A New Encryption Algorithm for Image Cryptosystems*

Chin-Chen Chang † Min-Shian Hwang‡
Tung-Shou Chen§

Department of Computer Science and †
Information Engineering,
National Chung Cheng University,
Chaiyi, Taiwan, R.O.C.

Department of Information Management ‡
Chaoyang University of Technology
168, Gifeng E.Rd., Wufeng,
Taichung County, Taiwan 413, R.O.C.
Email: mshwang@mail.cyut.edu.tw

Department of Information Management §
National Taichung Institute of Technology
Taichung, Taiwan, R.O.C.

April 12, 2001

*This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC89-2213-E-324-035.

‡Corresponding author.

A New Encryption Algorithm for Image Cryptosystems

Abstract

There are two major differences of the characteristics of the text-data and image data. One difference is that the size of image data is usually much larger than that of text data. The other is that plain data rarely permits loss when a compression technique is used, but image data does. In this paper, we design an efficient cryptosystem for images. Our method is based on vector quantization, which is one of the popular image compression techniques. Our method can achieve the following two goals. One goal is to design a high security image cryptosystem. The other goal is to reduce computational complexity of the encryption and decryption algorithms.
Keywords: Cryptography, Image Compression, Image Cryptosystem, Image Process, Vector Quantization

1 Introduction

A major issue for computer networks is to prevent important information from being disclosed to illegal users. For this reason, encryption techniques were introduced. Most encryption techniques have an easy implementation and are widely used in the field of information security.

During the last decade, the use of computer networks has grown spectacularly, and this growth continues unabated. Almost all networks are being installed, interconnected, and connected to the global internet. The internet is commonly seen as the first incarnation of an information superhighway. Today more and more information has been transmitted over the internet. The information is not only text, but also audio, image, and other multimedia. Images have been widely used in our daily life. However, the more extensively we use the images, the more important their security will be. For example, it is important to protect the diagrams of army emplacements, the diagrams of bank building construction, and the important data captured by military satellites. In addition, the number of computer crimes has increased recently. Image security has become an important topic in the current computer world.

Most traditional or modern cryptosystems have been designed to protect textual data (Denning, 1982). An original important and confidential plaintext is converted into ciphertext that is apparently random nonsense. Once the ciphertext has been produced, it is saved in storage or transmitted over the network. Upon reception, the ciphertext can be transformed back into the original plaintext by using a decryption algorithm.

However, images are different from text. Although we may use the traditional cryptosystems (such as RSA and DES-like cryptosystems) to encrypt images directly, it is not a good idea for two reasons. One is that the image size is almost always much greater than that of text. Therefore, the traditional cryptosystems need much time to directly encrypt the image data. The other problem is that the decrypted text must be equal to the original text. However, this requirement is not necessary for image data. Due to the characteristic of human perception, a decrypted image containing small distortion is usually acceptable.

Digital images are usually represented as two-dimensional (2D) arrays. For protecting the stored 2D data, it must be converted to one-dimensional arrays before using various traditional encryption techniques. The raster sequence image data can be encrypted into blocks by using a block cipher or a stream cipher (Denning, 1982). A product cipher can also be used to encrypt a file of image data. Since the image is very large, it is inefficient to encrypt or decrypt the picture directly. One of the best methods is to only encrypt/decrypt information that is used by image compression.

In this paper, we propose a fast encryption algorithm for image cryptosystems. Our scheme is based on vector quantization (VQ), cryptography, and other number theorems. VQ is an efficient approach to low bit-rate image compression. Its major advantage is that it has a simple hardware structure, especially the decoder. In VQ, the images are first decomposed into vectors and then sequentially encoded vector by vector. We can then use traditional cryptosystems from commercial applications. For enhancing security and reducing the computational complexity of encryption/decryption, we also apply some number theorems in our scheme.

The rest of this paper is organized as follows. Section 2 surveys some related image cryptosystems. We discuss some characteristics of an image cryptosystem, research issues of an image cryptosystem, and criteria for evaluating the security of image cryptosystems in Section 3, Section 4, and Section 5, respectively. Section 6 describes the basic operations of the VQ and an image cryptosystem. Next, we propose our new image cryptosystem in Section 7. The security analyses and computational complexity of the proposed image cryptosystem are presented and discussed in Sections 8 and 9. Finally, Section 10 concludes this paper.

2 Related Image Cryptosystems

According to the differences between image and text, recently there have been several cryptosystems(Schwartz, 1991; Bourbakis and Alexopoulos, 1992; Kuo, 1993; Chang and Liou, 1994) proposed for image security. Schwartz proposed a scrambling method to encrypt images (Schwartz, 1991). Its first step is to generate a sequence of random points on the original image. These random points are notably determined by the seed of the random number generator. The generator generates the same sequence of random points by using the same seed. Therefore, the seed is the private key of this method. Next, this method draws some graphical lines between each two consecutive points of this sequence. Moreover, its drawing pen is in the inverted mode, which changes each white pixel to black and vice versa. After drawing many inverse lines on the original image, the plainimage is thus encrypted. This method is simple and fast. However, its security degree is not high enough to protect images.

Bourbakis and Alexopoulos developed another method to encrypt images (Bourbakis and Alexopoulos, 1992). This method converts a two-dimensional image into a one-dimensional list, and employs a SCAN language (Alexopoulos, 1989) to describe the converted result. In this language, there are several SCAN letters. Each SCAN letter represents one kind of scan order. Different kinds of combinations of SCAN letters may generate different kinds of secret images. After determining the combination of SCAN letters, the scheme then generates a SCAN string. This string defines the scan order of the original image. Next, this method scans the original image in the determined order and, moreover, encrypts the SCAN string by using commercial cryptosystems. Since the illegal users cannot obtain the correct SCAN string, the original image is therefore secure. There is no image compression in this method. Therefore, the size of the image is very large, and thus it is inefficient to encrypt or decrypt the image directly.

Kuo proposed an encryption method that referred to the image distortion (Kuo, 1993). This method obtains the encrypted image by adding the phase spectra of the plainimage with those of another key image. Since the phase spectra of the original image are randomly changed, the cipherimage is unrecognizable. Thus this method is safe, but no image compression is considered.

Chang and Liou proposed an encryption method for images (Chang and Liou, 1994). This method employs two technologies to achieve the compression and encryption purposes. They are the quadtree data structure and the SCAN language, respectively. This method first compresses the original image by using a quadtree, and then encrypts the compressed data by SCAN. So, this method can compress and encrypt images concurrently. Quadtree is notably a lossless data compression technology. Therefore, this method is also lossless.

In this paper, we propose a fast encryption algorithm for image cryptosystems. The scheme is based on the VQ, number theorem, and cryptography. Our method can achieve the following two goals. One is that to design a highly secure image cryptosystem. The other is that to reduce computational complexity of the encryption and decryption algorithms.

3 Characteristics of an Image Cryptosystem

For studying an image encryption, we must first analyze the differences between implementations for image data and text data. Basically, there are some differences between image and text as follows:

1. When the ciphertext is produced, the ciphertext must be decrypted to the original plaintext in a full lossless manner. However, the cipherimage can be decrypted to the original plainimage in some lossy manner.

2. Text data are sequences of words. Therefore, the text data can be encrypted directly by using block or stream ciphers. However, digital image data are usually represented as two-dimensional arrays.

3. To encrypt text data, the minimal size can be anything from a single character to a full paper. However, to encrypt image data, the minimal size is one picture. Since the storage space of a picture is usually large (for example, a picture with 640×480 pixels needs 38 Kbytes space to store), it is inefficient to encipher the picture directly. Since the size of a picture is large, image compression techniques are always needed to compress the picture, for reducing both its storage space and its transmission time.

A good information security system is able to not only to protect confidential messages in the text form, but also in image form. In general, there are three basic characteristics in the information security field: privacy, integrity, and availability (Diffie and Hellman, 1976).

A perfect image cryptosystem is not only flexible in the security mechanism, but also has high overall performance. Thus, besides the above characteristics, the image security also requires the following characteristics.

1. The encryption system should be computationally secure. It must require an extremely long computation time to break, for example. Unauthorized users should not be able to read privileged images.

2. Encryption and decryption should be fast enough not to degrade system performance. The algorithms for encryption and decryption must be simple enough to be done by users with a personal computer.

3. The security mechanism should be as widespread as possible. It must be widely acceptable to design a cryptosystem like a commercial product.

4 Some Criteria for Evaluating the Security of Image Cryptosystems

We propose the following five attacks for evaluating the security of image cryptosystems. Each of them assumes that the cryptanalyst has the complete knowledge of the encryption algorithm used.

The first attack is called the cipherimage-only attack. In this attack, an illegal user is assumed to obtain the cipherimage from networks, but does not have the private key K. In other words, a cryptanalyst must determine the private key solely from an intercepted cipherimage, although the method of encryption, the type of the plainimage, and the format of the plainimage may be known.

The second attack is called the known-plainimage attack. The illegal users are assumed to have obtained several plainimage and cipherimage pairs in this attack. A cryptanalyst must determine or deduce the private key used to encrypt the plain-images, or an algorithm to decrypt any new cipherimage encrypted with the same private key.

The third attack is called the chosen-plainimage attack. In this attack, the illegal users are able to select the plain-images and obtain the corresponding cipher-images. This is more powerful than the known-plainimage attack, because cryptanalysts can choose some specific plain-images to encrypt, and this yields more information about the private key. When the cryptanalysts get more information about the private key, they can more easily deduce the private key used to encrypt the messages of plain-images.

The fourth attack is called the jigsaw puzzle attack. In this attack, the illegal users first divide a cipherimage into many small areas. Next, the cryptanalysts then break these areas one by one. Since each area is much smaller than the entire cipherimage, the computational load for breaking each area is thus much less than that for breaking the entire cipherimage. The jigsaw puzzle attack is therefore more efficient than other attacks.

The fifth attack is called the neighbor attack. In the attack, the illegal users are assumed to know a part of the plainimage. The changes across the boundaries of the areas are smooth in most images. Therefore, the cryptanalysts can use this attribute to speed up the selections for the boundaries of the neighboring areas. Since most images are smooth, the cryptanalysts thus can derive the neighboring pixels for the known part of the plainimage and break the whole cipherimage efficiently.

5 Research Issues of an Image Cryptosystem

According to the analyses stated in Section 3, there are three research issues on image cryptosystems as follows.

The first issue is to encrypt the image data using the same method as for text data. Images are usually represented as two-dimensional (2D) arrays. They should be converted into one-dimensional arrays before enciphering. Various encryption techniques can be used and applied on the one-dimensional lists. The raster sequence of image data can be encrypted by using block ciphers or stream ciphers. A product cipher can also be used in order to encrypt image data. Bourbakis's (Bourbakis and Alexopoulos, 1992) and Kuo's (Kuo, 1993) image cryptosystems are in this form. Since the image is large, it is inefficient to encrypt or decrypt the picture directly. Applying compression techniques to images and then encrypting the compressed images is also a way to use standard text encryption algorithms.

The second issue is to use the special features of images. The main feature of an image is that it allows a bit of distortion. Therefore, picture data can be compressed before transmitting, and be lossy decompressed with a small distortion after receiving the image compression. There are many lossy compression techniques for images. This issue is to encrypt the compressed image using the same method as for the text data. This enciphering process is shown in Figure 1. Chang and Liou's

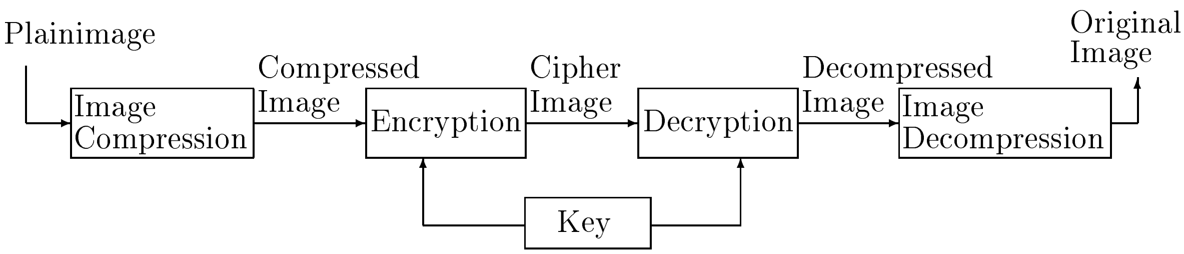

Figure 1: The block diagram of image compression and encryption.

image cryptosystem (Chang and Liou, 1994) is in this form.

In general, there are two types of compression techniques. One is a compression technique for auxiliary data such as the codebook of VQ. The other is a compression technique without any auxiliary data, such as a quadtree compression technique. When using compression techniques for image representations that have auxiliary data, there are two ways to encrypt the compressed image. One is to encrypt the compressed image data as for the above second issue. The other is to encrypt the auxiliary data. Our third issue is to only encrypt the auxiliary data by some encryption algorithms. Since the size of the auxiliary data is usually less than that of the compressed image, the time complexity for enciphering auxiliary data is less than that of the above two issues. In this paper, we will propose an image cryptosystem based on VQ. The scheme encrypts the codebook of VQ instead of the compressed image.

6 An Image Cryptosystem Based on VQ

Vector quantization (VQ) is an efficient technique for image compression (Linde et al.,1980; Gersho and Gray, 1992; Chen and Chang, 1997a; Chen and Chang, 1997b). Its basic idea is derived from Shannon's rate-distortion theory, which states that the better performance of an image compression is always achieved by coding image vectors instead of scalars (Gersho and Gray, 1992).

There are two advantages of using VQ for image compression (Linde et al., 1980; Gersho and Gray, 1992). One is that the required bit rate of VQ is small. Since VQ compresses the original image into a set of indices in the codebook, we can save lots of storage space and channel bandwidth. The other advantage is that VQ has a simple hardware structure for providing a fast decoding procedure.

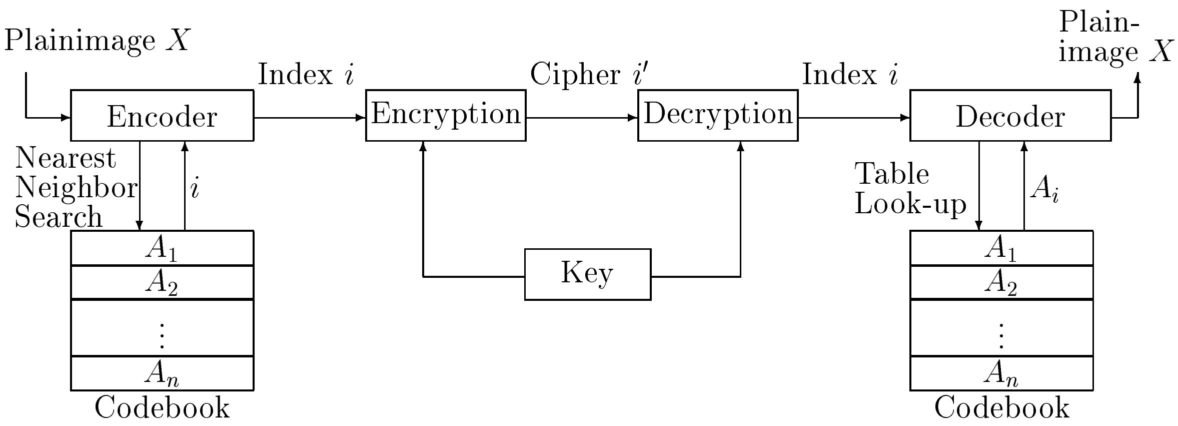

Figure 2: The block diagram for enciphering a compressed image in VQ.

Generally, we divide the VQ process into the following two phases: vector encoding and vector decoding. We present the process of the vector encoding phase in Figure 2. It decomposes the original image X into a set of vectors $\{X_1, X_2, \cdots, X_m\}$. Here m denotes the number of the vectors in X. VQ then chooses a suitable codebook to represent those vectors in X. Let A denote the codebook which consists of $\{A_1, A_2, \cdots, A_n\}$. Here A_i, $i = 1, 2, \cdots, n$, is a codeword and n is the size of the codebook. In general, the dimensionality of each codeword is 8×16 bits.

To encode each image vector X_i, VQ selects an appropriate codeword A_j for X_i such that the distortion (or distance) between A_j and X_i is the smallest. We call A_j the closest codeword of X_i. Here the distortion between A_j and X_i is measured by their squared Euclidean distance, that is,

$$d(X_i, A_j) = ||X_i - A_j||^2 = \sum_{l=1}^{u}(X_{i_l} - A_{j_l})^2, \tag{1}$$

here A_{j_l} and X_{i_l} denote the l-th component values of A_j and X_i, respectively. After the choice of A_j, VQ then uses the index j of A_j to replace X_i.

In the vector decoding phase, VQ uses the same codebook A to decode the encoded (or compressed) image. For each encoded index, the decoder selects a codeword in the codebook according to the index value. This codeword is the closest one chosen by the encoder in fact. The VQ decoder collects these closest codewords which are pointed at by the encoded indices, and then reveals the decompressed image X.

Note that VQ compresses the original image into a set of indices on the codebook. In other words, VQ transforms the original image into the combination of the codebook and a set of indices. Since our scheme is based on VQ, two data items need to be transmitted. One is the codebook. The other is a set of indices on the codebook. There are two ways to encrypt those items. One is to directly encrypt the set of indices on the codebook by using commercial cryptosystems (such as DES-like and RSA systems). The block diagram of an image cryptosystem which directly encrypts the set of indices is shown in Figure 2. The other is that to encrypt the codebook. The set of indices on the codebook is transmitted in plaintext form. The block diagram of an image cryptosystem that encrypts the codebook is shown in Figure 3. The latter is better than the former when a great deal of images with the same codebook are encrypted. In next section, we will propose an efficient scheme for the latter image cryptosystem.

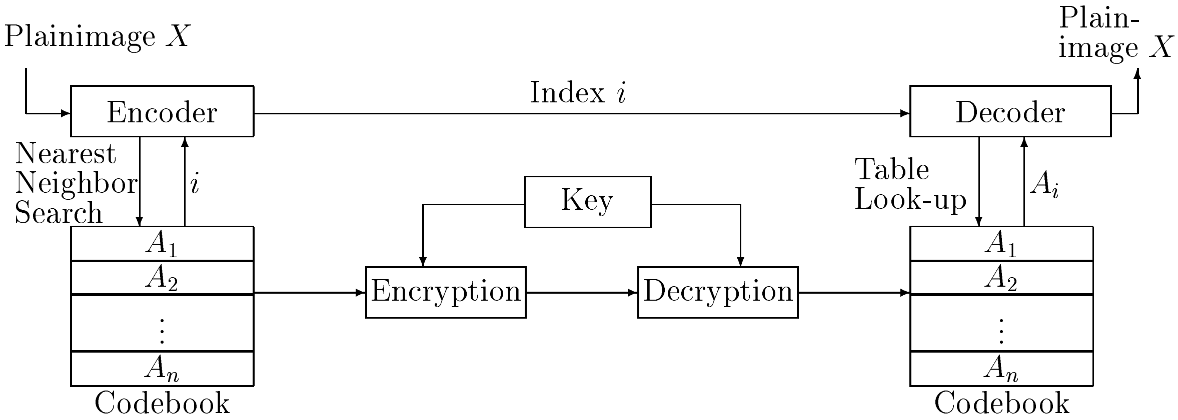

Figure 3: The block diagram for enciphering the codebook in VQ.

7 An Efficient Scheme

Shannon proposed two basic techniques for obscuring the redundancies in a plaintext message: diffusion and confusion (Denning, 1982). Confusion involves many substitutions into the relationship between the plaintext and the ciphertext. This frustrates the attempts to study the ciphertext looking for redundancies and statistical patterns. Diffusion involves many transformations (or permutations) at dissipate the redundancies of the plaintext by spreading it out over the ciphertext. A cryptanalyst looking for those redundancies will have a harder time to find them. Many ciphers, such as the DES and many public-key schemes, provide confusion and diffusion through complex enciphering transformations over large blocks of data. In addition to confusion and diffusion techniques, we also use some number theorems for our new image cryptosystem.

The new cryptosystem consists of the following three basic phases: encryption, transmission, and decryption phases. In the encryption phase, we first apply VQ to compress our original image into a set of indices. Next, we diffuse and confuse the codebook, and encrypt these parameters of the codebook by a symmetric cryptosystem (Denning, 1982; Smid and Branstad, 1988; Merkle, 1989).

In the transmission phase, our scheme sends the set of indices and the above encrypted data of the codebook by a public channel. The scheme also sends the secret key K to the receiver by a secret channel. Since K is the secret key to decrypt the cipherimage, we must send it to the legal receiver in secret. In general, there are two methods can be used to distribute the secret key K. One is by a secure channel. The other is based on the computational difficulty of computing discrete logarithms (Diffie and Hellman, 1976).

As the receiver receives the encrypted data and K, he/she may correctly decrypt the cipherimage in his/her decryption phase. The decryption procedure is notably symmetric with that of the encryption procedure.

The detailed procedures of our image cryptosystem are given as follows.

Step 1: Express the codebook in a matrix

We express codebook in the following matrix:

$$\begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_{256} \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,16} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,16} \\ \vdots & \vdots & \cdots & \vdots \\ a_{256,1} & a_{256,2} & \cdots & a_{256,16} \end{pmatrix} \tag{2}$$

Here we assume that $n = 256$ and $u = 16$. Let $A_i = (a_{i1}, a_{i2}, \cdots, a_{i,16})$ and $i = 1, 2, \cdots, 256$. The sender selects a prime P and randomly chooses a vector $E = (e_1, e_2, \cdots, e_{16})$ such that $A_i \cdot E \bmod P \leq A_{i+1} \cdot E \bmod P$ for each i, where P is less than 2^{20} and $\cdot$ is an inner product. Note that A_i is the ith codeword in codebook.

Step 2: Extend the above matrix

Randomly generate 240 vectors $(A'_1, A'_2, \cdots, A'_{240})$ such that $A'_i \cdot E \bmod P > A_{256} \cdot E \bmod P$ for all $i = 1, 2, \cdots, 240$. We give a method to generate these A'_i as follows.

Let $A_{256} \cdot E \bmod P = s$ and $A'_j = (x_{j1}, a'_{j2}, a'_{j3}, \cdots, a'_{j,16})$, where x_{j1} is a variable and a'_{ji}, $i = 2, 3, \cdots, 16$, is a random number. Since $A'_j \cdot E \bmod P > A_{256} \cdot E \bmod P$, $A'_j \cdot E \bmod P = s + Q_j$ where Q_j is a variable.

$$\begin{aligned}
& A'_j \cdot E \bmod P = s + Q_j & (3)\\
\Longrightarrow\ & (x_{j1}e_1 + a'_{j2}e_2 + a'_{j3}e_3 + \cdots + a'_{j,16}e_{16}) \bmod P = s + Q_j & (4)\\
\Longrightarrow\ & x_{j1}e_1 \bmod P = s + Q_j - (a'_{j2}e_2 + a'_{j3}e_3 + \cdots + a'_{j,16}e_{16}) \bmod P & (5)\\
\Longrightarrow\ & x_{j1}e_1 \bmod P = Q_j + \beta_j & (6)\\
\Longrightarrow\ & P|(x_{j1}e_1 - (Q_j + \beta_j)), & (7)
\end{aligned}$$

where $\beta_j = s - (a'_{j2}e_2 + a'_{j3}e_3 + \cdots + a'_{j,16}e_{16}) \bmod P$, and $a|b$ denotes a divides b. Since both e_1 and β_j are constants, we only choose x_{j1} and Q_j to satisfy Equation (7). When $A'_j, j = 1, 2, \cdots, 240$, has been generated, we put A'_j to A_{256+j}, $j = 1, 2, \cdots, 240$, and then append these A_{257}, A_{258}, $\cdots$, A_{496} to Equation (2) in Step 1.

Step 3: Generate agent codes

Our system generates an arbitrary nonsingular matrix Y in Galois Field GF(P) of size 16, say

$$\begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{16} \end{pmatrix} = \begin{pmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,16} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,16} \\ \vdots & \vdots & \cdots & \vdots \\ y_{16,1} & y_{16,2} & \cdots & y_{16,16} \end{pmatrix}. \tag{8}$$

These vectors $Y_j's$, $j = 1, 2, \cdots, 16$, constitute a basis such that corresponding to any arbitrary vector of size 16, e.g., $A_i = (a_{i1}, a_{i2}, \cdots, a_{i,16})$. Thus we find a coordinate vector $B_i = (b_{i1}, b_{i2}, \cdots, b_{i,16})$ when B is written as a linear combination of the basis elements, i.e.,

$$\begin{aligned} &(b_{i1}, b_{i2}, \cdots, b_{i,16}) \cdot (y_{j1}, y_{j2}, \cdots, y_{j,16}) = a_{ij} \\ &\text{for } i = 1, 2, \cdots, 496 \text{ and } j = 1, 2, \cdots, 16. \end{aligned} \tag{9}$$

For the above equation, the corresponding B is obtained by solving the set of linear equations.

Step 4: Generate short codes

1. randomly generate a vector $I = (i_1, i_2, \cdots, i_{16})$, where $0 \le i_p \le 15$ and $i_p \ne i_q$ for all $p \ne q$ and $1 \le p, q \le 16$.
2. put Y_{i_j} to Y_j', $j = 1, 2, \cdots, 16$.
3. randomly generate a binary string C of length 512 bits. There are 16 "0" bits and 496 "1" bits in the binary string C.
4. randomly put B_i $(i = 1, 2, \cdots, 496)$ to T_l for some l, $1 \le l \le 512$ which $C_l = 1$. And then put Y_j' to T_l, the jth position of $C_l = 0$, for all $j = 1, 2, \cdots, 16$.

Step 5: Encryption by a commercial symmetric cryptosystem

Encrypt $E = (e_1, e_2, \cdots, e_{16})$ in Step 1, and encrypt $I = (i_1, i_2, \cdots, i_{16})$ and a binary string C in Step 4 by using a symmetric cryptosystem. Let these cipher-texts be E', I', and C', respectively.

When we finish the above encryption procedure, our system then sends the set of indices in the plaintext form and the above encrypted public parameters: E', I', C', and T by public channels. As for the secret key K, we send it to the receiver by secret channels.

When the receiver receives the set of indices and the encrypted data (E', I', C') and T, he/she decrypts the cipherimage by the following decryption procedure. The decryption procedure is notably symmetric with that of the above encryption procedure.

Step 1: Apply the same symmetric cryptosystem in Step 5 of the above encryption procedures, and decrypt E', I' and C' to E, I, C, respectively.

Step 2: Retrieve T_j for some j which $C_j = 0$. Put T_j which is the lth position of $C_j = 0$ to Y_l' where $1 \le l \le 16$. Rearrange the order of Y' by I. In other words, put Y_j' to Y_{i_j} and $j = 1, 2, \cdots, 16$.

Step 3: Retrieve T_j for some j which $C_j = 1$. Put T_j which is the lth position of $C_j = 1$ to B_l' where $1 \leq l \leq 496$.

Step 4: Compute $B' \cdot Y = A'$.

Step 5: Sort A by $A_l' \cdot E \bmod P$, $l = 1, 2, \cdots, 496$. Put A_l' to A_1 which has the smallest $A_l' \cdot E \bmod P$, and reorder of A' by the following formula.

$$A_1 \cdot E \bmod P < A_2 \cdot E \bmod P < \cdots < A_{496} \cdot E \bmod P. \tag{10}$$

These vectors $A_1, A_2, \cdots, A_{256}$ are thus the original VQ-codebook. Using the codebook, the set of indices, and the VQ compression method, the original image is thus decompressed.

8 Security Analyses

The new image cryptosystem encrypts several important parameters of the plain-image by a symmetric cryptosystem, such as Data Encryption Standard (DES), Fast Data Encryption Algorithm (FEAL), and other DES-like cryptosystems (Denning, 1982). DES is the best-known and most widely used cryptosystem for commercial applications today. Since DES is a secure private-key cryptosystem, the cryptanalysts cannot break our cipherimage.

To prove the feasibility of our image cryptosystem, we analyze its security degree in this section. We analyze the security of our image cryptosystem for the following five types of attack: cipherimage-only attack, known-plainimage attack, chosen-image attack, jigsaw puzzle attack, and neighbor attack.

Under a cipherimage-only attack, the illegal users are assumed to obtain information from networks, but do not have the private key K. In generally, there are two ways to be taken. One is to directly cryptanalyze the existing symmetric cryptosystems. In this case, the cryptanalysts need to obtain E, I, and C since these parameters are encrypted by DES and the private key K. Thus the cryptanalysts cannot obtain our original image. However, suppose the illegal users try to guess private key K by brute force. Then, since a private key has 56 bits in DES, K has 2^{56} possible combinations. If the illegal users employ a 1000 MIPS computer to conjecture K, the computational load is then $\frac{2^{56}}{1000 \times 10^6 \times 60 \times 60 \times 24 \times 365} > 500$ years. This is a very long time. No image can be closed-door after five thousand years.

The other way is to directly cryptanalyze the vector T. In this case, the illegal users need to guess the combination and permutation of T correctly. The secure parameters of our cryptosystem are 256 vectors of B and 16 vectors of Y. Suppose the illegal users try to check all combinations and permutations of T. The total number of the possible combinations of T is $C_{256}^{512} \times C_{16}^{512-256}$. The total number of the possible permutations of the above right combination of T is $16! \times 256!$. The probability of the illegal users guessing the codebook is

$$\frac{1}{C_{256}^{512} \times C_{16}^{256} \times 16! \times 256!} \tag{11}$$

which is smaller than $\frac{1}{2^{272\times 8}}$. It is more difficult than guessing the private key K.

Under a known-plainimage attack, the illegal users are assumed to have obtained several plainimage and cipherimage pairs. These are denoted by (P_1, C_1), (P_2, C_2), and so on. Here C_i is a parameter which is encrypted from the plainimage P_i, and all of these pairs share a common private key K. For the attack, the purpose of the cryptanalysts is to analyze these pairs to obtain the common key K. After the acquisition of K, the illegal users can correctly decrypt the next cipherimage if the sender still encrypts his/her next original image by K. To prevent this attack, we define that our private key is disposable. If cryptanalysts get some pairs (P_1, C_1), (P_2, C_2), $\cdots$, (P_r, C_r), they still do not know the parameters E, I, and C except breaking the symmetric cryptosystems. Since the public parameters (E', I', C') are encrypted from (E, I, C), to break our cryptosystem under a known-plainimage attack is equivalent to breaking the symmetric cryptosystems under a cipherimage-only attack.

Under a chosen-plainimage attack, the illegal users are able to appoint the plain-images and obtain the corresponding cipher-images. They are denoted by $(P_1,\ C_1)$, (P_2, C_2), and so on. Here C_i is a cipherimage encrypted from the plain image P_i, and all of these pairs share a common key K. For this attack, the purpose of the cryptanalysts is to analyze these pairs to obtain the common key K. After the acquisition of K, the illegal users can correctly decrypt the next cipherimage if the sender still encrypts his/her next original image by K. Thus the cryptosystem is not secret now. This attack is more religious than the cipherimage-only and known-plainimage attacks. If cryptanalysts get some pairs (P_1, C_1), (P_2, C_2), $\cdots$, (P_r, C_r), they still do not know the parameters E, I, and C. Our image cryptosystem encrypts plain-images in two phases. The first phase is to convert plainimage to the parameters E, I, and C by a number theorem. The second phase is to encrypt E, I, and C by commercial cryptosystems. Therefore, to break the cryptosystem under a chosen-image attack is equivalent to breaking the symmetric cryptosystems under a cipherimage-only attack.

Under a jigsaw puzzle attack, the illegal users divide the cipherimage into many small areas. Next, the cryptanalysts break these areas one by one. Note that each area is much smaller than that of the entire cipherimage. Thus the computational load for breaking each area is also much less than that for breaking the entire cipherimage. Since VQ is a lossy data compression technique and our scheme is to encrypt the codebook, the jigsaw puzzle attack is no use in our image cryptosystem.

Under a neighbor attack, the illegal users are assumed to know a part of the plainimage. Since the changes across the boundaries of the areas are smooth in most images, the cryptanalysts can use this attribute to speed up the selections for the boundaries of the neighboring areas. The cryptanalysts can derive the neighboring pixels of the known part of the plainimage. The cryptanalysts thus obtain several neighbor pixel and cipherimage pairs. Next, the cryptanalysts use a known-plainimage attack to analyze these pairs to obtain the common private key K. Since VQ is a lossy data compression technique and, moreover, the boundaries of the blocks of a plainimage are always different, the cryptanalysts cannot apply the smooth attribute between neighboring boundaries to speed up their attack. Our cryptosystem is therefore immune from the neighbor attack.

9 Computational Complexity

We examined the computational complexity of each step in Section 7 as follows. The computation of Step 1 of encryption phase in Section 7 is based on that of choosing E such that $A_i \cdot E \bmod P \leq A_{i+1} \cdot E \bmod P$. Since $A_i \leq A_{i+1}$, it is easy to randomly choose this vector E. The number of operations for generating E is equal to $16RAN(8)$, where $RAN(8)$ is the time required to randomly generate an integer of length 8 bits.

The computation of Step 2 of encryption phase in Section 7 is based on that of generating 240 vectors $(A'_1, A'_2, \cdots, A'_{240})$ such that $A'_i \cdot E \bmod P > A_{256} \cdot E \bmod P$ for all $i = 1, 2, \cdots, 240$. The number of operations for generating A'_i is equal to $15RAN(8)$ plus $CAL(x_{j1})$, where $CAL(x_{j1})$ is the time required to generate x_{j1} and Q_j such that Equation (7) holds. The number of operations for generating x_{j1} is equal to $16MUL(8)$ plus $17ADD(64)$ plus $2MOD(20)$ plus one $RAN(8)$. Here, $MUL(8)$ denotes the number of multiplications of integers of length 8 bits; $ADD(64)$ denotes the number of additions of integers of length 64 bits; $MOD(20)$ denotes the number of modulus of integers of length 20 bits.

The computation of Step 3 of encryption phase in Section 7 is based on that of generating the coordinate vector B such that $B \cdot Y = A$. The number of operations for generating B is equal to $496LE(16)$, where $LE(16)$ is the time required to solve an 16 by 16 linear simultaneous equations. The computational effort for Gauss elimination is proportional to n^3 for an n by n linear simultaneous equations (Burden and Faires, 1989).

The computation of Step 4 of encryption phase in Section 7 is based on that of generating the vector I and the binary string C. The number of operations for generating I and C are equal to $16RAN(4)$ and $RAN(512)$, respectively.

The computation of Step 5 of encryption phase in Section 7 is based on that of enciphering E, I, and C by using a symmetric cryptosystem. This computation requires to encrypt/decrypt E/E' in 128 ($= 8 \times 16$) bits length, I/I' in 64 ($= 4 \times 16$) bits length, and C in 512 bits length in our scheme.

The computation of Step 5 in encryption phase is based on that of the private-key encryption algorithms. If DES is used as the symmetric cryptosystem, it partitions the plaintext into blocks of 64 bits/each. This requires

$$t = \lceil L/64 \rceil * DES(64), \tag{12}$$

where DES(64) is the time required to encipher 64 bits of text using the DES device and L is the length of the encrypted data. There are only three parameters that need to be encrypted: E, I, and C. The time required to encipher these parameters is as follows.

$$\begin{cases} t_E = \lceil 128/64 \rceil * DES(64) & = 2 * DES(64) \\ t_I = \lceil 64/64 \rceil * DES(64) & = 1 * DES(64) \\ t_C = \lceil 512/64 \rceil * DES(64) & = 8 * DES(64) \end{cases} \tag{13}$$

Thus,

$$t_{total} = 11 * DES(64).$$

Note that, when directly encrypting the whole codebook, it requires

$$t_{codebook} = \lceil (n * u)/64 \rceil * DES(64), \tag{14}$$

where n is the length of the codebook and u is the dimension of each vector in codebook. In general, u and n are equal to 8×16 bits and 256, respectively. Thus, $t_{codebook} = 512 * DES(64)$. This result is much bigger than t_{total}.

DES has been implemented in both software and hardware. Hardware implementations achieve the encryption rates of 1 gigabit per second (Eberle, 1992). A software implementation of DES on an Intel 80486/33 MHz microprocessor can perform 2.6 million bits per second; one on a Motorola 68020/16 MHz microprocessor can perform 0.22 million bits per second (Schneier, 1994). Therefore, t_{total} is equal to 0.7 us using hardware implementation and 3.2 ms using software implementation.

10 Discussions and Conclusions

We have proposed a fast encryption algorithm for image cryptosystems. Our method is based on VQ, which diffuses and confuses the codebook and encrypts these parameters of the codebook using a symmetric cryptosystem (Denning, 1982; Smid and Branstad, 1988; Merkle, 1989).

In our scheme, theoretically, many images can be sent with only one encrypted codebook designed for all these images. However, we suggest that each image is sent with its own encrypted codebook. Although, it is less efficient to send only one image each time, it is worthy to note that a confidential image can be transmitted with a higher degree of security and higher resolution.

Along with the coming of the multimedia and the network ages, more and more images are employed and transmitted among computer systems. The image security is therefore very important now. In this paper, we have proposed a fast encryption algorithm for image cryptosystems. Our method can achieve the following two goals. One is to design a highly secure image cryptosystem. The other is to reduce the time complexity for encryption and decryption.

According to the results of our security analyses, the proposed cryptosystem is safe from cipherimage-only (or brute force), known-plainimage, chosen-plainimage, jigsaw puzzle, and neighbor attacks. In addition, our cryptosystem can compress image data and, moreover, speed up the encryption processes. Therefore, we conclude that the proposed cryptosystem is very suitable for real application to image storage and transmission problems.

Furthermore, we have also discussed the characteristics of image cryptosystems, some criteria for evaluating the security of image cryptosystems, and research issues related to image cryptosystems.

Acknowledgements

The authors wish to thank many anonymous referees for their suggestions to improve this paper. Part of this research was supported by the National Science Council, Taiwan, R.O.C., under contract no. NSC89-2213-E-324-035.

References

A. Gersho and R. M. Gray, Vector quantization and signal compression, Kluwer Academic Publishers, Massachusetts, 1992.

Bruce Schneier, Applied Cryptography. Reading, New York: John Wiley & Sons, 1994.

C. Alexopoulos,SCAN, A language for 2-D sequential data accessing, Ph.D. thesis, University of Patras, Greece, 1989.

C. J. Kuo, Novel image encryption technique and its application in progressive transmission, Journal of Electronic Imageing, vol. 2, no. 4, pp. 345–351 , 1993.

C. Schwartz, A new graphical method for encryption of computer data, Cryptologia, vol. 15, no. 1, pp. 43–46, 1991.

D. E. R. Denning, Cryptography and Data Security. Reading, MA: Addison-Wesley, 1982.

H. Eberle, A high-speed DES implementation for network applications, in Advances in Cryptology, CRYPTO'92. New York: Springer-Verlag, pp. 527–545.

H. K. Chang and J. L. Liou, An image encryption scheme based on quadtree compression scheme, in Proc. International Computer Symposium, pp. 230–237, Taiwan, 1994.

M. E. Smid and D. K. Branstad, The data encryption standard: past and future, Proc. of the IEEE, vol. 76, no. 5, pp. 550–559, May 1988.

N. Bourbakis and C. Alexopoulos, Picture data encryption using SCAN patterns, Pattern Recognition, vol. 25, no. 6, pp. 567–581, 1992.

R. C. Merkle, One-way hash functions and DES, in Advances in Cryptology, CRYPTO'89. New York: Springer-Verlag, pp. 428–446.

R. L. Burden and J. D. Faires, Numerical Analysis. Reading, PWS-KENT Publishing Company (4th Edition), 1989.

T. S. Chen and C. C. Chang, A new image coding algorithm using variable-rate side-match finite-state vector quantization, IEEE Trans. on Image Processing, vol.6, no.8, pp.1185-1187, 1997.

T. S. Chen and C. C. Chang, Diagonal axes method (DAM): a fast search algorithm for vector quantization, IEEE Trans. on Circuits and Systems for Video Technology, vol.7, no.3, pp.555-559, 1997.

W. Diffie and M. E. Hellman, New directions in cryptography, IEEE Transactions on Information Theory, vol. 22, pp. 644–654, 1976.

Y. Linde, A. Buzo, and R. M. Gray, An algorithm for vector quantizer design, IEEE Transactions on Communications, vol. COM-28, no. 1, pp.84–95, 1980.

BIOGRAPHY

Chin-Chen Chang was born in Taichung, Taiwan, the Republic of China, on November 12, 1954. He received his B.S. degree in Applied Mathematics in 1977 and his M.S. degree in Computer and Decision Sciences in 1979 from National Tsing

Hua University, Hsinchu, Taiwan. He received his Ph.d. in Computer Engineering in 1982 from National Chiao Tung University, Hsinchu, Taiwan. From 1983 to 1989, he was among the faculty of the Institute of Applied Mathematics, National Chung Hsing University, Taichung, Taiwan. Since August 1989, he has worked as a professor of the Institute of Computer Science and Information Engineering at National Chung Cheng University, Chiayi, Taiwan. Dr. Chang is a Fellow of IEEE and a member of the Chinese Language Computer Society, the Chinese Institute of Engineers of the Republic of China, and the Phi Tau Phi Society of the Republic of China. His research interests include computer cryptography, data engineering, and image compression.

Min-Shiang Hwang received the B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, Republic of China (ROC), in 1980; the M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988; and the Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at National Cheng Kung University, Taiwan, from 1984-1986. Dr. Hwang passed the National Higher Examination in field "Electronic Engineer" in 1988. He also passed the National Telecommunication Special Examination in field "Information Engineering", qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a project leader for research in computer security at TL in July 1990. He obtained the 1997, 1998, and 1999 Distinguished Research Awards of the National Science Council of the Republic of China. He is currently a professor and chairman of the Department of Information Management, Chaoyang University of Technology, Taiwan, ROC. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include database and data security, cryptography, image compression, and mobile communications.

Tung-Shou Chen was born in Taichung, Taiwan, Republic of China, on October 14, 1964. He received the B.S. and Ph.D. degrees from National Chiao Tung University in 1986 and 1992, respectively, both in Computer Science and Information Engineering. He served at the computer center, Chinese Army Infantry School, Taiwan, from 1992 to 1994. During the academic years 1994-97, he was on the faculty of the Department of Information Management at National Chin-Yi Institute of Technology. From August 1998 to July 2000, he was a professor of the Department of Computer Science and Information Management at Providence University. Since August 2000, he has been a professor of the Department of Information Management at National Taichung Institute Technology, Taichung, Taiwan. His current research interests include data structures, image cryptosystems, and image compression.