



ELSEVIER

Information Sciences 111 (1998) 335–345

INFORMATION
SCIENCES
AN INTERNATIONAL JOURNAL

Sharing secret images using shadow codebooks

Chin-Chen Chang ^{a,*}, Ren-Junn Hwang ^b

^a *Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan 621, Republic of China*

^b *Department of Computer Science and Information Science, Chung Cheng Institute of Technology, Taoyuan, Taiwan 330, Republic of China*

Received 1 December 1996; received in revised form 25 December 1997; accepted 5 March 1998

Abstract

A practical specific (r, n) threshold scheme for secret digital images is proposed in this paper. By this scheme, we divide a secret digital image into n pieces and distribute them to n participants. The secret digital image can be reconstructed when and only when r or more participants cooperate for it. The time complexity of this proposed scheme is independent of the size of the secret image. In this scheme, the setup phase is bounded by $i \times j \times \lfloor r - 1/2 \rfloor + 2$ module multiplications plus $i \times j \times (r - 1)$ module additions, where $i \times j$ is the size of the codebook, except the time needed for generating the codebook. The reconstructing phase is bounded by $O(r \log^2 r)$. This scheme also achieves perfect secrecy. © 1998 Elsevier Science Inc. All rights reserved.

Keywords: Image compression; Lagrange's interpolation polynomial; Perfect secrecy; Threshold scheme; Vector quantization

1. Introduction

“How to keep secrets?” is a very significant and practical problem in today's information-based society. Sometimes secrets should not be always protected by only one person; they can be very easily lost, destroyed, or modified if they are held by only one person. Blakley [1] and Shamir [2] first proposed a concept,

* Corresponding author. Tel.: 886 5 2720859; fax: 886 5 2720859; e-mail: ccc@cs.ccu.edu.tw

named (r, n) threshold scheme, to solve this problem independently. An (r, n) threshold scheme consists of a trusted dealer and n participants. The dealer divides the secret document into n shadows and distributes them to n participants. The secret document cannot be reconstructed unless some specific groups of more than $r - 1$ out of n shadow holders pool their shadows. An (r, n) threshold scheme has the following features [3]:

1. The secret is divided into n shadows.
2. Any r or more shadows can be used to reconstruct the secret.
3. Any $r - 1$ or less shadows reveal no knowledge about the secret.

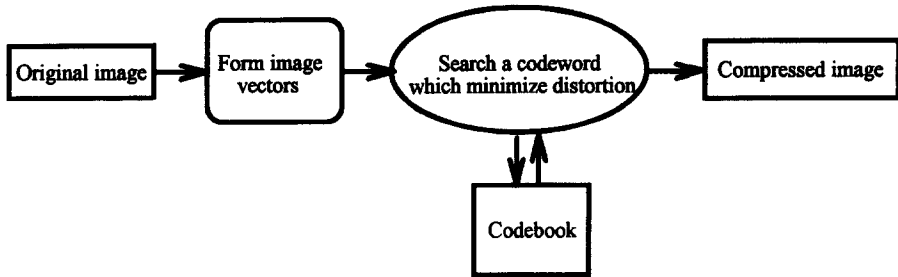
Threshold schemes are mainly used to protect secrets from being lost, destroyed or modified. In our society, there are many “true-to-life” examples taking place in the military, banks, and technical companies. Practically speaking, the secret “document” could be in many forms, for example, text files, photographs, blueprints, pictures, passwords, and encryption/decryption keys.

After Blakely [1] and Shamir [2] proposed the (r, n) -threshold scheme, there have been hundreds of technical reports exploring this problem. However, these schemes are only suitable for digital data such as text files, passwords, and encryption/decryption keys, while the type of photographs, blueprints, and pictures has to be in the form of digital images. A large number of bits are typically required to represent even a single digital image. It is impractical to apply the traditional threshold scheme to share a secret digital image directly. So, the development of a specific (r, n) threshold scheme for digital images is important and necessary.

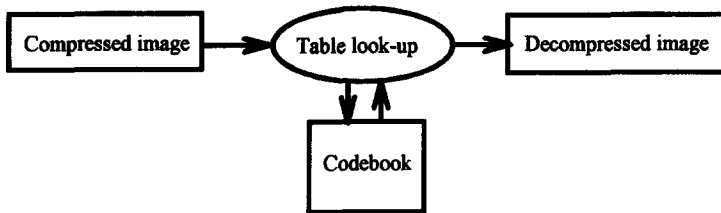
In order to utilize digital images effectively, we need specific methods to reduce the number of bits required for their representation. The technique of digital image processing that concerns this problem is named image compression [4]. A wide range of image compression methods have been developed over the years. Among them, vector quantization (VQ) is one of the most popular image compression methods. It has been shown to be simple and efficient by many researchers [4–15].

VQ is a method of image compression that can be viewed as a process of pattern recognition. An input pattern is approximated by one of the standard patterns from a predetermined table. The standard pattern table is named a VQ codebook. An image is divided into a set of input patterns, and each input pattern is matched with a codeword of the codebook. The encoder and the decoder use the index of this codeword respectively to compress and decompress. A block diagram of the basic VQ structure is shown in Fig. 1. Here, in this paper, we shall propose a specific (r, n) threshold scheme based on VQ for secret digital images.

In the rest of this paper, Section 2 introduces our (r, n) threshold scheme for single digital images. The extension of our (r, n) threshold scheme for multiple digital images is described in Section 3. Section 4 shows our experimental results. The computation complexity and security analysis are discussed in Section 5. Finally, our conclusions are stated in Section 6.



(a) The compressing phase of VQ



(b) The decompressing phase of VQ

Fig. 1. VQ block diagram: (a) the compressing phase of VQ; (b) the decompressing phase of VQ.

2. An (r, n) threshold scheme based on VQ

In our specific (r, n) -threshold scheme, we compress the secret image with the VQ compression method. The participants do not share the secret image directly; instead, they share the codebook which is used to compress the secret image. In the VQ compressed method, we shall emphasize that the compressed digital image is assembled by the indices of the codebook, whose codewords approximate the input patterns. Although these n participants share the codebook only, the secret image can be recovered easily when the correct codebook is recovered. Note that the secret image cannot be decompressed or reconstructed from the compressed code without the correct codebook.

In varieties of VQ compression methods, codewords are typically generated using a training set of images that are representatives of the images to be encoded. Hence, most of the image features which are specific to the particular image are adequately represented by the codewords. The optimal codebook would be generated using the image itself as the training set. This codebook is named a local codebook, and it usually results in good performance for moderate codebook sizes [4]. The quality of the secret image is a very crucial factor, so we assume that the compression of the secret image is conducted with the local codebook to ensure good quality.

The core technique of our (r, n) threshold scheme is the sharing of the original local codebook S among these n participants securely. Since our proposed scheme is based on Galois field $GF(p)$, the dealer first generates a modified codebook S' based on the original codebook S and then discards S when some of the components of S are not suitable for the arithmetic operation based on $GF(p)$. The dealer then divides the modified codebook S' into n shadow codebooks $B^{(1)}, B^{(2)}, \dots, B^{(n)}$ and distributes them to n participants secretly in such a way that:

1. the modified codebook S' can be reconstructed easily using any r or more shadow codebooks $B^{(t)}$ s;
2. nobody can get any knowledge of the original codebook S or the modified codebook S' using any $r - 1$ or less shadow codebooks $B^{(t)}$ s.

In other words, if r or more participants pool their shadow codebooks, they can decompress the shared secret image. Any $r - 1$ or fewer participants cannot do anything about it.

Let the original codebook S contain i codewords and let each codeword have j components. Each component of a codeword is a pixel value. Also, let each of the pixel value be no less than 0 and no greater than $k - 1$. The codebook can be viewed as an $i \times j$ matrix, and each component ranges from 0 to $k - 1$. Our goal is to generate n shadow codebooks $B^{(i)}$'s for the n participants such that if and only if there are r or more participants willing to cooperate with one another, they can generate the modified codebook S' with their shadow codebooks $B^{(i_1)}, B^{(i_2)}, \dots, B^{(i_r)}$. As Fig. 2 shows, there are two phases in our scheme: the setup phase and recovery phase. We shall describe these two phases in the paragraphs right below.

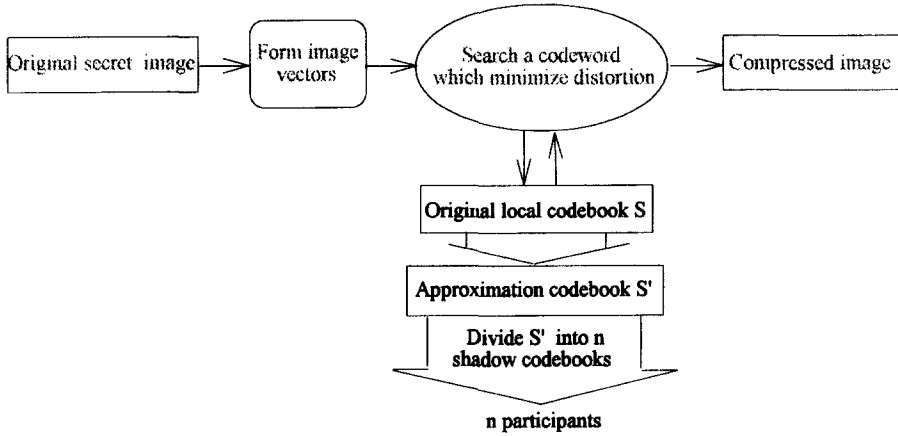
2.1. Setup phase (by the dealer)

Firstly, the dealer selects an integer p , which is the largest prime integer in the range $[0, k - 1]$. The modified codebook S' is generated from the original local codebook S following the rule:

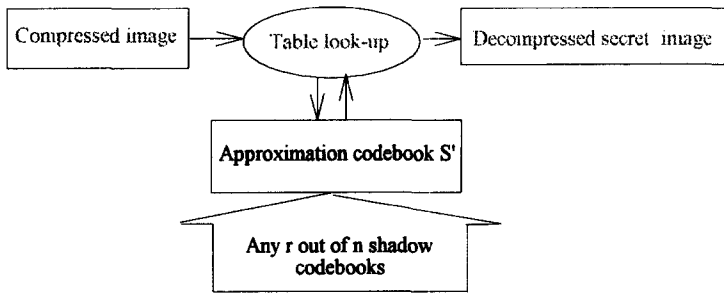
$$s'_{ij} = \begin{cases} s_{ij} & \text{if } s_{ij} < p, \\ p - 1 & \text{if } s_{ij} \geq p, \end{cases} \quad (1)$$

where s'_{ij} and s_{ij} are the components of S' and S , respectively. Later in the experimental results in Section 4, we shall see that there can only be few pixels of many local codebooks whose values are not smaller than the selected p . So the modified codebook S' is like the original codebook S .

Next, the dealer divides the modified codebook S' into n shadow codebooks $B^{(i)}$'s, for n participants. He/she randomly selects n integers, $\alpha_1, \alpha_2, \dots, \alpha_n$, from $[0, p)$, where $\alpha_i \neq \alpha_j$ for all $i \neq j$. The dealer then generates these shadow codebooks based on these n integers and the following formula.



(a) The dividing phase of our scheme



(b) The reconstructing phase of our scheme

Fig. 2. Our (r, n) threshold scheme diagram: (a) the dividing phase of our scheme; (b) the reconstructing phase of our scheme.

$$F(x) = S' + M^{(1)} \times x + M^{(2)} \times x^2 + \dots + M^{(r-1)} \times x^{r-1} \bmod p, \quad (2)$$

where $M^{(t)} (t = 1, 2, \dots, r-1)$ is a randomly chosen matrix with i rows and j columns, and each component of these matrices ranges from 0 to $p-1$. We use modular arithmetic instead of real arithmetic as is Shamir's idea [2]. The set in which all integers modulo a prime number p forms a Galois field $GF(p)$. In this field, we can reconstruct the polynomial $F(x)$ using the interpolation method in the recovery phase. In Formula (2), the $i \times j$ matrix $M^{(t)}$ module p is defined to be a new $i \times j$ matrix whose elements are equal to their corresponding elements in $M^{(t)}$ module p . Eqs. (3)–(5) show the shadow codebooks of Participants 1, 2, ..., and n :

$$B^{(1)} = F(\alpha_1) = S' + M^{(1)} \times \alpha_1 + M^{(2)} \times \alpha_1^2 + \cdots + M^{(r-1)} \times \alpha_1^{r-1} \bmod p, \quad (3)$$

$$B^{(2)} = F(\alpha_2) = S' + M^{(1)} \times \alpha_2 + M^{(2)} \times \alpha_2^2 + \cdots + M^{(r-1)} \times \alpha_2^{r-1} \bmod p, \quad (4)$$

$\vdots$

$$B^{(n)} = F(\alpha_n) = S' + M^{(1)} \times \alpha_n + M^{(2)} \times \alpha_n^2 + \cdots + M^{(r-1)} \times \alpha_n^{r-1} \bmod p. \quad (5)$$

Finally, the dealer distributes a codebook $B^{(h)}$ to each Participant h secretly.

Each participant, say Participant h , holds a shadow codebook $B^{(h)}$ and a shadow parameter α_h . The size of each shadow codebook is equal to the original local codebook, and the value of each component ranges from 0 to $p - 1$.

2.2. Recovery phase (by any r out of n participants)

In this phase, any r or more participants pool their shadow codebooks and shadow parameters to reconstruct the modified codebook S' . With this reconstructed modified codebook, they can decompress the secret image. How can these r participants reconstruct the modified codebook S' ? By the system of Eqs. (3)–(5), we know that each pair of a shadow codebook $B^{(h)}$ and its corresponding shadow parameter α_h satisfy the following relations:

$$b_{ij}^{(1)} = s'_{ij} + m_{ij}^{(1)} \times \alpha_1 + m_{ij}^{(2)} \times \alpha_1^2 + \cdots + m_{ij}^{(r-1)} \times \alpha_1^{r-1} \bmod p, \quad (6)$$

$$b_{ij}^{(2)} = s'_{ij} + m_{ij}^{(1)} \times \alpha_2 + m_{ij}^{(2)} \times \alpha_2^2 + \cdots + m_{ij}^{(r-1)} \times \alpha_2^{r-1} \bmod p, \quad (7)$$

$\vdots$

$$b_{ij}^{(n)} = s'_{ij} + m_{ij}^{(1)} \times \alpha_n + m_{ij}^{(2)} \times \alpha_n^2 + \cdots + m_{ij}^{(r-1)} \times \alpha_n^{r-1} \bmod p, \quad (8)$$

where s'_{ij} is the ij th component of S' , $b_{ij}^{(t)}$ is the ij th component of $B^{(t)}$ ($t = 1, 2, \dots, n$), and $m_{ij}^{(k)}$, the ij th component of $M^{(k)}$ ($k = 1, 2, \dots, r - 1$). We can also generalize these relations by a polynomial of degree $r - 1$

$$f_{ij}(x) = s'_{ij} + m_{ij}^{(1)} \times x + m_{ij}^{(2)} \times x^2 + \cdots + m_{ij}^{(r-1)} \times x^{r-1} \bmod p, \quad (9)$$

and this polynomial satisfies

$$b_{ij}^{(t)} = f_{ij}(\alpha_t) = s'_{ij} + m_{ij}^{(1)} \times \alpha_t + m_{ij}^{(2)} \times \alpha_t^2 + \cdots + m_{ij}^{(r-1)} \times \alpha_t^{r-1} \bmod p, \\ \text{for } t = 1, 2, \dots, n.$$

It is obvious that Polynomial (9) above passes through n points $\{(\alpha_1, b_{ij}^{(1)}), (\alpha_2, b_{ij}^{(2)}), \dots, (\alpha_n, b_{ij}^{(n)})\}$ on the 2-dimensional plane of x and $f(x)$.

Any component of the modified codebook, s'_{ij} , is a constant item of this specific polynomial. Although we do not know the polynomial $f_{ij}(x) = s_{ij} + m_{ij}^{(1)} \times x + m_{ij}^{(2)} \times x^2 + \dots + m_{ij}^{(r-1)} \times x^{r-1} \bmod p$, we can use Lagrange's interpolation method [5,16] to reconstruct this $(r-1)$ -degree-polynomial $f_{ij}(x)$ by interpolating on any r out of these n points $\{(\alpha_1, b_{ij}^{(1)}), (\alpha_2, b_{ij}^{(2)}), \dots, (\alpha_n, b_{ij}^{(n)})\}$ as follows:

$$\begin{aligned} f_{ij}(x) = & b_{ij}^{(h_1)} \frac{(x - \alpha_{h_2})(x - \alpha_{h_3}) \dots (x - \alpha_{h_r})}{(\alpha_{h_1} - \alpha_{h_2})(\alpha_{h_1} - \alpha_{h_3}) \dots (\alpha_{h_1} - \alpha_{h_r})} \\ & + b_{ij}^{(h_2)} \frac{(x - \alpha_{h_1})(x - \alpha_{h_3}) \dots (x - \alpha_{h_r})}{(\alpha_{h_2} - \alpha_{h_1})(\alpha_{h_2} - \alpha_{h_3}) \dots (\alpha_{h_2} - \alpha_{h_r})} + \dots \\ & + b_{ij}^{(h_r)} \frac{(x - \alpha_{h_1})(x - \alpha_{h_2}) \dots (x - \alpha_{h_{r-1}})}{(\alpha_{h_r} - \alpha_{h_1})(\alpha_{h_r} - \alpha_{h_2}) \dots (\alpha_{h_r} - \alpha_{h_{r-1}})} \bmod p. \end{aligned}$$

Then the component of the modified codebook s'_{ij} can be obtained from the constant item. In other words, each component of the modified codebook s'_{ij} can be revealed from the set of r points $\{(\alpha_{h_1}, b_{ij}^{(h_1)}), (\alpha_{h_2}, b_{ij}^{(h_2)}), \dots, (\alpha_{h_r}, b_{ij}^{(h_r)})\}$ using Lagrange's interpolation method. Each Y -coordinate value of these r points, $b_{ij}^{(h)}$, is the ij th component of the h th shadow codebook. The corresponding X -coordinate value α_h is the shadow parameter of the holder h . Finally, the secret image can be decompressed using this reconstructed modified codebook. The compressed code does not lose any information in these two phases, and the differences between the original local codebook and the modified codebook is only few and far between. Thus the PSNR values of the decompressed images which are decompressed with the modified codebook and with the original local codebook respectively are almost equal. We shall show our experimental results in Section 4. And the definition of PSNR will be introduced later in the same section.

3. Extension of our (r, n) threshold scheme for multiple digital images

In some cases, there could be multiple secret images that have different local codebooks shared by n participants. Now, the readers may ask: Can the secret images be decompressed when and only when there are r or more participants cooperating with one another? Using our (r, n) threshold scheme directly, the dealer generates different shadow codebooks for different images. Then each participant will have to hold many different shadow codebooks for these different secret images, which does not seem practical. Here, we extend our simple (r, n) threshold scheme by publishing some information such that these n participants can share these secret images while each of the participants still holds only one shadow codebook and one shadow parameter.

Without loss of generality, we assume that there are n participants sharing z secret images. Each image t corresponds to an original local codebook C_t . Firstly, the dealer generates n shadow codebooks for Image 1's modified

codebook C'_1 as stated above. Secondly, she/he generates $z - 1$ pieces of public information D'_t 's such that $D_t = C_t \oplus C'_1$ for $t = 2, 3, \dots, z$. The expression $W = U \oplus V$ means that the ij th component w_{ij} of W is equal to $u_{ij} \oplus v_{ij}$, where u_{ij} and v_{ij} are the ij th components of U and V , respectively. The operation " $u_{ij} \oplus v_{ij}$ " is defined as the pairwise bit exclusive or of the bit strings of u_{ij} and v_{ij} . Now, in our extended scheme, each participant still holds no more than a shadow codebook and a shadow parameter; what is added is that the dealer publishes information D_t for each secret image t ($t = 2, 3, \dots, z$). Note that no public information is there for Image 1 and that only r or more participants can reconstruct the modified codebook C'_1 .

As stated in Section 2, when and only when any r or more participants pool their shadow codebooks and shadow parameters, the modified codebook C'_1 can be generated. After C'_1 is generated, these participants can evaluate the other codebook C_t by $C_t = D_t \oplus C'_1$. Then the secret image t can be decompressed. Obviously, each participant holds only one shadow codebook, but they can share as many as z secret images in our extended scheme.

4. Experimental results and discussions

All our experiments are performed on a SUN SPARC10 workstation. Seven 512×512 and four 256×256 monochrome images are used in our experiments. Each pixel in these images contains 256 gray levels. The prime number of these experiments is 251. These images are divided into some 4×4 pixel blocks at first. Each block is therefore a 16-dimensional vector. Each image is used as the training set to generate the original codebook by the LBG algorithm [4] with a 0.001 threshold. The quality of the decompressed image is evaluated by the peak signal-to-noise ratio, which is defined as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}} \text{dB}.$$

For an $m \times m$ image, the mean-square error MSE is defined as

$$\text{MSE} = \left(\frac{1}{m}\right)^2 \sum_{i=1}^m \sum_{j=1}^m (x_{ij} - \hat{x}_{ij})^2,$$

where x_{ij} and $\hat{x}_{ij}$ denote the original and the quantized gray levels, respectively.

In our experiments, we find that almost all the pixel values of each image's original local codebook are smaller than 251, except for the Family image. Our experimental results are shown in Table 1. It is obvious that most of the original local codebooks are the same as their corresponding modified codebooks in our experiments. There is only one different pixel-pair between the original local codebook and the modified codebook in the Family image. Amazingly, the PSNR values of the decompressed images that are respectively decom-

pressed with the modified codebook and the original local codebook are equal, which can also be seen from Table 1. In our experiments, all the images' original local codebooks are the same as their corresponding modified codebooks except for the Family image.

5. Computation complexity and security analysis

5.1. Computation complexity

There are two phases in our scheme: the setup phase and the recovery phase. In the setup phase, the dealer has to compress the secret image and generate its codebook S first. The time needed to generate a codebook S grows only like $O(N \log N)$ in training set size by Equity's PNN algorithm [17]. Next, the dealer takes $i \times j$ comparison operation to generate the modified codebook S' , where $i \times j$ is the size of the original codebook S . Then she/he evaluates the polynomial value of order $r - 1$ following Formula (2) for each participant. In the worst case, it takes $i \times j \times \lfloor r - 1/2 \rfloor + 2$ module multiplications and $i \times j \times (r - 1)$ module additions. In the recovery phase, it is required to reconstruct the specific $(r - 1)$ -degree polynomial from the shadow codebooks of any r participants. It takes $O(r \log^2 r)$ to evaluate each component of the modified codebook using Lagrange's interpolation method or Newton's interpolation method [16,19]. Finally, the secret image can be reconstructed based on the modified codebook. The decomposition steps as a whole take a constant period of time to transform each index in the compressed code to its codeword. That is, our (r, n) threshold scheme works the same regardless of the size of the secret

Table 1
Experimental results

Image	The number of pixels whose values are greater than 251	PSNR of the decompressed image (using the original local codebook or the modified codebook)
256 × 256 Barbara	0	25.478
256 × 256 Boat	0	27.352
256 × 256 Family	1	27.034
256 × 256 Peppers	0	28.201
512 × 512 Airplane	0	30.124
512 × 512 Baboon	0	25.068
512 × 512 Farm	0	30.429
512 × 512 Lena	0	31.256
512 × 512 Robot	0	27.495
512 × 512 Landscape	0	28.815
512 × 512 Zelda	0	34.023

image. The setup phase is bounded by $i \times j \times \lfloor r - 1/2 \rfloor + 2$ module multiplications plus $i \times j \times (r - 1)$ module additions, and the recovery phase is bounded by $O(r \log^2 r)$. Thus our scheme is efficient and practical.

5.2. Security analysis

Our (r, n) threshold scheme is based on the polynomial

$$F(x) = S' + M^{(1)} \times x + M^{(2)} \times x^2 + \dots + M^{(r-1)} \times x^{r-1} \bmod p,$$

where each component of the coefficient matrix $M^{(h)}$ ($h = 1, 2, \dots, r - 1$) is chosen from a uniform distribution over the integers in the interval $[0, p)$ with S' being the modified codebook. The most mind-boggling aspect of our scheme is that each component of the coefficient matrix is chosen randomly and that any $r - 1$ or fewer participants with infinite computing power cannot know anything more than the length of each component. That is as secure as one-time pad: an attempt at an exhaustive search will reveal that any conceivable value could be the component of the modified codebook. In other words, our scheme achieves perfect secrecy [18].

6. Conclusions

In this paper, we propose a practical (r, n) threshold scheme for sharing of secret images. Our idea is mainly inspired from the threshold scheme proposed in [2]. We assume that the secret image is compressed with the VQ technique. Nobody can decompress this image except for those who have the modified codebook. The dealer divides the modified codebook into n shadow codebooks and distributes them to n participants. If and only if r or more participants cooperate together, the modified codebook can be reconstructed. Then the secret image can be decompressed. We also extend our scheme to share multiple secret images by publishing some information, so each participant still holds the simplicity of only one shadow codebook. The compressed code does not lose any information in our scheme. Also, the difference between the modified codebook and the original local codebook is rare. In our experiments, the PSNR values of each image's decompressed images which are decompressed respectively with the modified codebook and with the original local codebook are equal. Our scheme can be applied to any image system that compresses and decompresses images using the concept of codebooks as the basis of VQ. By ignoring the time complexity of compression steps, the time complexity of our (r, n) threshold scheme is independent of the size of the shared secret image. The setup phase is bounded by $i \times j \times \lfloor r - 1/2 \rfloor + 2$ module multiplications plus $i \times j \times (r - 1)$ module additions, and the recovery phase is bounded $O(r \log^2 r)$. Furthermore, our scheme achieves perfect secrecy.

Acknowledgements

We would like to thank the anonymous referees for their useful comments.

References

- [1] G.R. Blakley, Safeguarding cryptographic keys, *Proceedings AFIPS 1979 National Computer Conference*, New York, vol. 48, pp. 313–317.
- [2] A. Shamir, How to share a secret, *Communications of the Association for Computing Machinery* 22 (11) (1979) 612–613.
- [3] D.E. Denning, *Cryptography and Data Security*, Addison-Wesley, Reading, MA, 1982.
- [4] M. Rabbani, P.W. Jones, *Digital Image Compression Techniques*, Bellingham, Washington, 1991, pp. 144–158.
- [5] R. Aravind, A. Gersho, Image compression based on vector quantization with finite memory, *Optical Engineering* 26 (7) (1987) 570–580.
- [6] C.C. Chang, R.F. Chang, W.T. Lee, C.L. Kuo, A fast search algorithm for vector quantization, *Journal of Information Science and Engineering* 12 (4) (1996) 593–602.
- [7] R.F. Chang, W.T. Chen, Image coding using variable-rate side-match finite-state vector quantization, *IEEE Transactions on Image Processing* 2 (1993) 104–108.
- [8] C.C. Chang, T.S. Chen, A new tree-structured vector quantization with closest-coupled multipath searching method, *Optical Engineering* 36 (6) (1997) 1713–1720.
- [9] T.S. Chen, C.C. Chang, Diagonal axes method (DAM): A fast search algorithm for vector quantization, *IEEE Transactions on Circuits and Systems for Video Technology* 7 (3) (1997) 555–559.
- [10] T.S. Chen, C.C. Chang, A new image coding algorithm using variable-rate side-match finite-state vector quantization, *IEEE Transactions on Image Processing* (to appear).
- [11] T.S. Chen, C.C. Chang, A new image coding algorithm using variable-rate side-match finite-state vector quantization, *IEEE Transactions on Image Processing* (to appear).
- [12] R.M. Gray, Vector quantization, *IEEE ASSP Magazine*, 1984, pp. 4–29.
- [13] K.T. Lo, J. Feng, Predictive mean search algorithms for fast VQ encoding of images, *IEEE Transactions on Consumer Electronics* 41 (2) (1995) 327–331.
- [14] N.M. Nasrabadi, R.A. King, Image coding using vector quantization: A review, *IEEE Transactions on Communications* 36 (8) (1988) 957–971.
- [15] S.A. Rizvi, N.M. Nasrabadi, An efficient Euclidean distance computation for vector quantization using a truncated look-up table, *IEEE Transactions on Circuits and System for Video Technology* 5 (1995) 370–371.
- [16] D. Knuth, *The Art of Computer Programming*, vol. 2, *Seminumerical Algorithms*, Addison-Wesley, Reading, MA, 1969.
- [17] W.H. Equitz, A new vector quantization clustering algorithm, *IEEE Transactions Acoustics, Speech, and Signal Processing* 37 (1989) 1568–1575.
- [18] C.E. Shannon, Communication theory of secrecy systems, *Bell System Technical Journal* 28 (4) (1949) 656–715.
- [19] A. Aho, J. Hopcroft, J. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.