Sharing phase

Sharing phase consists of four sub-procedures that may be summarized as in Figure 4. Classification of primes is the first procedure used to indentify prime number for encoding a block. Indications of primes are embedded in secret image on using LSB matching function in the second procedure. Modified secret image is partitioned into shadows by sharing algorithm which is presented in the third procedure. Because Shamir's secret sharing method is employed, shadows look like noisy images. Thus, to produce user-friendly share images, an association between the shadow value and the corresponding average value of block is executed in the fourth procedure.

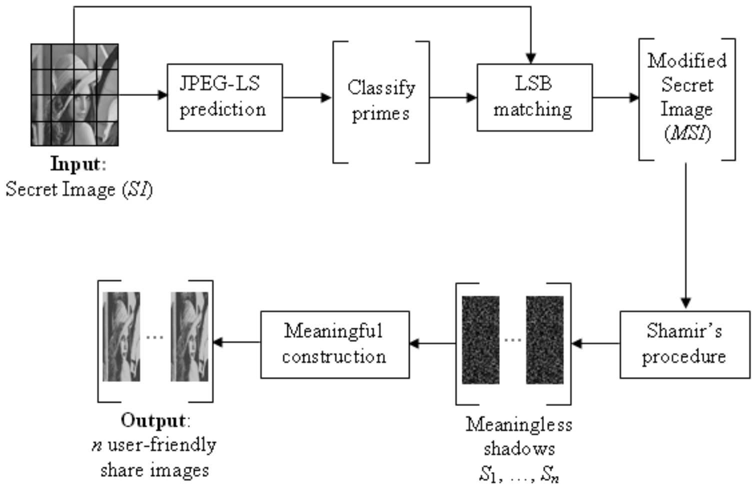

Figure 4. Flowchart of the sharing phase

1.3.1. Classification of primes procedure

To share an secret image (*SI*) size of ($M\times N$) pixels by using the proposed user-friendly (t, n)-threshold sharing system with various prime numbers, The dealer D first selects 2^t prime numbers (denoted as p_1, p_2, …, p_{2^t}) such that $p_1 < p_2 < \cdots < p_{2^t} \le 251$. These prime numbers used in encoding t-pixel non-overlapping blocks are classified according to the maximum distance of block in pixels. In contrast to scheme in [15], the proposed scheme uses the JPEG-LS prediction technique [18] to predict value of each pixel depending on the predictive pattern of pixels as in Figure 1 and Equation (2). By combining the predicted values and the current pixel value, the maximum distance of the m-th block (denoted as d_m) is defined as in (9):

$$d_m = \max(|P_m^i - X_m^i|) \quad \text{for } i = 1, 2, \cdots, t, \qquad (9)$$

Where P_m^i is the pixel value at the position i-th in the m-th block; X_m^i is the predicted value of pixel P_m^i. Next, the procedure is continued by indentifying the prime number for block. The criterion to determine prime number for the m-th block, called p_m , is defined as follows:

$$p_m = \begin{cases} p_1 & \text{if } d_m < (p_1 - 3)/2, \\ p_i & \text{if } (p_{i-1} - 3)/2 \le d_m < (p_i - 3)/2, \text{and } 2 \le i < 2^t, \\ p_{2^t} & \text{otherwise.} \end{cases} \qquad (10)$$

This example demonstrates the procedure, in which a (2, 4)-threshold scheme with four (2^2) chosen primes {p_1, p_2, p_3, p_4} = {17, 61, 131, 251}. A secret image *SI* and its predicted values are shown in Figure 5.

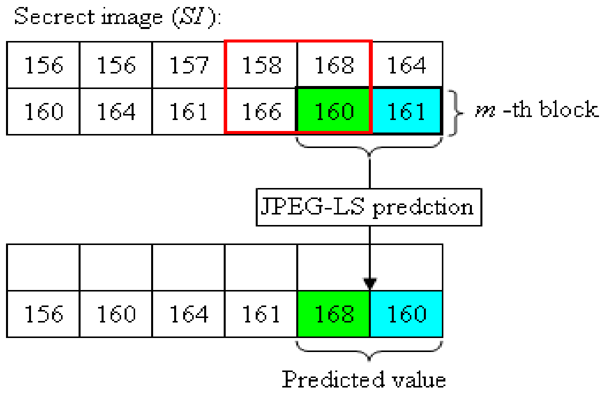

Figure 5. An example of a secret image SI and its predicted value

According to the Figure 5 and Equation (9), value d_m is computed (d_m = max (|160-168|, |161-160|) = 8). Following the Equation (10), the prime number in encoding *m*-th block is determined $(p_m = p_2 = 61)$.

1.3.2. Embedding the prime indicators procedure

Since Shamir's framework is used in the second phase (recovering phase), the prime numbers necessitate to remember for later use. The prime number is encoded as {(0…00), (0…01)… or (1…11)} to indicate the prime numbers {p_1, p_2… or p_{2^t} }. Other word, the indicator id_m of prime p_m is defined as Equation (11):

$$id_m = \begin{cases} (\overbrace{0\cdots00}^{t}) \text{ for } p_m = p_1 \\ (0\cdots01) \text{ for } p_m = p_2 \\ \vdots \\ (1\cdots11) \text{ for } p_m = p_{2^t} \end{cases} \quad (11)$$

This procedure would be continued by implementing the embedding indicators of primes into *SI* based on using LSB matching function [23], with the result that a matched secret image (*MSI*) is generated. In that, the LSBs of pixels in block indicate the prime number which is used in decoding the next block in the recovering phase. In other word, LSBs of pixels in the previous block of *m*-th block are compatible with the indicators of the prime number p_m.
The matching procedure treat prime indications of block as a sequence bits and consider *SI* as one-dimensional rather than two-dimensional array. The flow chart of matching procedure in (*t*, *n*)-threshold proposed scheme is shown as in Figure 6.

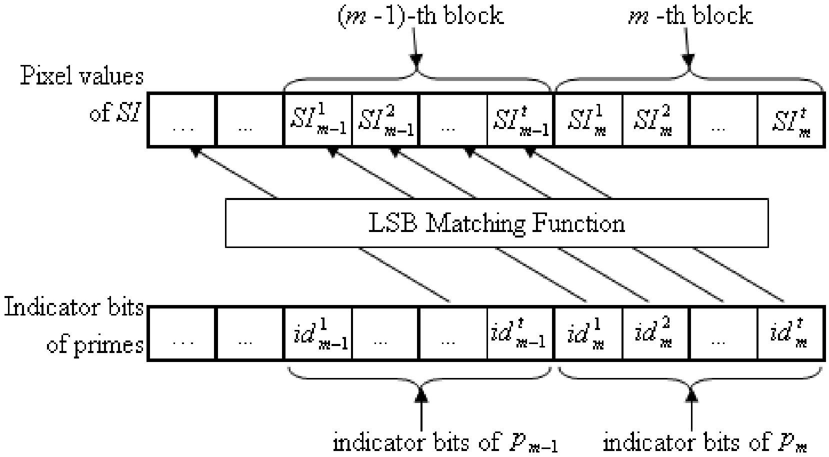

Figure 6. The flow chart of matching procedure in (*t*, *n*)-threshold system

The following example performs the effectiveness of the embedding procedure. As aforementioned, the prime numbers are longer used in the retrieving phase, the indicators of the prime $p_m = 61$ $(id_m = 01)$ are embedded into the *SI* using XOR function. The process of embedding procedure is shown in Figure 7. The indicator bits ($id_m^1 = 0$ and $id_m^2 = 1$) of prime p_m are matched with the LSBs of two pixels in the previous block of *m*-th block, respectively.

As can be seen in the Figure 7, the values of $XF(MSI^1_{m-1})$ and $XF(MSI^2_{m-1})$ are not equal to id^1_m and id^2_m, respectively. And according to the pseudo-code shown in Figure 3, the pixel value MSI^2_{m-1} is set as SI^2_{m-1}, the pixel value MSI^1_{m-1} is set as $SI^1_{m-1}+1$. Generally, in this case, if the conventional LSB replacement method is applied, there are two pixel values in *SI* are modified. In contrast, there is only one pixel in *SI* is adjusted in this method.

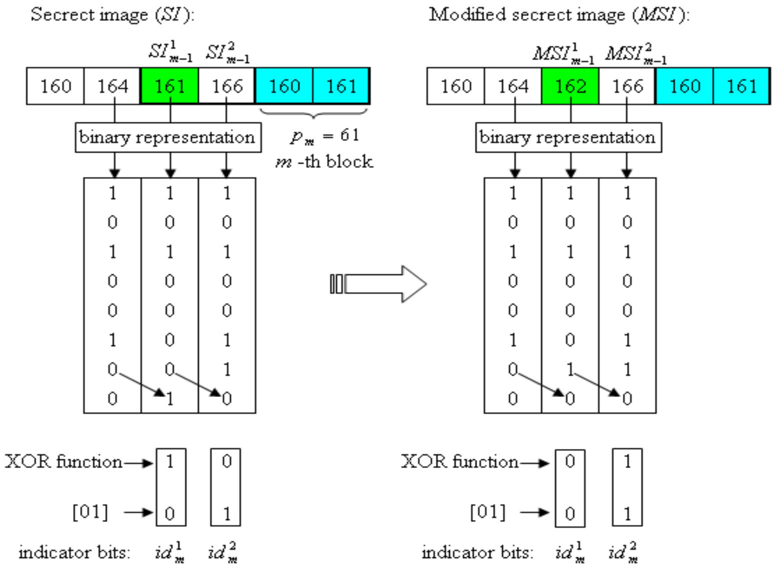

Figure 7. Embedding procedure

1.3.3. Partitioning procedure

Applying the Shamir's polynomial function of degree (t-1), the procedure divides *MSI* into non-overlapping blocks, each of which contains t pixels. Unlike in the original polynomial, coefficients a_1, a_2,…, a_{t-1} are random numbers, the process of partitioning m-th block starts by computing the coefficients through the equation as follows:

$$\begin{cases} f_i = (MSI^i_m - PMSI^i_m) + (p_m - 1)/2 & \text{if } p_m \neq p_{2^t}, \\ f_i = \lfloor (MSI^i_m - PMSI^i_m)/2 \rfloor + (p_m - 1)/2 & \text{otherwise.} \end{cases} \quad \text{(for } 1 \le i \le t\text{)}, \qquad (12)$$

where $PSMI^i_m$ is predicted value of i-th pixel MSI^i_m in m-th block. It means that the JPEG-LS prediction technique is one more time adopted on the *MSI* image. Then the coefficients f_i play as input values of the function:

$$S_m(x) = \sum_{i=1}^{t} f_i x^{i-1} \bmod p_m \qquad (13)$$

Now, n input valuesx_j ($1 \le j \le n$) are chosen freely and all x_j must be distinguished from each other. With each chosen x_j, a corresponding value of $S_m(x_j)$ is calculated by Equation (13).

Assume after the previous Subsection, the modified secret image (*MSI*) is obtained and apply the JPEG-LS prediction technique on *MSI* given as in Figure 8.

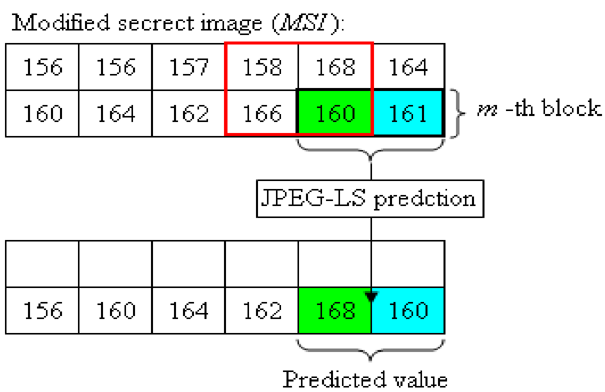

Figure 8. Anexample of MSI and its predicted values

To encodes m-th block of *MSI*, the Equation (13) is implemented with the determined prime number p_m=61, in which the coefficients f_1 and f_2 equal to 22 and 31 (according to Equation (12), $f_1 = (160-168)+(61-1)/2 = 22$,

$f_2 = (161-160)+(61-1)/2 = 31$). For example ($x_1$, x_2, x_3, x_4) = {1, 2, 3, 4} corresponds to the order of four involved participants. Block m is encoded into four packets: $S_m(x_1)=53, S_m(x_2)=23$, $S_m(x_3)=54$ and $S_m(x_4)=23$.

1.3.4. User-friendly Shadows construction procedure

Owing to using Shamir's framework, generated shadow images look like noise which may attract the attention of the malicious users during transmission. Thus, this Subsection generates user-friendly shadow pixels $\tilde{P}_m^j$ distributed to the j-th participant (1≤ j≤ n). The process produced pixel $\tilde{P}_m^j$ is an association between the value $S_m(x_j)$ and$m$-th block. The value of $\tilde{P}_m^j$ have to satisfy two criteria:

(i) $\tilde{P}_m^j = p_m \times k + S_m(x_j)$ where k is a non-negative integer,

(ii) $\tilde{P}_m^j$ is the closest value to the average value of m-th block.

Finally, the obtained value $\tilde{P}_m^j$ is delivered to the j-th participant.

Following the above example, the process calculates the shadow $\tilde{P}_m^1$ for the participant 1 combined $S_m(x_1)=53$ with the average value of pixels in block m ($\lfloor(160+161)/2\rfloor$=160). The value $\tilde{P}_m^1$ is chosen 175 because {53, 114, 175, 236} is the set of numbers which satisfies the form $61\times k+53$.

1.4. Retrieving phase

Retrieving phase consists of two sub-procedures: reconstruction and recovering which is summarized as in the Figure 9. The first sub-procedure is used to join t shadow images based on Lagrange interpolation. Second, recovering procedure reveals the original secret image. The details of these phase is presented below:

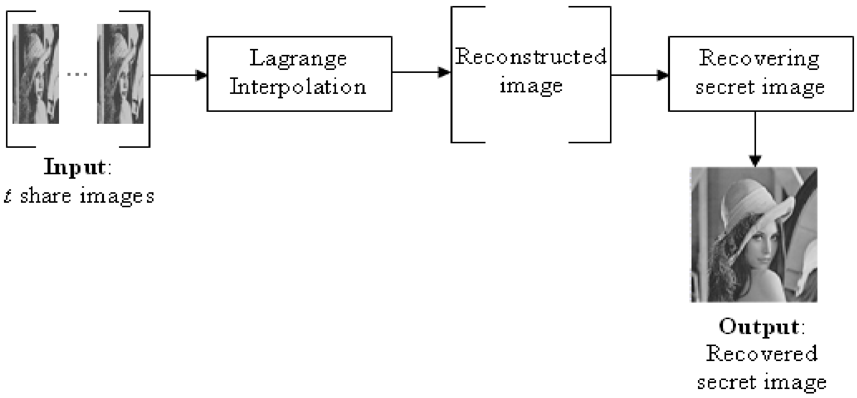

Figure 9. Flowchart of the retrieving phase

First of all, the reconstruction procedure is started by collecting the shadows from the authorized participants and determining the prime number from the initial shadow pixels. Any t of them will be sufficient to reconstruct image. According to the Equation (1), the coefficients f_1, f_2,…, f_t of the Equation (13) are solved. After that, the recovering original image is restored through the Equation (12).

IV. DISCUSSIONS AND EXPERIMENTAL RESULTS

Results of the tests executed to demonstrate the feasibility of the proposed scheme are presented in this section. A set of images of size 512×512 shown in Figure 10 is used as test images.

Figure 10. A set of test images of size 512×512: (a) Lena; (b) Jet; (c) Baboon

www.ijmer.com Vol.3, Issue.1, Jan-Feb. 2013 pp-139-148 ISSN: 2249-6645

Proposed method uses a (t, n)-threshold sharing with different prime numbers to generate shadows. According to the Subsection 3.1.4, obtained value $\tilde{P}_m^j$ is similarity to the corresponding pixel in m-th block of *MSI*. Furthermore, the size of each shadow image is 1/t the size of the secret image. On the other hand, according to the ensuring of the Shamir' method, even if (n–t) shadows are lost or corrupted when transferred or stored, it is still possible to recover the secret image.

Therefore, it can be said that the characteristics of user-friendly image sharing are satisfied in the proposed scheme. Figure 11 presents the results obtained using a (2, 4)-threshold scheme with chosen primes $\{p_1, p_2, p_3, p_4\} = \{17, 61, 131, 251\}$. The scheme encodes images shown in Figure 10 into four shadows separately; the obtained shadows in Figure 11 (a1)-(a4), (b1)-(b4), (c1)-(c4) are the results of the Figure 10 (a), (b), (c), respectively.

Figure 11. Obtained shadows from (2, 4)-threshold user-friendly image sharing scheme with four different primes $\{p_1, p_2, p_3, p_4\} = \{17, 61, 131, 251\}$

Notably, each shadow image looks like a shrunken version of the original image. So, from the view point of image database manager, this similarity allows users to recognize each shadow image with its original image without any computation. Of course, this characteristic only provides to distinguish content among multiple shadow images. Because, too much detailed information of the secret image is omitted, illegal users cannot obtain a high-quality image from taking a single shadow image. The image quality of the reconstructed image is the peak signal-to-noise ration (*PSNR*) and is defined as

$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right)\text{DB} \tag{14}$$

where MSE is the mean squared error between original image and recovered image. MSE is evaluated as follow:

$$MSE = \frac{1}{M \times N}\sum_{i}^{M}\sum_{j}^{N}(p_{i,j} - p'_{i,j})^2 \tag{15}$$

Where (M×N) is the size of an image; $p_{i,j}$ is the original pixel value; and $p'_{i,j}$ is the recovered pixel value.

Figure 12 shows images (a), (b) and (c) obtained by direct expansion images of the shadow images Figure 11 (a1), (b1) and (c1) to the size of the original image. The *PSNR* values between these images and the original image are 27.62dB, 27.79 and 21.46dB respectively.

(a) $PSNR = 27.62$dB (b) $PSNR = 27.79$dB (c) $PSNR = 21.46$dB

Figure 12. Results obtained by expanding a single shadow image to the size of original image: images (a), (b) and (c) obtained from Figure 11 (a1), (b1) and (c1)

In the revealing phase, proposed scheme uses Lagrange's interpolation to restore the original image. Any two of shadow images are gathered, the original image can be reconstructed. Figure 13 (a), (b) and (c) are recovered from any two shadows of Figure 11 (a1)-(a4), (b1)-(b4) and (c1)-(c4), respectively.

(a) $PSNR = 52.91$dB (b) $PSNR = 52.90$dB (c) $PSNR = 52.89$dB

Figure 13. Reconstructed images from any 2 shadow images

During the embedding procedure, in the sharing phase, hides indicators of prime numbers into secret image on using LSB matching function [23], which is used to limit the modification of the secret image. LSB of each pixel is matched with the indicator based on XOR function (Equation (3)). Yang *et al.*'s method uses the conventional LSB replacement which embeds indicators of primes into secret image by replacing the LSBs of the secret image with indicator bits. Besides, in the scheme of Yang *et al*., the determination of prime numbers for a block fully depends on the last pixel of the previous block. The proposed scheme, by contrast, uses the JPEG-LS prediction technique that estimates accurately predicted pixel value. The proposed method rarely uses the equation like Equation (8) in Equation (12). Therefore, the proposed method can obtain better quality of the recovered images. Table 1 presents the image qualities of (2, 4)-threshold of the proposed scheme compared with those for two previous published user-friendly image sharing method. The values in Table 1 are the *PSNR* values of the recovered images. It reveals that the proposed scheme can reconstruct the original image with a higher quality than that of previous schemes.

Table 1. Qualities of the recovered imagesbased on different user-friendly image-sharing schemes in a set of test images

	Thien *et al*. [14]	Yang *et al.* [15]	Proposed scheme
Lena	37.37	50.53	52.91
Jet	39.19	49.76	52.90
Baboon	34.75	49.17	52.89

Table 2. Comparison of the proposed scheme with others in terms of the average *PSNRs* value of expanded shadow images

	Thien *et al*. [14]	Yang *et al.* [15]	Proposed scheme
Lena	24.80	23.32	27.62
Jet	25.65	23.14	27.79
Baboon	20.55	18.52	21.46

The values shown in the Table 2 are the average *PSNRs* values of the expanded image and the original image by the proposed method and other schemes. Notably, the *PSNR* value of each shadow image is slightly higher than that of the schemes in [14] and [15]. The moderate quality of shadow images in the proposed scheme is convenient in images management, rather than recovering a high-quality image based on image-interpolation technique by illegal users.

V. CONCLUSIONS

This paper proposed a novel user-friendly image sharing method with different prime numbers that provides shadow images among authorized participants. Each participant receives a meaningful image looking like a shrunken version of the original image. The proposed scheme is based on Yang *et al.*'s framework. However, in this paper, the prime number is determined according to the predicted value which leads to the small prime number in encoding block. In addition, the indicators of prime numbers are embedded into secret image based on the XOR function that limits modification of original pixel value, so that a better visual quality of the reconstructed image is guaranteed in the proposed scheme.

References

[1] W. Bender, D. Gruhl, N. Morimoto and A. Lu, “Techniques for data hiding,” IBM Systems Journal, Vol. 35, pp. 312-336, 1996.
[2] C. T. Hsu and J. L. Wu, “Hiding digital watermarks in images,” IEEE Transactions of Image Processing, Vol. 8, pp. 58-68, 1999.
[3] D. C. Wu and W. H. Tsai, “Data hiding in images via multiple-based number conversion and lossy compression, “IEEE Transactions on Consumer Electronics Vol. 44, pp. 1406-1412, 1998.
[4] A. Shamir, “How to share a secret,” Communications of the ACM, Vol. 22, no. 11, pp. 612–613, 1979.
[5] G. R. Blakley, “Safeguarding cryptographic keys,” AFIPS 1979 National Computer Conference, Vol. 48, pp. 313–317, 1979.
[6] C. C. Thien, J. C. Lin, “Secret image sharing,” Computers and Graphics, Vol. 26, pp. 765–770, 2002.
[7] S. K. Chen and J. C. Lin, “Fault-tolerant and progressive transmission of images,” Pattern Recognition, Vol. 38, pp. 2466-2471, 2005.
[8] C. C. Lin, W. H. Tsai, “Secret image sharing with steganography and authentication,” Journal of Systems and Software, Vol. 73, pp. 405-414, 2004.
[9] C. N. Yang, T. S. Chen, K. H. Yu, C. C. Wang, “Improvements of image sharing with steganorgraphy and authentication,” Journal of Systems and Software, Vol. 80, pp. 1070-1076, 2007.
[10] C. C. Chang, Y. P. Hsieh, C. H. Lin, “Sharing secrets in stego images with authentication,” Pattern Recognition, Vol. 41, pp. 3130-3137, 2008.
[11] P. Y. Lin and C. S. Chan, “Invertible secret image sharing with steganography,” Pattern Recognition Letters, Vol. 31, pp. 1887-1893, 2010.
[12] Z. Eslami and J. Z. Ahmadabadi, “Secret image sharing with authentication-chaining and dynamic embedding,” Journal of Systems and Software, Vol. 84, pp. 803-809, 2011.
[13] M. Ulutasa, G. Ulutas and V. V. Nabiyeva, “Medical image security and EPR hiding using Shamir's secret sharing scheme,” Journal of Systems and Software, Vo. 84, pp. 341-353, 2011.
[14] C. C. Thien and J. C. Lin, “An image-sharing method with user-friendly shadow images,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, pp. 1161–1169, 2003.
[15] C. N. Yang, K. H. Yu, and R. Lukac, “User-friendly image sharing using polynomials with different primes,” International Journal of Imaging Systems and Technology Vol. 17, pp. 40–47, 2007.
[16] Zhicheng Ni, Yun Q. Shi, Nirwan Ansari, Wei Su, Qibin, and Xiao Lin, “Robust lossless image data hiding designed for semi-fragile image authentication,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 18, pp. 497–509, 2008.
[17] Jun Tian, “Reversible data embedding using a difference expansion,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, pp.890–896, 2003.
[18] M. Weinberger, G. Seroussi, G. Sapiro, “The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS,” HPL-98-193, 1998.
[19] B. Carpentieri, M. J. Weinberger and G. Seroussi, “Lossless compression of continuous-tone images,” Proceeding of the IEEE, Vol. 88, 2000.
[20] J. Jiang, B. Guo and S. Y. Yang, “Revisiting the JPEG-LS prediction scheme,” IEE Proceedings- Vision, Image and Signal Processing, Vol. 147, pp. 575-580, 2000.
[21] S. Bedi, E. A. Edirisinghe and G. Grecos, “Improvement to the JPEG-LS prediction scheme,” Journal of Image and Vision Computing, Vol. 22, pp. 9-14, 2004.
[22] J. Mielikainen, “LSB matching revisited,” IEEE Signal Processing Letters, Vol. 13, pp. 285-287, 2006.
[23] C. S. Chan, “On using LSB matching function for data hiding in pixels,” Fundamenta Informaticae, Vol.96, pp. 49-59, 2009.