IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 4, No 2, July 2010
ISSN (Online): 1694-0784
ISSN (Print): 1694-0814

A New Public-Key Encryption Scheme Based on Non-Expansion Visual Cryptography and Boolean Operation

Abdullah M. Jaafar and Azman Samsudin

School of Computer Sciences, Universiti Sains Malaysia
11800 Penang, Malaysia

Abstract

Currently, most of the existing public-key encryption schemes are based on complex algorithms with heavy computations. In 1994, Naor and Shamir proposed a simple cryptography method for digital images called visual cryptography. Existing visual cryptography primitives can be considered as a special kind of secret-key cryptography that does not require heavy computations for encrypting and decrypting an image. In this paper, we propose a new public-key encryption scheme for image based on non-expansion visual cryptography and Boolean operation. The proposed scheme uses only Boolean operations and therefore requires comparatively lower computations.

Keywords: *Public-key encryption, Visual cryptography, Pixel expansion, Boolean operation.*

1. Introduction

Information security in the present era is becoming very important in communication and data storage. Data transferred from one party to another over an insecure channel (e.g., Internet) can be protected by cryptography. The encrypting technologies of traditional and modern cryptography are usually used to avoid the message from being disclosed [1-5]. Public-key cryptography usually uses complex mathematical computations to scramble the message [1, 2, 6].

In 1976, Diffie and Hellman [7] introduced the first concept of public-key (asymmetric-key) cryptography to solve the key exchange problem. Public-key cryptography is one of the greatest contributions in the history of cryptography. Nowadays, public-key cryptography is practically utilized in everyday life to attain privacy, authenticity, integrity, and non-repudiation [6, 8, 9]. One of the main branches and applications of the public-key cryptography is a public-key encryption scheme which allows two parties to communicate securely over an insecure channel without having prior knowledge of each other to establish a shared secret key. Unlike secret-key (symmetric-key) encryption scheme, public-key encryption scheme does not use the same key to encrypt and decrypt a message. Instead, each one of the two parties has two different keys but related mathematically, the public key known to everyone and the private key known only to receiver of the message. There are some popular public-key encryption algorithms, for example, RSA, ElGamal, and ECC. The security of the most public-key encryption algorithms is based on discrete logarithms in finite groups or integer factorization [6, 9, 10]. However, with all benefits and advantages of public-key encryption schemes, these schemes require a great deal of complex computations to generate the keys and to encrypt and decrypt confidential information; hence computing devices (computers) are fundamental for generation of the keys and for both encryption and decryption [1]. Conducting such computations without the assistance of a computing device (e.g., computer) is a difficult task, if not impossible [11]. Under this situation, we propose a new public-key encryption scheme with easy encryption/decryption algorithms and a comparatively low computation complexity [12].

In 1994, Naor and Shamir [13] suggested an emerging cryptography method, namely, visual cryptography (VC), which is very easy to use and perfectly safe. The encryption process is performed by simple and low computational device, whereas the decryption process is performed directly by human visual system without any complex computations. VC can be used where computers are scarce or access to them is not possible [2, 13-18]. VC divides a secret image into *n* transparent shares and it uses the human visual system to recover the secret image by superimposing and aligning carefully all or some of *n* transparent shares according to visual cryptography scheme used [2].

Because Boolean AND operation is simple, quick, and very adaptive to be implemented to an image cryptography system, the proposed public-key encryption scheme in this paper is also based on this operation.

In this paper, a novel public-key encryption scheme is proposed, which is based on non-expansion visual cryptography and Boolean operation, to overcome the problem of complex computations as in most of the existing public-key encryption schemes. The proposed scheme begins to establish a shared visual secret key between two communicating parties such as Alice and

Bob. After that, this shared visual secret key will be used in our proposed scheme during encryption and decryption processes.

The remaining of this paper is as follows. Section 2 gives a brief background on the conventional public-key encryption scheme and the visual cryptography. Section 3 describes and explains our proposed method. We discuss the security analysis and computational complexity of our proposed method in Section 4. Section 5 gives the experimental results. Finally, the conclusion is given in Section 6.

2. Background

2.1 Conventional public-key encryption scheme

In the process of encryption, the sender encrypts his or her confidential information in such a way that only the intended recipient can decrypt the confidential information. Using public-key encryption mechanism all communications between two parties over an unprotected channel involve only public keys, and without need for exchanging any private (secret) key [3]. Public-key encryption depends on a different two keys but mathematically linked, the first key is the public key which is put in a public, used for encryption; and the second key is the private (secret) key which is kept secret, used for decryption. In addition, it is computationally difficult to derive the private key from the public key [6, 10, 19, 20]. Fig. 1 shows the concept of public-key encryption technique. For this system, suppose that the receiver, Bob, has private key and public key are (PR_R) and (PU_R) respectively. Receiver's public key (PU_R) is publicly known, used for encryption; and receiver's private key (PR_R) is kept secret, used for decryption. Suppose that the sender, Alice, wants to send an original secret message (SM) to the receiver (Bob), Alice (the sender) will encrypt her secret message (SM) using Bob's public key (PU_R) to get the encrypted secret message, which is known as cipher message (C), and sends it to Bob (the receiver). The receiver (Bob) can decrypt the cipher message (C) by using only his private key (PR_R).

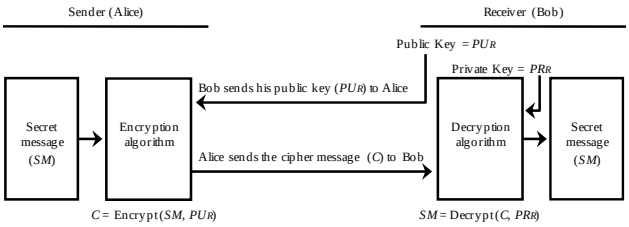

Fig. 1 The concept of public-key encryption technique

The RSA public-key encryption algorithm [21] is the first practical encryption scheme based on the concept of the public-key cryptography. There are many public-key encryption algorithms published after the RSA public-key encryption, such as ElGamal public-key encryption [22] and elliptic curve public-key encryption [23] and others.

2.2 Visual cryptography

Visual Cryptography is a special kind of cryptography which encrypts visual information (i.e., pictures, printed text, handwritten notes, etc.) into n transparent images (shares) so that humans can perform the decryption visually without the assistance of computers [13, 24]. Each one of n transparent images is an indistinguishable from random noise, thus they can be transmitted or distributed over an unprotected communication channel (i.e., Internet). In other words, because the shares appear as random binary patterns, the attackers cannot sense any hints about a secret image from individual shares [2]. The secret information can be decrypted from the shares directly by the human visual system when all or any majority of the shares are stacked together so that the subpixels are carefully aligned. On the other hand, any minority number of stacked shares or every share individually cannot leak any hint about the secret information, even if computers are available [13, 25, 26]. The basic model of visual cryptography, addressed by Naor and Shamir, divides every pixel in an original image into 2×2 black and white subpixels in the two shares on the basis of the rules in Table 1. As in Table 1, a white pixel is encrypted into two identical blocks in the two shares, and a black pixel is encrypted into two complementary blocks in the two shares. Each block is 2×2 black and white subpixels [13, 26, 27]. Take Fig. 2 for example. It shows the result obtained according to Table 1. The original secret image (a) is encrypted into two transparent shares (b) and (c). We can get the recovered secret image (d) when these two transparent shares (b) and (c) are superimposed together and carefully aligned.

Table 1: Naor and Shamir's (2, 2) visual cryptography scheme of black and white pixels (adapted from [2, 11, 13, 27-30])

Pixel of the secret image	***White pixel***		***Black pixel***	
Share 1				
Share 2				
Stacked results (Share 1+ Share 2)				

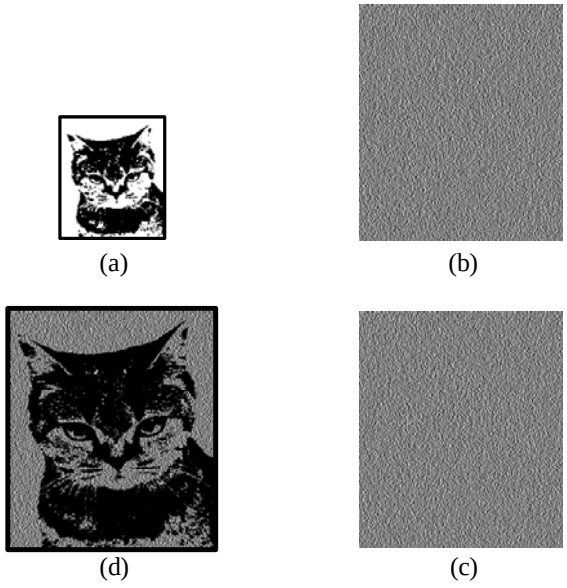

Fig. 2 The concept of Naor and Shamir's (2, 2) visual cryptography scheme with four subpixels: (a) The original secret image, (b) The first share, (c) The second share, (d) The recovered image by superimposing (b) and (c)

Most studies of visual cryptography schemes are based on the technique of pixel expansion; therefore, the resultant shares of encrypted secret image by this technique are expanded several times of the original size thereby causing many problems such as image distortion, use of more memory space, and difficulty in carrying shares. To overcome the above-mentioned problems, Ito et al. [18] and Yang [31] proposed non-expansion visual cryptography or so-called probabilistic visual cryptography (ProbVC) model for black and white images, namely, they merged the conventional visual cryptography with the concept of the probability and without pixel expansion. In their models the sizes of the original image, shares (shadow images), and the recovered image are the same. Each pixel in the original secret image is represented as a black or white pixel in the shares and the original secret image can be distinguished by superimposing these shares together. ProbVC models directly use the ready-made two $n \times n$ Boolean basis matrices S^0 and S^1 to generate the shares. To encrypt a pixel from the secret image in ProbVC models, one randomly selects a column in S^0 or S^1 according to the color of the pixel (white or black), and assigns i-th row of the selected column to i-th share (corresponding share). Ito et al. [18] defined a new parameter $\beta = |p_0 - p_1|$ to represent the contrast of the recovered image, where p_0 and p_1 are the probabilities with which a black pixel on the recovered image is created from a white and black pixel on the secret image, respectively. Table 2 shows Ito et al.'s (2, 2) ProbVC scheme that a pixel on a black and white secret image is mapped into a corresponding pixel in each of the two shares (without pixel expansion). The secret image is recovered by stacking and aligning carefully the pixels of the two shares, where every pixel in share 1 is superimposed on the corresponding pixel in share 2; this is performed through the OR operation on the two transparent shares [32]. Fig 3 shows the result obtained according to Table 2. In the following section, ProbVC models will be used to construct our proposed public-key encryption scheme.

Table 2: Ito et al.'s (2, 2) ProbVC scheme of black and white pixels (adapted from [18, 32])

Pixel of the secret image	*Share 1*	*Share 2*	*Recovered results*	*Probability*
□	□	□	□	0.5
	■	■	■	0.5
■	□	■	■	0.5
	■	□	■	0.5

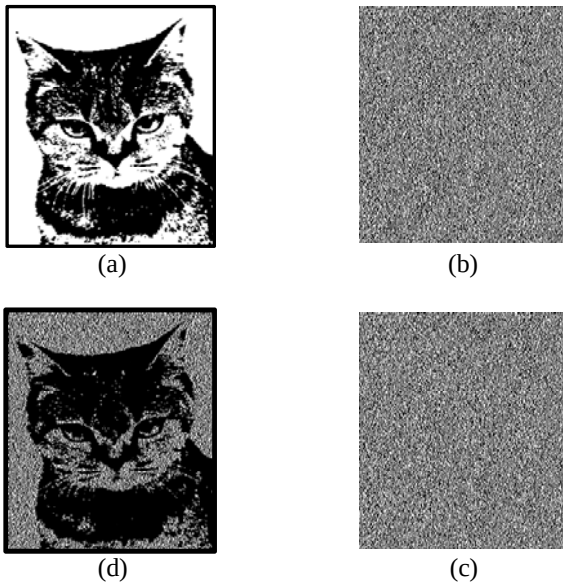

Fig. 3 An example of (2, 2) ProbVC scheme without pixel expansion: (a) The original secret image, (b) The first share, (c) The second share, (d) The recovered image by superimposing (b) and (c)

3. The proposed method

In this section, we propose a new approach to public-key encryption scheme based on a non-expansion visual cryptography and Boolean AND operation. Instead of generating and computing large and long random integer values as in the most of the existing public-key encryption schemes, our scheme generates shadow images (shares) and manipulates them by using simple Boolean OR and

AND operations. The Boolean OR operation can be performed by the human visual system on the transparent shares (i.e., superimposing the shares) as shown in [18, 31], whereas the Boolean AND operation can be performed by any simple low-end computer. Table 3 gives the truth tables of the OR (∨) and the AND (∧) operations for binary inputs.

Table 3: The truth tables of OR and AND Boolean operations for binary inputs

∨	$b=0$	$b=1$
$a=0$	0	1
$a=1$	1	1

∧	$b=0$	$b=1$
$a=0$	0	0
$a=1$	0	1

The OR (∨) and the AND (∧) of two $N_{Row} \times N_{Column}$ binary matrices can be described by the following formulas:

$\forall\, a_{ij} \in A, b_{ij} \in B,$
$C = A \vee B = [a_{ij} \vee b_{ij}], i = 1, \dots, N_{Row};\ j = 1, \dots, N_{Column}.$
$D = A \wedge B = [a_{ij} \wedge b_{ij}], i = 1, \dots, N_{Row};\ j = 1, \dots, N_{Column}.$

The expression $C = A \vee B$ means that the ij-th element c_{ij} of matrix C is equal to $(a_{ij} \vee b_{ij})$, where a_{ij} and b_{ij} are the ij-th elements of matrix A and matrix B, respectively. Similarly, the expression $D = A \wedge B$ means that the ij-th element d_{ij} of matrix D is equal to $(a_{ij} \wedge b_{ij})$, where a_{ij} and b_{ij} are the ij-th elements of matrix A and matrix B, respectively.
The proposed scheme also involves the binary inner product of two $N_{Row} \times N_{Column}$ binary matrices, denoted as $A \odot B$ which is computed by using simple Boolean OR (∨) and AND (∧) operations as follows:

$$E = A \odot B = \left[\bigvee_{k=1}^{N} (a_{ik} \wedge b_{kj}) \right],$$
$$\text{where } i = 1, \dots, N_{Row};\ j = 1, \dots, N_{Column}.$$

The expression $E = A \odot B$ means that the ij-th element e_{ij} of matrix E is equal to $(a_{i1} \wedge b_{1j}) \vee (a_{i2} \wedge b_{2j}) \vee \cdots \vee (a_{iN} \wedge b_{Nj})$, where a and b are the binary elements of matrix A and matrix B, respectively.
The proposed scheme consists of the notations used, the shared visual secret key generation phase, the encryption phase, the decryption phase, and brief comparison with the conventional public-key encryption schemes. The proposed scheme will begin to establish a shared visual secret key between two communicating parties such as Alice and Bob. Next, we will use this shared visual secret key at encryption and decryption phases.

3.1 The notations

Table 4 summarizes notations used in this paper.

Table 4: The notations

Notation	***Description***
G	An integer number with $G \geq 2$
PU	A visual public share (common shadow image)
IM_A	A black and white secret image selected by the first party (Alice) for generating G visual private keys
IM_B	A black and white secret image selected by the second party (Bob) for generating G visual private keys
PRA_i	First party's visual private keys, where $i = 1, \dots, G+2$
PRB_i	Second party's visual private keys, where $i = 1, \dots, G+2$
PRB_{G+1}^{-1}	an inverse matrix (key) of second party's visual private key PRB_{G+1}
PUA_1	First party's first visual public key
PUA_2	First party's second visual public key
PUB_1	Second party's first visual public key
PUB_2	Second party's second visual public key
CA_i	First party's first intermediate shares in the first stage of the construction procedure, where $i = 1, \dots, G$
CB_i	Second party's first intermediate shares in the first stage of the construction procedure, where $i = 1, \dots, G$
DA_j	First party's second intermediate shares in the first stage of the construction procedure, where $j = 1, \dots, G$
DB_j	Second party's second intermediate shares in the first stage of the construction procedure, where $j = 1, \dots, G$
EA_i	First party's first intermediate shares in the second stage of the construction procedure, where $i = 1, \dots, G$
EB_i	Second party's first intermediate shares in the second stage of the construction procedure, where $i = 1, \dots, G$
FA_j	First party's second intermediate shares in the second stage of the construction procedure, where $j = 1, \dots, G$
FB_j	Second party's second intermediate shares in the second stage of the construction procedure, where $j = 1, \dots, G$
H_A	First party's third intermediate share in the second stage of the construction procedure
H_B	Second party's third intermediate share in the second stage of the construction procedure
$SVSK_A$	First party's shared visual secret key
$SVSK_B$	Second party's shared visual secret key
CI	Ciphered image

3.2 Shared visual secret key generation phase

This phase consists of the initialization and construction procedure as follows.

A) Initialization

- The first party (Alice) and the second party (Bob) agree on a public integer G greater than or equal to 2 and a visual public key PU in the form of $n \times n$ pixels, where PU is a non-invertible and a non-identity matrix.
- The first party (Alice) chooses secretly a black and white image IM_A with size $n \times n$ pixels, and uses (k, k) ProbVC scheme as shown in [18, 31] (here k is equal to G) to encrypt IM_A into G visual private keys, $PRA_1, \dots, PRA_G$, where each one of them is in the form of $n \times n$ pixels.
- The first party (Alice) builds the visual private key PRA_{G+1} in the form of $n \times n$ pixels, where PRA_{G+1} is not an identity matrix.
- The second party (Bob) chooses secretly a black and white image IM_B with size $n \times n$ pixels, and uses (k, k) ProbVC scheme as shown in [18, 31] (here k is equal to G) to encrypt IM_B into G visual private keys, $PRB_1, \dots, PRB_G$, where each one of them is in the form of $n \times n$ pixels.
- The second party (Bob) builds the visual private key PRB_{G+1} and its inverse PRB_{G+1}^{-1}, where each one of them in the form of $n \times n$ pixels and PRB_{G+1} is not an identity matrix.

B) Construction procedure: It consists of the following two stages:

(a) First stage of the construction procedure: First party (Alice) produces her first visual public key PUA_1 and the second party (Bob) produces his two visual public keys PUB_1 and PUB_2. Below are the details of the first stage of the construction procedure, which is performed simultaneously by Alice and Bob.

First stage (generated by first party, Alice):

Step 1: Generate the first visual public key PUA_1 as follows.

First, construct the first intermediate shares $CA_1, \dots, CA_G$ of G as follows:

$$CA_i = PRA_i \vee PU \qquad (i = 1, \dots, G) \qquad (1)$$

Second, construct the second intermediate shares $DA_1, \dots, DA_G$ of G as follows:

$$DA_j = PRA_{G+1} \vee CA_j \quad (j = 1, \dots, G) \qquad (2)$$

Third, superimpose the second intermediate shares $DA_1, \dots, DA_G$ of G for getting the first visual public key PUA_1 as follows:

$$PUA_1 = DA_1 \vee DA_2 \vee \cdots \vee DA_G \qquad (3)$$

Step 2: Send PUA_1 to the second party (Bob).

First stage (generated by second party, Bob):

Step 1: Generate the first visual public key PUB_1 as follows.

First, construct the first intermediate shares $CB_1, \dots, CB_G$ of G as follows:

$$CB_i = PRB_i \vee PU \qquad (i = 1, \dots, G) \qquad (4)$$

Second, construct the second intermediate shares $DB_1, \dots, DB_G$ of G as follows:

$$DB_j = PRB_{G+1} \vee CB_j \quad (j = 1, \dots, G) \qquad (5)$$

Third, superimpose the second intermediate shares $DB_1, \dots, DB_G$ of G for getting the first visual public key PUB_1 as follows:

$$PUB_1 = DB_1 \vee DB_2 \vee \cdots \vee DB_G \qquad (6)$$

Step 2: Compute the second visual public key PUB_2 as follows.

$$PUB_2 = PRB_{G+1} \odot PU \qquad (7)$$

Note that PUB_2 is the Boolean product of Bob's visual private matrix PRB_{G+1} and the visual public matrix PU.

Step 3: Send PUB_1 and PUB_2 to the first party (Alice).

(b) Second stage of the construction procedure: The first party (Alice) generates the shared visual secret key $SVSK_A$. The second party (Bob) generates the second visual public key PUB_2 and then generates the shared visual secret key $SVSK_B$, (note that $SVSK_A$=$SVSK_B$). Below are the details of the second stage of the construction procedure for the first party and the second party.

Second stage (generated by first party, Alice):

Step 3: Receive the second party's (Bob's) visual public keys PUB_1 and PUB_2.

Step 4: Compute the second visual public key PUA_2 as follows.

$$PUA_2 = PUB_2 \odot PRA_{G+1} \quad (8)$$

Note that PUA_2 is the Boolean product of Bob's second visual public matrix PUB_2 and Alice's visual private matrix PRA_{G+1}.

Step 5: Send PUA_2 to the second party (Bob).

Step 6: Generate the shared visual secret key $SVSK_A$ as follows.

First, construct the first intermediate shares $EA_1, \ldots, EA_G$ of G as follows:

$$EA_i = PRA_i \vee PUB_1 \quad (i = 1, \ldots, G) \quad (9)$$

Second, construct the second intermediate shares $FA_1, \ldots, FA_G$ of G as follows:

$$FA_j = PRA_{G+1} \vee EA_j \quad (j = 1, \ldots, G) \quad (10)$$

Third, superimpose the second intermediate shares $FA_1, \ldots, FA_G$ of G for getting the third intermediate share H_A as follows:

$$H_A = FA_1 \vee FA_2 \vee \cdots \vee FA_G \quad (11)$$

Fourth, compute the visual private key PRA_{G+2} as follows:

$$PRA_{G+2} = PU \odot PRA_{G+1} \quad (12)$$

Note that PRA_{G+2} is the Boolean product of the visual public matrix PU and Alice's visual private matrix PRA_{G+1}.

Fifth, superimpose the third intermediate share H_A and the visual private key PRA_{G+2} for getting the shared visual secret key $SVSK_A$:

$$SVSK_A = H_A \vee PRA_{G+2} \quad (13)$$

Second stage (generated by second party, Bob):

Step 4: Receive the first party's (Alice's) visual public keys PUA_1 and PUA_2.

Step 5: Generate the shared visual secret key $SVSK_B$ as follows.
First, construct the first intermediate shares $EB_1, \ldots, EB_G$ of G as follows:

$$EB_i = PRB_i \vee PUA_1 \quad (i = 1, \ldots, G) \quad (14)$$

Second, construct the second intermediate shares $FB_1, \ldots, FB_G$ of G as follows:

$$FB_j = PRB_{G+1} \vee EB_j \quad (j = 1, \ldots, G) \quad (15)$$

Third, superimpose the second intermediate shares $FB_1, \ldots, FB_G$ of G for getting the third visual intermediate share H_B as follows:

$$H_B = FB_1 \vee FB_2 \vee \cdots \vee FB_{G-1} \vee FB_G \quad (16)$$

Fourth, compute the visual private key PRB_{G+2} as follows:

$$PRB_{G+2} = PRB_{G+1}^{-1} \odot PUA_2 \quad (17)$$

Note that PRB_{G+2} is the Boolean product of the inverse of the visual private key PRB_{G+1}, namely, PRB_{G+1}^{-1}, and Alice's second visual public key PUA_2.

Fifth, superimpose the third intermediate share H_B and the visual private key PRB_{G+2} for getting the shared visual secret key $SVSK_B$:

$$SVSK_B = H_B \vee PRB_{G+2} \quad (18)$$

3.3 Encryption phase

Suppose that one of the two parties has a black and white secret image and he/she wants to send it to another party. Because the first party's shared visual secret key $SVSK_A$ is equal to the second party's shared visual secret key $SVSK_B$ (i.e., $SVSK = SVSK_A = SVSK_B$), the shared visual secret key $SVSK$ can serve as an encryption key in the sender party and as a decryption key in the receiver party. Suppose that the first party (such as Alice) has a black and white secret image SI with size $n \times n$ pixels and she wants to send it to the second party (such as Bob). The sender (Alice) must do the following steps:

Step 1: Use the (2, 2) ProbVC scheme as shown in Subsection 2.2, for encoding (encryption) the secret image SI into two shares (shadow images), where each share is in the form $n \times n$ pixels. The first share should be equal to the shared visual secret key $SVSK_A$ (which serves as an encryption key) which has been established previously in the shared visual secret key generation phase and the

second share will be the ciphered image *CI* which computes from the original secret image *SI* and the shared visual secret key $SVSK_A$ as shown in the following formula.

$$CI = ENCRYPT_{SVSK_A}^{(2,2)\text{ProbVC}}(SI) \quad (19)$$

Step 2: Send the ciphered image *CI* to the receiver, Bob.

3.3 Decryption phase

The receiver (Bob) must do the following steps:

Step 1: Receive the ciphered image *CI* from the sender, Alice.

Step 2: Superimpose the shared visual secret key $SVSK_B$ (which serves as a decryption key) and the ciphered image *CI*, align them carefully, for recovering the secret image *SI* as follows.

$$SI = SVSK_B \vee CI \quad (20)$$

Note that the recovered secret image *SI* is in the form of $n \times n$ pixels.

Fig. 4 gives the basic idea of the proposed public-key encryption scheme.

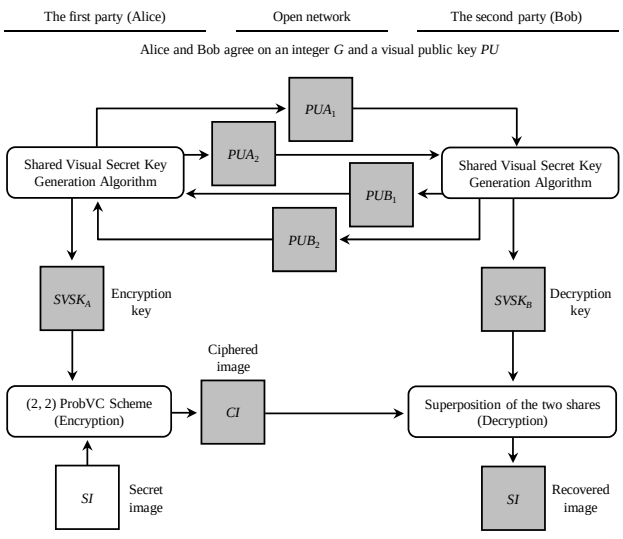

Fig. 4 The basic idea of the proposed public-key encryption scheme

3.4 Comparison with the conventional public-key encryption schemes

Our proposed scheme has some advantages and benefits compared with conventional public-key encryption schemes. Table 5 gives a summary of the comparison.

Table 5: Brief comparison between conventional public-key encryption schemes and the proposed scheme

Algorithm	***Requirement***	***Secret information***	***Complex computation***
RSA	Modular exponentiation arithmetic	Numbers in finite fields	High
ElGamal			
The proposed scheme	Simple Boolean arithmetic	Shadow images	Low

4. Security analysis and computational complexity

4.1 Security analysis

Because the ciphered image and the visual public keys (i.e., *CI*, PUA_1, PUA_2, PUB_1, PUB_2) are open to public, the attackers may try to generate the shared visual secret key from *CI*, PUA_1, PUA_2, PUB_1, and PUB_2 in order to decrypt the ciphered image *CI*, that is, to recover the original secret image. The security of the proposed encryption algorithm is based on the security of the shared visual secret key ($SVSK = SVSK_A = SVSK_B$) which depends on the solving of the problem of Boolean algebra arithmetic, where it is not possible to obtain the shared visual secret key from the public information (i.e., *CI*, PUA_1, PUA_2, PUB_1, and PUB_2). This is especially true when using a large integer number *G*, which leads to the use of huge probabilities and a large number of Boolean operations. In addition, it is practically impossible to find the inverse of PUA_2 or PUB_2 from the visual public key *PU* because *PU* is a non-invertible matrix as we conditioned above. Therefore, an attacker will face a hard time to obtain the shared visual secret key and to discover the secret image *SI*. In encryption phase, the sender (Alice) can encrypt the secret image *SI* by the (2, 2) ProbVC scheme into two shares (shadow images). The first share must be equal to the shared visual secret key *SVSK* (*SVSK* has already been established between the two parties and here it serves as an encryption key) while the second share is the ciphered image *CI*. The sender sends *CI* to the receiver. In decryption phase, the receiver can recover the secret image *SI* by stacking the shared visual secret key *SVSK* (here *SVSK* serves as a decryption key) and the ciphered image *CI*, but *CI* alone cannot disclose any information about the original secret image. In addition, if the ciphered image *CI* is changed and forged by an attacker, the stacked image will be unclear and the secret is still unidentified. Therefore, our proposed scheme is secure.

4.2 Computational complexity

Image encryption includes two steps: first, obtain the shared visual secret key, and then create the ciphered image *CI*. The shared visual secret key includes constructing 2*G* of the intermediate shares, $D_1,\ldots, D_G$ and $F_1,\ldots, F_G$, and constructing two multiplications of two binary matrices. The time complexity of constructing 2*G* of the intermediate shares is $O(n^2G) + O(n^2G) = O(n^2G)$ and the time complexity of constructing two multiplications of two binary matrices is $O(n^3) + O(n^3) = O(n^3)$ if neglecting the constant and the multiplication of two binary matrices is carried out naively. Therefore, the total time complexity of the shared visual secret key is either $O(n^3) + O(n^2G) = O(n^3)$ when $G \leq n$, or $O(n^3) + O(n^2G) = O(n^2G)$ when $G \geq n$, excluding the time needed to generate 2*G*+2 distinct random shares, where the size of the share is equal to $n \times n$ pixels. Generating the shared visual secret key requires 6*G*-1 of OR operations (here OR operations mean superimposing the shares), and at most two multiplications of two binary matrices (keys) in the sender party (here multiplication means performing the OR and the AND operations of two binary matrices as shown previously in Section 3). The time complexity of the ciphered image *CI* is $O(n^2)$. Therefore, the total time complexity for image encryption is either $O(n^3) + O(n^2) = O(n^3)$ when $G \leq n$, or $O(n^2G) + O(n^2) = O(n^2G)$ when $G \geq n$.

Image decryption includes two steps: first, obtain the shared visual secret key and then superimpose the shared visual secret key and the ciphered image *CI* for reconstructing the secret image. The time complexity for reconstructing the secret image is equal to the time complexity of the shared visual secret key which is, as we already mentioned in this Subsection, either $O(n^3)$ when $G \leq n$ or $O(n^2G)$ when $G \geq n$, excluding the time needed to generate 2*G*+2 distinct random shares. Reconstructing the image requires 6*G* of OR operations (here OR operations mean superimposing the shares), and at most two multiplications of two binary matrices (keys) in the receiver party (here multiplication means performing the OR and the AND operations of two binary matrices as shown previously in Section 3).

Suppose that an attacker wants to recover our public-key encryption scheme, two steps are required:

- Suppose that there is no a computer; Table 6 shows how much time is needed to reconstruct the visual public-key encryption scheme, performed by a human who does not use any computational devices (i.e., done manually). We will assume using some different sizes of shares and *G* is equal to 2, 64, and 128. Also, we assume that a person performs one operation per minute. From Table 6, first, we can see that the time required increases with the increasing size of share while using the value of *G* less than or equal to *n*. Second, we can also see that the time required increases with the increasing value of *G* (here *G* is greater than or equal to *n*) while using the same size of share. For example, from same table, the time required to reconstruct our scheme is more than half year when the share size is 64×64 pixels and *G* is less than or equal to $n = 64$. The time required to reconstruct our scheme is more than one year when the share size is 64×64 pixels and $G = 128$, and so on. Therefore, we can say that our scheme is secure when performing the computation manually and using a proper size of the share with a proper value of *G*.

Table 6: The Time spent to reconstruct our scheme manually

Description		Share size (pixels)	*G*	Number of Boolean operation	Time required*
1	Shared visual secret key	2×2	2	2^3	8 min
	Encryption		64	2^8	4.26 h
	Decryption		128	2^9	8.59 h
2	Shared visual secret key	4×4	2	2^6	1.06 h
	Encryption		64	2^{10}	17.06 h
	Decryption		128	2^{11}	1.42 d
3	Shared visual secret key	8×8	2	2^9	8.59 h
	Encryption		64	2^{12}	2.84 d
	Decryption		128	2^{13}	5.68 d
4	Shared visual secret key	16×16	2	2^{12}	2.84 d
	Encryption		64	2^{14}	11.37 d
	Decryption		128	2^{15}	22.75 d
5	Shared visual secret key	32×32	2	2^{15}	22.75 d
	Encryption		64	2^{16}	1.51 mth
	Decryption		128	2^{17}	3.03 mth
6	Shared visual secret key	64×64	2	2^{18}	6.06 mth
	Encryption		64	2^{18}	6.06 mth
	Decryption		128	2^{19}	1.01 yr
7	Shared visual secret key	128×128	2	2^{21}	4.04 yr
	Encryption		64	2^{21}	4.04 yr
	Decryption		128	2^{21}	4.04 yr

* min = minutes, h = hours, d = days, mth = months, yr = years

- Suppose that there is a computer; Tables 7 shows how much time is spent to recover the shared visual secret key when performed by a human who uses a computational device (i.e., a computer). We will assume a computer that executes one billion instructions per second. We will also assume using shadow images (shares) of large sizes and *G* is equal to 256, 4096, and 32768. From Table 7, first, we can see that the time required increases with increasing the size of share while using the value of *G* is less than or equal to *n*. Second, from the same table, we can also see that the time required increases with increasing the value of *G* (here *G* is greater than or equal to *n*) while using shares of the same size.

Table 7: The time spent to reconstruct our scheme by a computer

Description	Share size (pixels)	*G*	Number of Boolean operation	Time required*
Shared visual secret key	256×256	256	2^{24}	16 ms
Encryption		4096	2^{28}	268 ms
Decryption		32768	2^{31}	2 sec
Shared visual secret key	512×512	256	2^{27}	134 ms
Encryption		4096	2^{30}	1 sec
Decryption		32768	2^{33}	8 sec
Shared visual secret key	1024×1024	256	2^{30}	1 sec
Encryption		4096	2^{32}	4 sec
Decryption		32768	2^{35}	34 sec
Shared visual secret key	2048×2048	256	2^{33}	8 sec
Encryption		4096	2^{34}	17 sec
Decryption		32768	2^{37}	2.29 min
Shared visual secret key	4096×4096	256	2^{36}	1.14 min
Encryption		4096	2^{36}	1.14 min
Decryption		32768	2^{39}	9.16 min
Shared visual secret key	8192×8192	256	2^{39}	9.16 min
Encryption		4096	2^{39}	9.16 min
Decryption		32768	2^{41}	36.65 min
Shared visual secret key	16384×16384	256	2^{42}	1.22 h
Encryption		4096	2^{42}	1.22 h
Decryption		32768	2^{43}	2.44 h

* ms = milliseconds, sec = seconds, min = minutes, h = hours

5. Working example of the proposed scheme

In this section, we will present an example of applying our proposed scheme to a black and white secret image. Fig. 5 shows one of the experimental results. In the initialization phase of establishing the shared visual secret key *SVSK*, the first party (Alice) and the second party (Bob) agree on an integer G with $G \geq 2$ and a common shadow image *PU*. For simplicity we assume they choose $G = 16$ and a common shadow image *PU* with size 512×512 pixels as shown in Fig. 5(a). Also, in the same phase, each party constructs his/her G+1 visual private keys by using the (k, k) ProbVC scheme as shown in [18, 31]. Each party generates his/her visual public keys (i.e., PUA_1 and PUA_2 for Alice and PUB_1 and PUB_2 for Bob) and sends them to the other party as shown in Fig. 5(b)-(e), and then they establish the shared visual secret key ($SVSK = SVSK_A = SVSK_B$), as shown in Fig. 5(f). In the encryption phase we assume that the sender takes a message of 512×512 pixels image as shown in Fig. 5(g) as the secret image *SI*. The secret image *SI* encrypts into two 512×512 shadow images (shares) by using the (2, 2) ProbVC scheme as shown in Subsection 2.2, where the first share is equal to the sender's shared visual secret key *SVSK* (*SVSK* serves as an encryption key) and the second share is the ciphered image *CI* as shown in Fig. 5(h) which will be sent to the receiver party. On the side of the receiver party, because the sender's shared visual secret key is equal to the receiver's shared visual secret key, the receiver's shared visual secret key *SVSK* serves as a decryption key. Fig. 5(i) shows that the secret image's message can be recovered by stacking the receiver's shared visual secret key *SVSK* and the ciphered image *CI*.

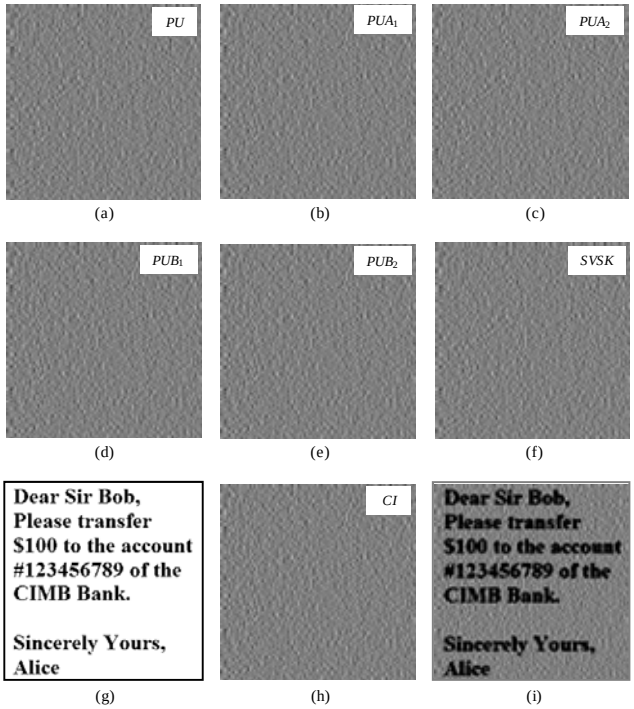

Fig. 5 Experimental results: (a) Common shadow image, (b)-(c) Alice's visual public keys, (d)-(e) Bob's visual public keys, (f) Shared visual secret key, (g) Secret image, (h) Ciphered image, and (i) Recovered image.

6. Conclusions

In this paper, we presented a new public-key encryption scheme based on non-expansion visual cryptography and Boolean operation. Our scheme allows one party to send a secret image to another party over the open network, even if many eavesdroppers listen. We used simple Boolean operations to construct our scheme, in which the secret image can encrypt and decrypt easily without complex computations. Therefore, our scheme can be useful in many applications. Our scheme gives reliable security especially when using a large value of the G and a large size of the share

References

[1] Y. C. Hou, and S. F. Tu, "A Visual Cryptographic Technique for Chromatic Images Using Multi-pixel Encoding Method", Journal of Research and Practice in Information Technology, Vol. 37, No. 2, (2005), pp. 179-192.

[2] Y. C. Hou, "Visual cryptography for color images", Pattern Recognition, Vol. 36, No. 7, (2003), pp. 1619-1629.
[3] A. MS, "Public Key Cryptography-Applications Algorithms and Mathematical Explanations", Tata Elxsi Ltd, (2007).
[4] I. Ozturk, and I. Sogukpinar, "Analysis and Comparison of Image Encryption Algorithms", International Journal of Information Technology, Vol. 1, No. 2, (2005), pp. 64-67.
[5] C. Chan, and Y. Wu, "A Visual Information Encryption Scheme Based on Visual Cryptography and D-H Key Agreement Scheme", International Journal of Computer Science and Network Security, Vol. 8, No. 4, (2008), pp. 128-132.
[6] W. Stallings, Cryptography and Network Security-Principles and Practices, Prentice Hall, Inc, 4th Ed., (2006).
[7] W. Diffie, and M. Hellman, "New Directions in Cryptography", IEEE Transactions in Information Theory, Vol. IT-22, No. 6, (1976), pp. 644-654.
[8] K.-Y. Chen, "The study and Implementations of Certificates in PKI", Ph.D. Thesis, Department of Electrical Engineering, National Cheng Kung University, Taiwan, (2004).
[9] C.-S. Laih, and K. Y. Chen, "Generating visible RSA public keys for PKI", International Journal of Information Security, Vol. 2, No. 2, Springer-Verlag, Berlin, (2004), pp. 103-109.
[10] J. J. Amador, and R. W. Green, "Symmetric-Key Block Cipher for Image and Text Cryptography", International Journal of Imaging and Technology, Vol. 15, No. 3, (2005), pp. 178-188.
[11] I. Muecke, "Greyscale and Colour Visual Cryptography", M.Sc. thesis, Faculty of Computer Science, Dalhousie University, USA, (1999).
[12] W. D. Shun, Z. Lei, M. Ning, and H. L. Sheng, "Secret Color Images Sharing Schemes Based on XOR Operation", Department of Computer Science and Technology, Tsinghua University, Beijing, China, (2005).
[13] M. Naor, and A. Shamir, "Visual cryptography", Advances in Cryptology-EUROCRYPT'94, lecture Notes in Computer Science, Vol. 950, Springer-Verlag, Berlin, (1995), pp. 1-12.
[14] M. Nakajima, and Y. Yamaguchi, "Extended Visual Cryptography for Natural Images", Department of Graphics and Computer Sciences, Graduate School of Arts and Sciences, University of Tokyo, (2002).
[15] C. Lin, and W. Tsai, "Visual cryptography for gray-level images by dithering techniques", Pattern Recognition Letter, Vol. 24, No. 1-3, (2003), pp. 349-358.
[16] D. Tsai, T. Chen, and G. Horng, "A cheating prevention scheme for binary visual cryptography with homogeneous secret images", Pattern Recognition, Vol. 40, No. 8, (2007), pp. 2356-2366.
[17] C. Yang, and T. Chen, "Colored visual cryptography scheme based on additive color mixing", Pattern Recognition, Vol. 41, No. 10, (2008), pp. 3114-3129.
[18] R. Ito, H. Kuwakado, and H. Tanaka, "Image size invariant visual cryptography", IEICE Trans. Fund., Vol. E82-A, No.10, (1999), pp. 2172-2177.
[19] M. Hellman, "An Overview of Public Key Cryptography", Communications Magazine, Vol. 16, No. 6, (1978), pp. 24-32.
[20] Y. Xue, "Overview of Public-Key Cryptography", CS 291: Network Security, Department of Electrical Engineering and Computer Science, Vanderbilt University, (2006).
[21] R. A. Rivest, A. Shamir, and L. Adleman, "A method for Obtaining Digital Signatures and Public-Key Cryptosystems", Communications of the ACM, Vol. 21, No. 2, (1978), pp. 120-126.
[22] T. ElGamal, "A Public-key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", IEEE Transactions on Information Theory, Vol. IT-31, No. 4, (1985), pp.469-472.
[23] N. Koblitz, "Elliptic Curve Cryptosystems", Mathematics of Computation, Vol. 48, No. 177, (1987), pp. 203-209.
[24] A. D. Bonis, and A. D. Santis, "Secret Sharing and Visual Cryptography Schemes", Proceedings of the IFIP TC11 16th International Conference on Information Security, (2001), pp. 123-138.
[25] R. Youmaran, A. Adler, and A. Miri, "An Improved Visual Cryptography Scheme for Secret Hiding", 23rd Biennial Symposium on Communications, (2006), pp. 340-343.
[26] Z. Zhou, "Advances on Digital Video and Visual Cryptography", Ph.D. thesis, Faculty of Electrical Engineering, Delaware University, (2004).
[27] D. Q. Viet, K. Kurosawa, "Almost Ideal Contrast Visual Cryptography Scheme with Reversing", lecture Notes in Computer Science, Vol. 2964, Springer, (2004), pp. 353-365.
[28] C. Hsu, and Y. Hou, "Visual cryptography and statistics based method for ownership identification of digital images", in: Proceedings of the International Conference on Signal Processing (ICSP'2004), Istanbul, Turkey, (2004), pp. 221-224.
[29] C.-S. Hsu, "A study of Visual Cryptography and Its Applications to Copyright protection Based on Goal programming and Statistics", Ph.D. Dissertation, Department of Information Management, National Central University, Taiwan, (2004).
[30] H.-C. Lin, "New Digital Image Encryption/Decryption Algorithms and Hidden Visual Cryptography Algorithm", M.Sc. thesis, Institute of Communication Engineering, Tatung University, (2004).
[31] C.-N. Yang, "New visual secret sharing schemes using probabilistic method", Pattern Recognition Letter, Vol. 25, No. 4, (2004), pp.481-494.
[32] S.-F. Tu, "On the design of protection scheme for digital images and documents based on visual secret sharing and Steganography", Ph.D. Dissertation, Department of Information Management, National Central University, Taiwan, (2004).

Abdullah M. Jaafar was born in Taiz, Republic of Yemen in March 7, 1977. He received the B.Sc. degree in Computer Science from Al-Mustansiriyah University, Baghdad, Republic of Iraq in 1999, and the M.Sc. degree in Computer Science from Iraqi Commission for Computers and Informatic, Institute for Post Graduate Studies in Informatic, Baghdad, Republic of Iraq in 2003. During 2004-2006, he worked as a lecturer at Taiz University in Republic of Yemen. Currently he is a Ph.D. student at the School of Computer Sciences, Universiti Sains Malaysia.

Azman Samsudin is a lecturer at the School of Computer Sciences, Universiti Sains Malaysia. He received the B.Sc. degree in Computer Science from University of Rochester, USA, in 1989. He obtained his M.Sc. and Ph.D. degrees in Computer Science from University of Denver, USA, in 1993 and 1998, respectively. His current research interests are in the fields of Cryptography and Parallel Distributed Computing.