A Compressive Sensing Based Image Encryption and Compression Algorithm With Identity Authentication and Blind Signcryption

XINYAN LI¹,², DI XIAO², (Member, IEEE), HUAJIAN MOU³, DIANJUN LU⁴, AND MEI PENG³

¹School of Mathematics and Statistics, Yangtze Normal University, Chongqing 408100, China
²Key Laboratory of Dependable Service Computing in Cyber Physical Society of Ministry of Education, College of Computer Science, Chongqing University, Chongqing 400044, China
³College of Computer Engineering, Yangtze Normal University, Chongqing 408100, China
⁴School of Mathematics and Statistics, Qinghai Normal University, Qinghai 810008, China

Corresponding author: Di Xiao (xiaodi_cqu@hotmail.com)

This work was supported in part by the Research Program of Chongqing Education Commission under Grant KJQN202001438 and Grant KJQN202001436, in part by the Applied Basic Research Project of Qinghai Province under Grant 2019-ZJ-7099, in part by the Chongqing Research Program of Basic Research and Frontier Technology under Grant csc2017jcjBX0008, and in part by the Scientific Research Innovation Team Project Affiliated to Yangtze Normal University under Grant 2016XJTD01.

ABSTRACT Recently, a robust and secure image sharing scheme with personal identity information embedded was proposed based on Compressive Sensing, Secret Image Sharing and Diffie-Hellman Agreement. However, there exists a security flaw in this scheme. It cannot resist the man-in-the-middle attack in the authentication stage. Anyone can disguise himself as a legal person and get the information when exchanging the secret keys, which provides the possibility for information leakage, tampering, and other attacks. In this paper, we propose an image encryption and compression algorithm with identity authentication and blind signcryption based on Parallel Compressive Sensing (PCS), Secret Sharing(SS) and Elliptic Curve Cryptography (ECC). Firstly, Logistic-Tent system and PCS are employed to complete compression and lightweight encryption in the compression stage. Secondly, random sequences are generated based on Chebyshev map to construct four encryption matrices to perform the encryption process. Meanwhile, the participants’ identity authentication and blind signcryption can be achieved by using ECC. Finally, we prove the efficiency and security of the blind signcryption, which can authenticate the participants’ identity before restoring the original image. Experiments and security analysis demonstrate that the proposed scheme not only reduce the storage space and computational complexity effectively, but also has resistance against the man-in-the-middle attack, forgery attack and chosen-text attack.

INDEX TERMS Parallel compressive sensing, secret image sharing, elliptic curve cryptographic algorithm, identity authentication, blind signcryption.

I. INTRODUCTION

In recent years, big data and artificial intelligence bring great convenience to people through the Internet of Things. A large amount of image information can be stored and transmitted as quickly as possible. However, there are many problems during image transmission and storage, such as illegal personnel invasion, malicious tampering and legitimate personnel information forgery in the transmission process, and space limitation in the storage process. Meanwhile, there also exist the problems of information leakage, low transmission efficiency and the low-security level. Encryption [1], [2] can make information chaotic and keep it secure. As we known, Elliptic Curve Cryptography (ECC) [3] is one of the best public key cryptosystems. It can be defined as the bilinear map among groups based on Weil pairing or Tate pairing [4], while the bilinear map has numerous applications in cryptography, such as identity when encryption. ECC has many advantages. For example, it provides an equivalent or higher level of security through using the key smaller than other public key cryptography, such as RSA or DSA [5]. ECC not only can be used for encryption [6]–[8], but also can be used for authentication [9], [10] and key management [11]. Based on ECC, many signcryption schemes...
have been proposed [12]–[15], which achieve signature and encryption simultaneously. Meanwhile, when facing illegal data usage, none of copyright protection, access control and encryption are enough to ensure information security, so it is essential to turn to the Secret Sharing scheme (SS) [16], which can prevent information from being too centralized and ensure information with more security. SS was firstly put forward by Shamir and Blakley in 1979. The SS algorithm decomposes the secret information into a few meaningless shares or shadows. Only the legal participants can reconstruct the secret information through recovery algorithm. Many secure image sharing schemes were proposed [17]–[21]. By combining with Compressive Sensing (CS) [22], they achieved a higher level of security and smaller storage space than others. In recent years, CS [23]–[25] exploits far efficiency than the conventional sampling under the Shannon theorem and achieves compression and lightweight encryption simultaneously. Based on Parallel Compressive Sensing (PCS) [26], cryptographic specialists also focus on other technique to combine encryption and compression [27]. However, none of them can verify the participants’ identity authentication and achieve tampering and forgery prevention. In [28], Wang et al. proposed an image compression scheme with personal identity information based on CS and Secret Image Sharing. But this scheme cannot resist the man-in-the-middle attack in the authentication stage, so anyone can disguise himself as a legal person to access the information.

This paper first gives a detailed introduction and analysis of the scheme in [28], and then proposes a CS based image encryption and compression algorithm with identity authentication and blind signcryption. Firstly, in the compression phase, Logistic-Tent system is adopted to generate the measurement matrix of PCS, and the original image is compressed by PCS. Then Chebyshev map and four matrices are employed to encrypt the compressed values. Secondly, the encrypted values are divided into many shadows under image sharing algorithm. Meanwhile, a bind signcryption system with personal identity information based on CS and Secret Image Sharing. However, there exists drawback in this scheme and the analysis is given. The specific steps are as follows.

II. THE PROCESS OF WANG’S SCHEME
Wang’s scheme in [28] is illustrated in this part, which contains the encryption process and decryption process, respectively. The encryption process includes agreement, constructing sensing matrix, sampling, quantization, sharing, and transmission; while the decryption process includes authentication, combination, anti-quantization and reconstruction. However, there exists drawback in this scheme and the analysis is given. The specific steps are as follows.

A. THE ENCRYPTION PROCESS (Encoding PHASE)
1) AGREEMENT
Step1: The distributor D publishes \((P,f,g)\) among participant \(P_i\), holds \((e,v)\) securely, where \(ev = 1(modP)\), \(P\) is a large integer. Meanwhile, the order of \(g\) modulo is \(f\) and satisfies \(g^f = 1(modP)\).

Step2: The participant \(P_i\) randomly chooses an integer \(b_i \in [1,f]\), computes \(R_i = g^{-b_i}(modP)\), and sends \(R_i\) to the distributor by the public channel.

Step3: When receiving \(R_i\), the distributor D computes \(H_i = (R_i)^{e}(modP)\) and \(R_0 = g^{-e}(modP)\), then publicly sends \(R_0\) to the participant \(P_i\).

Step4: The participant \(P_i\) computes \(H_i = (R_0)^{b_i}(modP)\) by their own integer \(b_i\) respectively. Denotes

\[PINs = [PIN_1, PIN_2, \ldots, PIN_n] = [H_1, H_2, \ldots, H_n].\]

2) CONSTRUCTING THE SENSING MATRIX
Step1: Under the initial parameters \((\mu, z_0)\) of the Tent map, generate a chaotic sequence: \(Z(d, l, \mu, z_0) = \{z_0, z_{h+d}, \ldots, z_{h+l-1d}\}\) with the length \(l = M \times N\).

Step2: A regularized sequence

\[Q(d, l, \mu, z_0) = [q_0, q_1, \ldots, q_{l-1}].\]

can be further generated from \(Z(d, l, \mu, z_0)\) as follows:

\[q_i = 1 - 2z_{m+i}, i \in \{0, 1, \ldots, l - 1\}.\]
Step 3: Compose a chaotic matrix $\Phi_{M \times N}$ from $Q(d, l, \mu, z_0)$ as follows:

$$
\Phi = \frac{1}{\sqrt{M \sigma^2}} \begin{bmatrix}
q_0 & q_M & \cdots & q_M(N-1) \\
q_1 & q_M+1 & \cdots & q_M(N-1)+1 \\
& & \ddots & \vdots \\
q_{M-1} & q_{M-1} & \cdots & q_{MN-1}
\end{bmatrix},
$$

where $\frac{1}{\sqrt{M \sigma^2}}$ is normalization, $\sigma = 0.5$ is the variance of the sequence $Q(d, l, \mu, z_0)$.

3) SAMPLING

Compute $Y = \Phi I$, where $I \in R^{N \times N}$ is a two-dimensional image and $Y \in R^{M \times MN} (M \ll N)$ is the measurement.

4) QUANTIZATION

Step 1: Rearrange $Y$ column by column as:

$$y = \text{vec}(Y) = \begin{bmatrix} y_0, y_1, \cdots, y_{MN-1} \end{bmatrix}.$$  

Step 2: Compute $\tilde{y}_i = H \left( \frac{2}{1+e^{E(y_i)-E(y_j)}} - 1 \right)$, where $E(y)$ is the mean of $y$, $\lambda$ is the parameter to adjust the output boundary $H$, and $\tilde{y}_i \in (-H, H)$.

Step 3: Compute $D(\tilde{y}_i) = c_i$, then encode it into an 8-bit $b_j$: $E(c_i) = b_j$, where $D(\cdot)$ and $E(\cdot)$ denote the discretization and the binary coding operation, respectively. Denote $B = [b_0, b_1, \cdots, b_{MN-1}]$, and send the bitstream to the local distributor by public channel.

Step 4: After receiving the bitstream $B = [b_0, b_1, \cdots, b_{MN-1}]$ from the local source encoder, the distributor $D$ takes 8-bits from $B$ sequentially to form a real value vector $P = [p_0, p_1, \cdots, p_{MN-1}]$, then construct a matrix $F$:

$$F = \begin{bmatrix}
p_0 & p_M & \cdots & p_{MN-1} \\
p_1 & p_{M+1} & \cdots & p_{M(N-1)+1} \\
& & \ddots & \vdots \\
p_{M-1} & p_{2M-1} & \cdots & p_{MN-1}
\end{bmatrix} \mod 251.$$

5) SHARING

Step 1: The participant’s identification numbers $\mathbf{PN}_i = [\mathbf{PN}_1, \mathbf{PN}_1, \cdots, \mathbf{PN}_n]$ are transformed into a decimal set $\mathbf{PN}_i' = [\mathbf{PN}_1', \mathbf{PN}_1', \cdots, \mathbf{PN}_n']$ with 8-bit per-coefficient. And compute $\mathbf{PN}_i'' = \mathbf{PN}_i' \mod g^i + (i-1)g^i$, $i \in [1, n]$, and $\mathbf{PN}_i''$ is scaled down to $\bar{w}_i, \bar{w}_i$, the matrix $F_i$ and $W_i$ are obtained by the following equation:

$$w_i = \begin{bmatrix} w_0, w_1, \cdots, w_{\bar{w}_i-1} \end{bmatrix}.$$  

Step 3: Let the $(t-1)$ degree polynomial

$$f(x) = a \cdot x = [a_0, a_1, \cdots, a_{t-1}] \begin{bmatrix} 1, x, x^2, \cdots, x^t \end{bmatrix} \mod p',
$$

and $g(x, y) = [x^0, x^1, x^2, \cdots, x^{t-1}, y^0, y^1, \cdots, y^{t-2}]$, where the prime $p' = 251$, $x$ is a base and $y$ is the size of the output vector.

Step 4: For the matrix $F$ and $W_i$, the shadow $I_i$ is obtained by the following equation:

$$I_i = F \cdot \begin{bmatrix} g(a_0, t), g(a_1, t), \cdots, g(a_{\bar{w}_i-1}, t) \end{bmatrix}^T.$$

Step 5: Repeat steps (2) and (4) until the shadows $\{I_1, I_2, \cdots, I_n\}$ are obtained.

6) TRANSMISSION

Transmit $\{I_1, I_2, \cdots, I_n\}$ to $n$ participants by the public channel.

B. THE DECRYPTION PROCESS OF WANG’s SCHEME (DECODE PHASE)

1) AUTHENTICATION

The combiner verifies whether the equation $\mathbf{HN}_i' = H_i' = (g^{-b_i})^v = g^{-b_i} = R_i (\mod P)$ holds or not. If yes, it means they are legitimate participants, otherwise, they are illegal participants.

2) COMBINATION

By Lagrange interpolation and the corresponding $\mathbf{HN}_i$, the matrix $F$ and $I_i$ can be reconstructed with no less than $t$ verified shadows.

3) ANTI-QUANTIZATION

After the inverse operations and the decimal-to-binary conversion, the anti-quantizer is used to get the vektory and its matrix form is $Y = \text{vec}^{-1}(y)$.

4) RECONSTRUCTION

Input $(\mu, z_0)$ to the Tent map, generate the sensing matrix, then the original image can be reconstructed by the OMP algorithm.

C. ANALYZE THE PROBLEM OF WANG’s SCHEME

However, in the authentication process of decryption, the algorithm is vulnerable to the man-in-the-middle attack, and cannot authenticate the legitimate participant so that illegal personnel can tamper or forge the information, and transmit the wrong information to the combiner. The detailed proof process is as follows.

1) INITIALIZATION

The parameters are the same as Wang’s scheme. Firstly, the distributor $D$ generates a larger integer $P$, such that $P = pq, p = 2fp + 1, q = 2fq + 1, where p and q are two primes, $f, q$ are distinct primes. Then let $g$ be an integer with an order $f$, that is $g^f = 1 (\mod P)$. Finally, $(p, f, g)$ are published and $(p, q)$ are securely kept.

2) AGREEMENT

Step 1: The participant $P_i$ randomly chooses an integer $b_i \in [1, f]$, computes $R_i = g^{-b_i} (\mod P)$, and then publicly sends $R_i$ to $D$. If the attacker $E_i$ captures $R_i$, he randomly chooses an integer $b_i' \in [1, f]$, makes it $R_i = g^{-b_i'} (\mod P)$, then sends $R_i'$ to the distributor $D$.

Step 2: When receiving $R_i'$, $D$ generates a key-pair, such that $ev = 1 (\mod \phi(P))$, then computes $H_i' = (R_i')^e (\mod P)$. 

211678
and the integer $R_0 = g^{-x} (\mod P)$, and sends $R_0$ to the attacker $E_i$ by the public channel.

3) AUTHENTICATION

In the authentication process, the authorized combiner obtains $\nu$ from D by the secret channel, get $H'_{i} = (R'_{i})^{\nu} = g^{-\nu l_{i}} (\mod P)$ and $R'_{i}$ from the attacker $E_i$ by the public channel. Then the authorized combiner verifies whether the equation $(H'_{i})^{\nu} = R'_{i}$ holds or not.

In fact, since the equation $(H'_{i})^{\nu} = (R'_{i})^{\nu} = (g^{-\nu l_{i}})^{\nu} = R'_{i}$ holds, the authorized combiner believes that $E_i$ is a legitimate participant, and the combiner will be able to further transmit information with the attacker $E_i$. Therefore, the illegal personnel $E_i$ can invade in the system and tamper the information arbitrarily, or forge information and so on.

III. BASIC KNOWLEDGE

A. COMPRESSIVE Sensing(CS) AND PARALLEL COMPRESSIVE Sensing(PCS)

As is well known, Nyquist-Shannon sampling theorem [23] uses the traditional method for signal processing which follows the "sample-then-compress" framework. CS [24], [25] combines signal sampling and compression simultaneously, which saves the computing resources of information acquisition and exploits far efficiency than Shannon theorem. So CS is suitable for some special high-speed signal acquisition and processing system. Assuming $x$ is a natural image, $x$ has the representation with $\Psi$ ( $\Psi$ is an orthogonal basis):

$$x = \Psi \theta, \quad (1)$$

$x$ is called $k$-sparse when $\theta$ has only $k$ non-zero entries, where $k$ is the order of $x$, $k \ll N$. And $x$ can be precisely reconstructed in high probability with

$$M = O \left( K \cdot log \left( \frac{N}{R} \right) \right) (M \ll N).$$

The mathematical representation is:

$$y = \Phi x = \Phi \Psi \theta, \quad (2)$$

where $y$ denotes the random measurement values and $\Phi$ is a random measurement matrix with the size of $M \times N$. When $\Phi \Psi$ satisfies the Restricted Isometry Property (RIP) [17], $x$ can be reconstructed by solving the $l_1$-norm problem.

$$\min \| \hat{\theta} \|_1 \; s.t. \; y = \Phi \Psi \hat{\theta}, \quad (3)$$

where $\| \cdot \|_1$ denotes the $l_1$ norm of a vector.

Traditionally, CS operates on 1D signals. While sampling a multidimensional signal, the size of the measurement matrix is large, and the computational complexity dramatically increases. It is necessary to propose PCS [27], which requires lower storage and computational complexity than traditional CS. Firstly, a multidimensional signal $X$ with $N \times N$ is sparsified in basis, such as DCT, DWT, then denoted as $x_i$ with $N \times 1$, $X = [x_1, x_2, \ldots, x_N]$. Finally, it can be sampled in a column-wise manner by the same measurement matrix $\Phi$:

$$y_i = \Phi x_i = \Phi \Psi \theta_i = \Theta \theta_i, \quad (4)$$

where $\Theta = [\theta_1, \theta_2, \ldots, \theta_N]$ and its size is $N \times 1$. The measurement values $y = [y_1, y_2, \ldots, y_N]$ with the size of $M \times N$. Theoretically, when satisfies Restricted Isometry Property (RIP) [26], the original signal can be recovered column-by-column by the following equation:

$$\min \| \hat{\theta} \|_1 \; s.t. \; y = \Theta \hat{\theta}, \quad (5)$$

Many reconstruction algorithms can solve the equation, such as orthogonal matching pursuit (OMP), matching pursuit (MP) and convex optimization method.

Compared with the traditional CS, PCS has lower storage and computational complexity. This theory has a wide range of applications in image reconstruction, medical imaging, radar imaging, channel coding, and so on.

In order to resist the chosen-plaintext attack, this proposed scheme combines the counter mode to construct the measurement matrix $\Phi$ in the compression process and to construct the encryption matrices in the encryption stage.

B. ELLIPTIC CURVE Cryptography(ECC)

The elliptic Curve Cryptography (ECC) [3] is one of the best public key cryptosystems and provides an equivalent or higher level of security using smaller key than other public key cryptography, such as RSA or DSA [5]. The definition of ECC is based on the elliptic curve, which is a set of points that represents the Weierstrass equation:

$$y^2 + axy + by = x^3 + cx^2 + dx + e (\mod p), \quad (6)$$

the simplified equation $E_p(a, b)$ is:

$$y^2 = x^3 + ax + b (\mod p), \quad (7)$$

where $a, b$ are two constants that satisfy $4a^3 + 27b^2 \neq 0$, and $p$ is a prime or an integer shaped like $2^i$. Elliptic Curve Cryptography (ECC) was proposed by Victor Miller (IBM) and Neil Koblitz in 1985 [3]. This security is based on the discrete logarithm problem. Suppose $Q$ and $G$ are two points on the elliptic curve, an integer $d$ is found such that $Q = dG$. This process called Elliptic Curve Discrete Logarithm Problem. Its abbreviated form is ECDLDP [4], and it is computationally infeasible to find $d$. It can be used not only as the public key cryptography to encrypt information [5]–[8], but also as a signature system [9]–[15].

C. SECRET SHARING SCHEME

In order to prevent information from being too centralized and ensure information with more security, it is essential to employ the Secret Sharing scheme (SS). SS was first presented by Shamir and Blakley in 1979 [16], also known as Shamir’s threshold algorithm.
In this scheme, a secret $a_0$ can be divided into $n$ non-overlapping parts, and the sharing values are generated by $t-1$ degree polynomial interpolation as follows:

$$f(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{t-1}x^{t-1} \mod p,$$  \hspace{1cm} (8)

where $a_0, a_1, \cdots + a_{t-1} \in Z_p$, $a_0$ is the secret, and $p$ is a large prime number.

Then the sharing values are transmitted to $n$ receivers. The secret $a_0$ can be recovered from any $t(t < n)$ shadows by Lagrange interpolation \[17\] as follows:

$$a_0 = f(0) = \sum_{i=1}^{t} f(x_i) \prod_{j=1, j \neq i}^{t} \frac{-j}{i-j} \mod p,$$  \hspace{1cm} (9)

where $j = 1, 2, \cdots , t$.

By the Secret Sharing scheme, the proposed scheme can achieve secret image sharing.

**D. CHAOTIC MAPS**

To expand the key space and increase the key security, two chaotic maps are employed in this scheme. In the CS phase of the proposed scheme, the Logistic-Tent map \[27\] is employed to construct a measurement matrix, which is the combination of Logistic map and Skew Tent map, and has superior performance than them. The dynamic equations are as follows:

1) **LOGISTIC MAP**

The definition of the one dimension generalized logistic map is as follows:

$$z_{n+1} = \mu z_n (1- z_n), z_n \in (0, 1).$$ \hspace{1cm} (10)

When $\mu \in [3.57, 4)$, the map becomes chaotic state.

2) **SKEW MAP**

Skew Tent map is defined as

$$z_{n+1} = \begin{cases} \frac{z_n}{t}, & 0 < z_n < r \\ \frac{1-z_n}{1-r}, & r < z_n < 1 \end{cases},$$ \hspace{1cm} (11)

where the initial value $z_0 \in (0, 1)$ and the control parameter $r \in (0, 1)$.

3) **LOGISTIC-TENT MAP**

$$b_{n+1} = \begin{cases} \frac{(r(t_n(1-t_n)+0.5 \times (4-r)t_n)) \mod 1, & t_n < 0.5 \\ (r(t_n(1-t_n)+0.5 \times (4-r)(1-t_n)) \mod 1, & t_n > 0.5, \end{cases}$$ \hspace{1cm} (12)

where the in initial value $t_0 \in (0, 1)$, and the control parameter $r \in (0, 4]$. In the encryption phase of this scheme, Chebyshev map also plays an important role when generating chaotic sequences, and constructing four encryption matrices, which can be defined by equation(13).

4) **CHEBYSHEV MAP**

$$t_{n+1} = T_k(t_n) = \cos (k \cdot \arccos t_n), t_n \in [-1, 1].$$ \hspace{1cm} (13)

When the order of Chebyshev map $k > 2$, it has the positive Lyapunov index, and becomes chaotic state.

**IV. THE PROPOSED SCHEME**

Based on Wang’s scheme, this paper proposes a compressive sensing based image encryption and compression algorithm, which can reduce the storage space, authenticate the legitimate participant and perform blind signcryption. Meanwhile, this algorithm can resist the man-in-the-middle attack, forgery attack and chosen-text attack. This scheme contains the encoding phase and decoding phase.

**A. THE ENCODING PHASE**

The encoding phase employs Logistic-Tent, Chebyshev map, PCS, ECC and Secret Sharing scheme. There are four major stages, including the compression stage, encryption stage, sharing stage and blind signcryption stage. In the compression phase, Logistic-Tent system is adopted to generate the measurement matrix of PCS, and the original image is compressed by PCS. In the encryption stage, Chebyshev map and four matrices are employed to encrypt the compressed values. And in the sharing stage, the encrypted values are divided into many shadows under image sharing algorithm. In the blind signcryption stage, identity authentication and blind signcryption can be realized by ECC. The specific flow chart (FIGURE 1) is as follows.

1) **COMPRESSION STAGE**

In the compression stage, this algorithm employs the compression method in \[27\], which combines PCS and the counter mode, and can be immune to the chosen-plaintext attack. The detail steps are as follows.

**Step1:** The plain image $P$ with the size $N \times N$ is decomposed by the first-level DWT to obtain the low frequency coefficient $LL_1$ and the high frequency coefficients $LH_1$, $HL_1$, $HH_1$. Then $LL_1$ is decomposed by the second-level DWT to obtain $LL_2$, $LH_2$, $HL_2$, $HH_2$. Denote $LH_1$, $HL_1$, $HH_1$ and $LH_2$, $HL_2$, $HH_2$ as $X_i$.

**Step2:** To resist the chosen-plaintext attack, choose the initial value $(r, t_0)$ of the Logistic-Tent map, generate the sequence $N_i = (N_{i-1} + 1) \mod 2^n$, compute $r_i = (N_i \times 2^{-n} + r) \mod 4$ and $t_i = (N_i \times 2^{-n} + t_0) \mod 1$. If $N_0$ is the random initialization vector and $h$ is the distance, then $T(r, t, h) = \left\{t_{n_0+jh}\right\}_{j=1}^{MN}$ is constructed as follows:

$$b_{n+1} = \begin{cases} \frac{(r(t_n(1-t_n)+0.5 \times (4-r)t_n)) \mod 1, & t_n < 0.5 \\ (r(t_n(1-t_n)+0.5 \times (4-r)(1-t_n)) \mod 1, & t_n > 0.5, \end{cases}$$ \hspace{1cm} (12)

where the control parameter $r \in (0, 4]$ and the system initial value $t_0 \in (0, 1)$.

**Step3:** Set the threshold and calculation formula, generate $u_i = 1 - t_{n_0+jh}$, $v_i \in (-1, 1)$, and construct different
measurement matrices $\Phi_i$ with $M \times N (M \ll N)$, which can be described as:

$$\Phi_i = \sqrt{\frac{2}{M}} \begin{bmatrix} \nu_1 & \nu_{M+1} & \cdots & \nu_{MN-M+1} \\ \nu_2 & \nu_{M+2} & \cdots & \nu_{MN-M+2} \\ \vdots & \vdots & \ddots & \vdots \\ \nu_M & \nu_{2M} & \cdots & \nu_{MN} \end{bmatrix}.$$ 

**Step4:** For the high frequency coefficients LH$_1$, HL$_1$, HH$_1$ and LH$_2$, HL$_2$, HH$_2$, compute $Y_i = \Phi_i X_i$. Meanwhile, the low frequency coefficient LL$_2$ remains unchanged, and the compressed image $Y_i$ is obtained.

**2) ENCRYPTION STAGE**

Based on the compressed image $Y_i$ and Chebyshev map, this part proposes an encryption algorithm which contains the permutation and diffusion processes. Firstly, input the key pairs which have some features of the original image into a Chebyshev map, and obtain random sequences. Then construct four matrices to realize the processes of permutation and diffusion. The encrypted image $Y_i'$ can be achieved by the detail steps.

**Step1:** Compute $s = \sum_{i=1}^{M \times M} p_i$, where $p_i$ is i-th pixel value of plain image. Divide $s$ into many groups and every three data as a group. Sum the groups and denote each of them as $\lambda_0$, then compute $\lambda = \lambda_0 10^{-n}$, where $\lambda \in (0, 1)$.

**Step2:** Iterate Chebyshev map $3MN+h$ times with $(k, u_0 + \lambda)$ to get the sequence $A = (b_1, b_2, \cdots, b_{3MN+h})$, where $k$ is the parameter and $u_0$ is the initial value of Chebyshev map.

**Step3:** Get $A_1 = (b_1, b_2, \cdots, b_{MN})$, $A_2 = (b_{MN+1}, b_{MN+2}, \cdots, b_{2MN})$, $A_3 = (b_{2MN+1}, b_{2MN+2}, \cdots, b_{3MN})$. 

**3) Sharing stage**

**4) Blind signcrytion stage**
Denote

\[
A_1 = \begin{bmatrix}
  b_1 & b_{M+1} & \cdots & b_{MN-M+1} \\
  b_2 & b_{M+2} & \cdots & b_{MN-M+2} \\
  \vdots & \vdots & \ddots & \vdots \\
  b_M & b_{2M} & \cdots & b_{MN}
\end{bmatrix},
\]

\[
A_2 = \begin{bmatrix}
  b_{MN+1} & b_{MN+M+1} & \cdots & b_{2MN-M+1} \\
  b_{MN+2} & b_{MN+M+2} & \cdots & b_{2MN-M+2} \\
  \vdots & \vdots & \ddots & \vdots \\
  b_{MN+M} & b_{MN+2M} & \cdots & b_{2MN}
\end{bmatrix},
\]

\[
A_3 = \begin{bmatrix}
  b_{2MN+1} & b_{2MN+M+1} & \cdots & b_{3MN-M+1} \\
  b_{2MN+2} & b_{2MN+M+2} & \cdots & b_{3MN-M+2} \\
  \vdots & \vdots & \ddots & \vdots \\
  b_{2MN+M} & b_{2MN+2M} & \cdots & b_{3MN}
\end{bmatrix},
\]

and generate the cyclic matrix:

\[
R = \begin{bmatrix}
  b_1 & b_2 & b_3 & \cdots & b_M \\
  b_M & b_1 & b_2 & \cdots & b_{M-1} \\
  b_{M-1} & b_M & b_1 & \cdots & b_{M-2} \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  b_2 & b_3 & b_4 & \cdots & b_1
\end{bmatrix}.
\]

**Step4:** Permutation. \( Y'_i \) is the permuted image. For matrix \( A_1 \), by rearranging it in ascending order to obtain the index matrix \( A'_1 \), the compressed image \( Y'_i \) is permuted under the index matrix \( A'_1 \). The permuted image is denoted as \( Y''_i \).

**Step5:** Diffusion. \( Y''_i \) is the encrypted image, which can be calculated by the equations, and the specific steps are as follows:

- If \( A_1 = 0.31 \sim 0.4 \) or \( 0.61 \sim 0.8 \),
  \[
  Y''_i = A_2 \oplus Y'_i \oplus R.
  \]
- If \( A_1 = 0.1 \sim 0.3 \) or \( 0.8 \sim 0.9 \),
  \[
  Y''_i = A_3 \oplus Y'_i \oplus R.
  \]

Other intervals, \( Y''_i = \oplus Y'_i \oplus R \).

**3) SHARING STAGE**

Secret Sharing scheme (SS) is employed to prevent information from being too centralized, and ensure information more secure. Based on SS, the encrypted image \( Y''_i \) can be shared to \( \omega \) shadows as the following steps, and at least \( t \) shadows can recover \( Y''_i \).

**Step1:** Divide \( Y''_i \) into \( \frac{MN}{T} \) groups and each group gets \( t \) elements. Denote

\[
C_1 = (c_1(0), c_1(1), \cdots, c_1(t-1)),
\]

\[
C_2 = (c_2(0), c_2(1), \cdots, c_2(t-1)),
\]

\[
\vdots
\]

\[
C_{MN/T} = (c_{MN/T}, c_{MN/T}, \cdots, c_{MN/T}(t-1)),
\]

**Step2:** Sequentially select sequence \( C_1 \) as the sharing coefficients. Let

\[
f_1(x) = \left[ c_1(0) + c_1(1)x + c_1(2)x^2 + \cdots + c_1(t-1)x^{t-1} \right] \mod 251.
\]

**Step3:** Compute \([1, f_1(1)], [2, f_1(2)], \cdots, [t, f_1(t)]\).

**Step4:** Repeat Step (2) and (3) until all the elements in groups are processed. Then, obtain the shadows \( M_1 = f_1(j), M_2 = f_2(j), \cdots, M_\omega = f_\omega(j) \), \( \omega = \frac{MN}{T} \), \( j = 1, 2, \cdots, t \), respectively.

**4) BLIND SIGNCRYPTION STAGE**

To prevent attackers from capturing the shadows and generating some fake shadows, it must complete the blind signcrytion, which includes five processes: the initial stage, send request, identity verification, send feedback and blind signcrytion. This process consists of participants \( P_1, P_2, \cdots, P_\omega \), distributor \( D \) and combiner. The notation and definition can be described as TABLE 1.

**Step1 (The initial stage):** The distributor first generates the parameters of \( E_p(a, b), G, \) and \( n \).

**Step2 (Send request):** \( P_1, P_2, \cdots, P_\omega \) randomly choose an integer \( k_i \in [1, n-1] \) respectively, and compute \( R_i = k_i \cdot G \mod n = (x_{R_i}, y_{R_i}), \) where \( x_{R_i}, y_{R_i} \) are X-coordinate and Y-coordinate of \( R_i \); Let \( r_i = x_{R_i} \mod n \), and use their private keys to compute \( t_i = d_i + k_i \cdot H(ID_i\|r_i) \), where \( H \) is the hash function. Then send \( R_i, t_i, ID_i \) to the distributor.

**Step3 (Identity verification):** When the distributor receives \( R_i, t_i, ID_i \), he verifies whether the equation \( t_i \cdot G = R_i \cdot H(ID_i\|r_i) + Q_i \) holds or not. If yes, it means \( P_i \) is the legitimate participant, then this participant is allowed to save the shadow \( M_i \); otherwise, \( P_i \) is the illegal participant, and cannot keep the shadow \( M_i \).

When the combiner wants to access the shadows, he must execute the following procedure.

**Step4 (Send feedback):** The combiner randomly chooses an integer \( l \in [1, n-1] \), computes \( L = (l \cdot G) \mod p \), and sends \( L \) to participants by public channel.

**Step5 (Blind signcrytion):** When participant \( P_i \) receive \( L \), he uses the private key \( d_i \) and \( L \) to encrypt the shadow by \( \tilde{M}_i = \left( M_i \cdot d_i^{-1} \cdot L^{-1} \right) \mod p \), and computes \( S_i = (k_i - d_i \cdot \tilde{M}_i) \cdot L^{-1} \mod p \). Then send \( \tilde{M}_i, S_i \) and \( R_i \) to the combiner by public channel.

**B. THE DECODING PHASE**

The decoding phase contains four stages: verification, reorganization, decryption and restoration stage. Firstly, the blind shadows must be verified by the combiner. Then the original shadows are conserved by at least \( t \) participants. By the inverse operations of permutation and diffusion, \( Y_i \) can be recovered. Finally, with the help of the measurement matrix and reconstruction algorithm OMP, the column vector \( X_i \) can be restored, and the plain image can be obtained by the inverse
TABLE 1. Notation and definition.

| Notation | Definition |
|----------|------------|
| $E_p(a, b)$ | An elliptic curve $E_p(a, b)$ with the formula: $y^2 = x^3 + ax + b (\mod p)$ where $a, b$ are two constants that satisfy $4a^3 + 27b^2 \neq 0$, $p$ is a prime or an integer shaped like $2^i$, $q$ is the power, and $E_p(a, b)$ is public. |
| $G, n$ | $G$ is a point on elliptic curve $E_p(a, b)$ with order $n$, where $n$ is a large prime, $G$ is public. |
| $(d_B, Q_B)$ | The private key $d_B$ and the public key $Q_B$ of combiner, which satisfy $Q_B = d_B \cdot G (\mod p)$. |
| $(d_i, Q_i), i = 1, \cdots, \omega$ | The private key $d_i$ and the public key $Q_i$ of participants $P_i$, which satisfy $Q_i = d_i \cdot G (\mod p)$. |

operation of the wavelet decomposition. The flow chart is shown as FIGURE2.

1) VERIFICATION AND REORGANIZATION

**Step1**: When the combiner receives $\tilde{M}_i, S_i$ and $R_i$, he verifies whether the equation $M_i = (Q_i \cdot 1 \cdot \tilde{M}_i^{-1}) \mod p$, and conserves $M_i(i = 1, 2, \cdots, \omega)$. Otherwise, the message has been tampered. Re-transmission is needed until it is equal, or the combiner rejects $\tilde{M}_i$, $S_i$ and $R_i$.

**Step2**: Any $r$ participants can reconstruct $Y''_i$ with the help of their shadows $\tilde{M}_i$ and Lagrange interpolation.

2) DECRYPTION STAGE

**Step1**: Input $(k, u_0 + \lambda)$ into Chebyshev map, where $(k, u_0, \lambda)$ are the keys, and generate the sequence $A = (b_1, b_2, \cdots, b_{3MN+h})$.

**Step2**: Get the matrixes $A_1, A_2$ and $A_3$ from $A = (b_1, b_2, \cdots, b_{3MN+h})$.

**Step3**: Generate the cyclic matrix $R$ based on $b_1, b_2, \cdots, b_{3MN}$. Based on the exclusive or operation again, $Y'_i$ can be obtained, which can be calculated by

If $A_1 = 0.31 \sim 0.4$ or $0.61 \sim 0.8$, $Y'_i = A_2 \bigoplus Y''_i \bigoplus R$. If $A_1 = 0.1 \sim 0.3$ or $0.8 - 0.9$, $Y'_i = A_3 \bigoplus Y''_i \bigoplus R$.

Other intervals, $Y'_i = Y''_i \bigoplus R$.

**Step4**: For matrix $A_1$, rearrange it in ascending order, then get the index matrix $A'_1$. $Y_i$ can be obtained under the index matrix $A'_1$ and $Y'_i$.

3) RESTORING STAGE

To recover the plain image, the OMP algorithm, the measurement matrix $\Phi_i$ and the inverse operation of wavelet decomposition are employed to acquire the plain image.

**Step1**: Choose the initial value $(r, t_0)$ for the Logistic-Tent map, generate the sequence $N_i = (N_{i-1} + 1) \mod 2^8$. Compute $r_i = (N_i \times 2^{-n} + r) \mod 4$ and $t_i = (N_i \times 2^{-n} + t_0) \mod 1$, where $N_0$ is the random initialization vector.

**Step2**: Consist $T(r, t, h) = \{t_{n_0+jh}\}_{j=1}^{MN}$, where $h$ is the distance. Set threshold and the calculation formula, generate $v_i$, where $v_i = 1 - t_{n_0+jh}$, $v_i \in (-1, 1)$, and construct the measurement matrix $\Phi_i$.

**Step3**: With the help of the measurement matrix $\Phi_i$ and OMP algorithm, the column vector $X_i$ can be restored, and then the plain image $P$ can be obtained by the inverse operation of the wavelet decomposition.

V. CORRECTNESS PROOF AND SECURITY ANALYSIS

**Theorem 1**: The equation for verifying the identity of participants is correct.

In the identity verification process, the distributor must verify whether the equation $t_i \cdot G = R_i \cdot H(ID_i||r_i) + Q_i$ holds or not. If yes, it means they are legitimate participants; otherwise, they are illegal participants.

**Proof**: In fact,

$$R_i = (k_i \cdot G) \mod p = (x_{R_i}, y_{R_i})$$

$$t_i = d_i + k_i \cdot H(ID_i||r_i),\ r_i = x_{R_i} \mod p,$$

then

$$t_i \cdot G = [d_i + k_i \cdot H(ID_i||r_i)] \cdot G$$

$$= d_i \cdot G + k_i \cdot H(ID_i||r_i) \cdot G$$

$$= Q_i + R_i \cdot H(ID_i||r_i).$$

This proof process is valid, and can verify the identity of participants.

**Theorem 2**: The algorithm of blind signcryption is available.

In the designcryption process, when the combiner receives $\tilde{M}_i, S_i$ and $R_i$, he can decrypt the shadows $M_i$ using $\tilde{M}_i, Q_i$ and $\lambda$, because of $M_i = \{Q_i \cdot 1 \cdot \tilde{M}_i^{-1}\} \mod p$. Meanwhile, the combiner verifies whether the equation $\tilde{M}_i \cdot Q_i + S_i \cdot G \cdot L = R_i$ holds or not. If yes, this means $\tilde{M}_i$ is complete and correct;
otherwise, the message has be tampered. Re-transmission is needed until they are equal, or the combiner rejects \( \tilde{M}_i, S_i \) and \( R_i \).

Proof: In fact,
\[
S_i = (k_i - d_i \cdot \tilde{M}_i) \cdot L^{-1} \mod p,
\]
then
\[
\tilde{M}_i \cdot Q_i + S_i \cdot G \cdot L = \tilde{M}_i \cdot Q_i + (k_i - d_i \cdot \tilde{M}_i) \cdot G \cdot L^{-1} \cdot L = \tilde{M}_i \cdot Q_i + k_i \cdot G - d_i \cdot \tilde{M}_i \cdot G = \tilde{M}_i \cdot Q_i + k_i \cdot G - \tilde{M}_i \cdot Q_i = R_i.
\]
This proof process is also valid, which means \( \tilde{M}_i \) is complete and correct, and the combiner conserves \( M_i, (i = 1, 2, \cdots, \omega) \).

Theorem 3: The security of blind signcryption is based on ECCDLP, and all of the parameters are safe. This algorithm is immune to chosen-text attack.

As we known, solving the large integer decomposition and solving discrete logarithm problem in prime fields have the same difficulty, but solving the elliptic curve discrete logarithm problem is more difficult than the above two problems. The security of RSA depends on the length of modulus, while the security of the elliptic curve discrete logarithm problem depends on the number of points on the elliptic curve. Researchers have shown that the attack resistances of the elliptic curve cryptosystem implemented with 160bit length infield GF (2160) is equivalent to RSA with 1024 bit modulus. While ensuring the same security, ECC has a shorter key length and smaller storage space than others.

Proof: If attacker wants to obtain \( l \) from \( M_i = (Q_i \cdot \tilde{M}_i^{-1}) \mod p \), to obtain \( k_i \) from \( R_i = (k_i \cdot G) \mod p \), to obtain \( d_i \) and \( k_i \) from \( t_i = d_i + k_i \cdot H(ID_i || r_i) \), to obtain \( l \) from \( L = (l \cdot G) \mod p \) and to obtain \( d_i \) from \( Q_i = (d_i \cdot G) \mod p \), these are all equivalent to solving the elliptic curve discrete logarithm problem (ECCDLP). Therefore, all of the parameters are safe in this scheme, and
it means that this algorithm is immune to the chosen-text attack.

**Theorem 4:** The algorithm of identity verification can prevent the man-in-the-middle attack.

In the identity verification process, when the participant $P_i$ sends $R_i, t_i, ID_i$ to the combiner, if the attacker $E_j$ captures $R'_i, t'_i, ID'_i$ and modifies those to $R'_j, t'_i, ID'_j$, and then he sends $R'_j, t'_i, ID'_j$ to the combiner. When the combiner receives $R'_i, t'_i, ID'_i$, he verifies whether the equation $t'_i \cdot G = R'_i \cdot H(ID'_i || r'_i) + Q'_i$ holds or not.

**Proof:** if

$$t'_i = d'_i + k'_i \cdot H(ID'_i || r'_i),$$

then

$$t'_i \cdot G = \left[ d'_i + k'_i \cdot H(ID'_i || r'_i) \right] \cdot G = d'_i \cdot G + k'_i \cdot H(ID'_i || r'_i) \cdot P = R'_i \cdot H(ID'_i || r'_i) + Q'_i \neq R'_i \cdot H(ID'_i || r'_i) + Q_i.$$

In this equation, $Q_i$ is the public key of participant, and cannot be changed. The equation does not hold, and it means that there are illegal participants. This process can effectively prevent the man-in-the-middle attack.

**Theorem 5:** The algorithm of blind signcryption can prevent the forgery attack.

In the blind signcryption, if the attacker captures $\tilde{M}'_i, S'_i, L'_i$, modifies those to $\tilde{M}'_i, S'_i, L'_i$, and then he sends them to the combiner. When the combiner receives $\tilde{M}'_i, S'_i, L'_i$, he verifies whether the equation $\tilde{M}'_i \cdot Q_i + S'_i \cdot G \cdot L' = R'_i$ holds or not.

**Proof:** if

$$S'_i = \left[ (k'_i - d'_i \cdot \tilde{M}'_i) \cdot (L')^{-1} \right] \mod p$$

then

$$\tilde{M}'_i \cdot Q_i + S'_i \cdot G \cdot L' = \tilde{M}'_i \cdot Q_i + (k'_i - d'_i \cdot \tilde{M}'_i) \cdot (L')^{-1} \cdot G \cdot L' = \tilde{M}'_i \cdot Q_i + (k'_i \cdot G \cdot d'_i \cdot \tilde{M}'_i \cdot G) = \tilde{M}'_i \cdot Q_i + R'_i - \tilde{M}'_i \cdot Q'_i \neq R'_i,$$

where $Q'_i = d'_i \cdot G(\mod p)$ is the public key of attacker. It does not hold, which means that $\tilde{M}'_i$ are incorrect shadows. Re-transmission is needed until they are equal. This procedure can effectively prevent the forgery attack.

**VI. EXPERIMENTAL ANALYSIS AND PERFORMANCE COMPARISON**

To demonstrate the security and the efficiency of the proposed scheme, numerical experiments and performance comparisons are given in detail in this section, under the Matlab R2016a platform, a desktop machine with 3.4 GHz and 8GB memory. Five test images with $512 \times 512$ (Lena, Boat, Photography, Peppers, and Saturn) are employed. This section includes the aspects of PSNR, key space, key sensitivity, performance comparison, and so on. In the compression stage, the key pairs $r, t_0$ of the Logistic-Tent map are the secret keys, where $r = 3.0321, t_0 = 0.6122$. In the encryption stage, the secret keys are $k = 4, u_0 = 0.62354987$ and $\lambda$, where $\lambda$ is the characteristic of the plain image. Input the key pairs $(k, u_0 + \lambda)$ of Chebyshev map to generate random sequence, where the random sequence has $3 \times 256 \times 256 + 3000$ random data. The threshold scheme $(6, 8)$ is used in the sharing stage. Meanwhile, in the identity authentication and blind signcryption stage, when the encryption key of ECC is 200bit, it can ensure security and better than 176bit [33]. In the restore process, the OMP algorithm is employed to deal with the inverse operation of CS.

**A. PSNR**

The peak signal-to-noise ratio (PSNR) is used to evaluate the image quality of the reconstructed image. A larger value of the PSNR implies smaller distortion.

$$PSNR = 10 \log_{10} \left( \frac{255^2}{\text{MSE}} \right) \text{dB},$$

where mean square error (MSE) can be calculated by

$$\text{MSE} = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} (X_{xy} - X'_{xy})^2,$$

where $X_{xy}$ and $X'_{xy}$ are the pixel value of original image and the reconstructed image, respectively. In this scheme, all plain images $P$ with $512 \times 512$ are first compressed to a quarter, then encrypted and get the encrypted image $Y''$ with a size of $256 \times 256$. The PSNR values of Lena, Boat, Photography, Peppers and Saturn are 31.5463dB, 30.8446dB, 32.6072dB, 32.0690dB, 37.8722dB, respectively.

**B. HISTOGRAM ANALYSIS**

The histogram is one of the criteria in the cipher text to analyze the value distribution of an image, which is uniform and has random behavior in an ideal state. In the simulation, we choose five images to encrypt, the histograms of the original images and the encrypted images are shown in FIGURE 3. For the original images, the pixel value distribution is relatively concentrated, and the histogram distributions are not uniform at all. But for the encrypted images, they have uniform histograms by the effective image encryption algorithm.

**C. CORRELATION ANALYSIS**

Correlation analysis [7] is the relationships among pixels and their neighboring pixels for a natural image at horizontal, vertical and diagonal directions. The values of those relationships can be shown in TABLE 2. From the values, the correlation of adjacent pixels for plain image are all close to 1, while the values for cipher images are all close to 0, which means that the incoherence has satisfactory effect.
D. INFORMATION ENTROPY

Information entropy indicates the randomness of an information source, which definition is as follows:

\[ H(s) = - \sum_{i=1}^{2^N-1} p(s_i) \log_2 p(s_i), \]  

(16)

where \( s_1, s_2, \ldots, s_{2^N-1} \) are the sources, and \( p(s_i) \) is the probability of \( s_i \). Accordingly, the entropy of cipher image with 256 gray levels in an effective algorithm should ideally be 8. To explain the information entropy of the original image and the cipher image, the images in the 1st and 2nd columns of FIGURE 3 are selected as the test images, and then corresponding values of entropy for different images can be obtained. TABLE 3 reveals that the entropy values of all cipher images are close to 8 and have the ideal values.

E. KEY SPACE ANALYSIS AND BRUTE-FORCE ATTACK

To resist the brute-force attack, an effective algorithm must have enough large key space. Logistic-Tent map and Chebyshev map are employed in this algorithm. In the transform equations, \( t_0, u_0 \) are secret keys, \( r \in (0, 4) \) and \( k > 2 \) are the control parameter. The IEEE floating point standard [32] requests that the calculative precision of the 64-bit double-precision number is about \( 10^{-15} \). Therefore, the secret keys must be in \((0, 1)\) with \( 10^{15} \) possible values, the key space should be \((10^{15} \times 10^{15} \times 10^{15} \times 10^{15}) = 10^{60} \approx 2^{200}\). Thus, this scheme has large key space against brute-force attack.

F. KEY SENSITIVITY ANALYSIS

As we know, NPCR and UACI, defined in equations (17)(18)[7], are usually used to evaluate the change of cipher
image when little bit changes in the original image or the initial value. Lena is chosen to analyze the key sensitivity. In the ideal state, the average of NPCR is about 0.9961, and the average of UACI is about 0.3346 \[8\]. Suppose that \(P_1(i, j)\) and \(P_2(i, j)\) are the \((i, j)\) th pixel of the image \(P_1\) and \(P_2\). This character is called resisting differential attack.

\[
\text{NPCR} = \frac{\sum_{ij} \delta(i, j)}{M \times N} \times 100\% \tag{17}
\]

\[
\text{UACI} = \frac{1}{M \times N} \times \left[ \frac{\sum_{ij} \left| P_1(i, j) - P_2(i, j) \right|}{255} \right] \times 100\% \tag{18}
\]

where \(D(i, j) = 0\) if \(P_1(i, j) = P_2(i, j)\), otherwise \(D(i, j) = 1\).

1) SENSITIVITY ANALYSIS OF PLAINTEXT AND DIFFERENTIAL ATTACK
In this experiment, we randomly change two-bit, four-bit, six-bit, eight-bit in the original image of Lena, From TABLE 4, the NPCR and UACI of the encrypted image are close to 0.9961 and 0.3346, respectively. This means the key sensitivity has satisfactory effect to resist differential attack.

2) SENSITIVITY ANALYSIS OF KEY
For Lena, when we encrypt the same image with different initial values in a tiny change of \(10^{-11}, 10^{-12}, 10^{-13}\) and \(10^{-15}\), the NPCR and UACI of the encrypted image are shown in the following TABLE 5. From the TABLE 5, the NPCR and

**TABLE 1.** Correlation of adjacent pixels for five images.

| Direction | Lena | Boats | Photography | Peppers | Saturn |
|-----------|------|-------|-------------|---------|--------|
|            | Plain image | Cipher image | Plain image | Cipher image | Plain image | Cipher image | Plain image | Cipher image | Plain image | Cipher image |
| Horizontal | 0.9708 | 0.002 | 0.9482 | -0.0040 | 0.002315 | 0.0018 | 0.9812 | -0.000 | 0.9984 | -0.000 |
|           | 68425 | 31574 | 42697 | 777731 | 7428425 | 11754 | 28501 | 87727 | 76357 | 05827 |
| Vertical  | 0.9856 | 0.0028 | 0.9754 | -0.0028 | 0.9902 | 0.0106 | 0.9841 | -0.003 | 0.9986 | -0.003 |
|           | 42012 | 61637 | 54662 | 820236 | 07002 | 14606 | 79943 | 26670 | 55869 | 84184 |
| Diagonal  | 0.9574 | 0.0021 | 0.9303 | -0.00231 | 0.9735 | 0.0061 | 0.9700 | 0.0040 | 0.9956 | 0.0070 |
|           | 55450 | 15429 | 02571 | 03615 | 59500 | 77826 | 16453 | 56163 | 30295 | 03644 |

**TABLE 2.** Information entropies of five images.

| Lena | Boats | Photography | Peppers | Saturn |
|------|-------|-------------|---------|--------|
|      | Plain image | Cipher image | Plain image | Cipher image | Plain image | Cipher image | Plain image | Cipher image |
| Plain image | 7.59292865 | 7.197081373 | 7.989154251 | 7.990013289 | 4.12899755 |
| Cipher image | 7.98941243 | 7.98813436 | 7.989154251 | 7.99001328 | 7.993823644 |

**TABLE 3.** The values of NPCR and UACI when add two-bit, four-bit, six-bit, eight-bit.

| Lena | NPCR | UACI | NPCR | UACI | NPCR | UACI | NPCR | UACI |
|------|------|------|------|------|------|------|------|------|
|      | 0.99560 | 0.33746 | 0.99617 | 0.33977 | 0.99598 | 0.34330 | 0.99708 | 0.34177 |
|      | 546875 | 804243 | 004395 | 141467 | 6938477 | 8428599 | 5571299 | 850452 |

In the ideal state, the average of NPCR is about 0.9961, and the average of UACI is about 0.3346 \[8\]. Suppose that \(P_1(i, j)\) and \(P_2(i, j)\) are the \((i, j)\) th pixel of the image \(P_1\) and \(P_2\). This character is called resisting differential attack.
TABLE 5. The values of NPCR and UACI when changing a single data on the initial value.

|     | 10^{-11} | 10^{-12} | 10^{-13} | 10^{-15} |
|-----|----------|----------|----------|----------|
| NPCR | 0.996246 | 0.996200 | 0.995986 | 0.996353 |
| UACI | 0.340933 | 0.341446 | 0.347515 | 0.339755 |
| NPCR | 33789063 | 5615234  | 93847656 | 14941406 |
| UACI | 19108080 | 9111558  | 60178308 | 95154412 |

TABLE 6. The performance comparison with other schemes.

| Scheme | Compression | PSNR(dB) | Encryption | Identity Authenticity | Secret image sharing | Against the man-in-the-middle attack |
|--------|-------------|----------|------------|-----------------------|----------------------|-------------------------------------|
| [7]    | No          | No       | Yes        | No                    | No                   | No                                  |
| [27]   | Yes         | 33.42    | Yes        | No                    | No                   | No                                  |
| [28]   | Yes         | 28.04    | Yes        | Yes                   | Yes                  | No                                  |
| [30]   | Yes         | 32.53    | Yes        | No                    | No                   | No                                  |
| [31]   | Yes         | 35.4     | No         | No                    | Yes                  | No                                  |
| proposed | Yes      | 32.61    | Yes        | Yes                   | Yes                  | Yes                                 |

UACI values are all close to 0.9961 and 0.3346, respectively, which means it is in the ideal state and can resist differential attack.

G. RESIST KNOWN/CHOSEN PLAINTEXT ATTACK

In this scheme, in the compression process, the measurement matrix Φ is generated by (r, t₀) in the Logistic-Tent map and iteration, where t₀ and t₁ are the secret keys, r and k are the control parameters of chaotic systems. Meanwhile, in the encryption process, A₁, A₂, A₃ are generated by the Chebyshev map with the initial value u₀ + λ, where λ is the character of the original image, A₁ decides the indexes of permutation, A₂ and A₃ decide the diffusion value. If an attacker intercepts a part of plaintext and the corresponding ciphertext, he cannot get r, t₀, k, u₀ and λ. Therefore, this scheme can resist known/known plaintext attack, so any attacker cannot obtain the key even if he obtains plaintext.

H. THE PERFORMANCE AND COMPARISON

The security of this scheme is based on CS, Secret Sharing and the difficulty of solving the ECDLP. For other schemes in [27] and [28], they can complete compression and encryption, but cannot resist the man-in-the-middle attack. The schemes in [7] and [30] can complete encryption, but cannot satisfy identity authenticity and resist the man-in-the-middle attack. For this proposed scheme, it has superior performance, which not only completes compression, encryption and image sharing, but also has the characters of identity authenticity and against the man-in-the-middle attack. The detailed comparisons are shown in TABLE 6.

VII. CONCLUSION

Based on Parallel Compressive Sensing, Secret Sharing and Elliptic Curve Cryptographic, this paper proposes an image encryption and compression algorithm, which achieves compression, encryption, identity authentication, and blind signature. The proposed algorithm can resist various attacks, such as the man-in-the-middle attack, forgery attack and chosen-text attack. Meanwhile, the proposed scheme has lower storage and computational complexity, high security and PSNR. By blind signature, the identity of participants and the shadows can be guaranteed, and the verifiability is strictly proved in this paper. Numerical experimental results, proofs and security analysis demonstrate that the scheme is secure and more practical when compared with other existing schemes.

REFERENCES

[1] G. J. Simmons, “Symmetric and asymmetric encryption,” ACM Comput. Surv., vol. 11, no. 4, pp. 305–330, Dec. 1979.
[1] Y. Zhang, D. Xiao, Y. Shu, and J. Li, “A novel image encryption scheme based on a linear hyperbolic chaotic system of partial differential equations,” Signal Process., Image Commun., vol. 28, no. 3, pp. 292–300, Mar. 2013.

[2] V. S. Miller, “Use of elliptic curves in cryptography,” in Proc. Adv. Cryptol.-CRYPTO, New York, NY, USA, 1985, pp. 417–426.

[3] N. Kobitz, “Elliptic curve cryptosystems,” Math. Comput., vol. 48, no. 177, pp. 203–209, 1987.

[4] S. Tyagi, M. Rana, N. Aggarwal, and P. Bhandana, “A comparative study of public key cryptosystem based on ECC and RSA,” Int. J. Comput. Sci. Eng., vol. 3, no. 5, pp. 1904–1909, May 2011.

[5] L. Tawalbeh, M. Mowafi, and W. Ajiboye, “Use of elliptic curve cryptography for multimedia encryption,” IET Inf. Secur., vol. 7, no. 2, pp. 67–74, Jun. 2013.

[6] Y. Luo, X. Ouyang, J. Liu, and L. Cao, “An image encryption method based on elliptic curve elgamal encryption and chaotic systems,” IEEE Access, vol. 7, pp. 38507–38522, 2019.

[7] S. Bendaoud, F. Amounas, and E. H. El Kinani, “A new image encryption scheme based on enhanced elliptic curve cryptosystem using DNA computing,” in Proc. 2nd Int. Conf. Netw., Inf. Syst. Secur. (NISS), 2019, pp. 1–5.

[8] H.-L. Yeh, T.-H. Chen, and W.-K. Shih, “Robust smart card secured authentication scheme on SIP using elliptic curve cryptography,” Comput. Standards Interfaces, vol. 36, no. 2, pp. 397–402, Feb. 2014.

[9] Z. Zhang, Q. Qi, N. Kumar, N. Chilamkurti, and H.-Y. Jeong, “A secure authentication scheme with anonymity for session initiation protocol using elliptic curve cryptography,” Multimedia Tools Appl., vol. 74, no. 10, pp. 3477–3488, May 2015.

[10] N. Samal, K. Georgantas, and A. Gurtov, “Lightweight authentication and key management on 802.11 with elliptic curve cryptography,” in Proc. IEEE Wireless Commun. Netw. Conf. (WCNC), Apr. 2015, pp. 1830–1835.

[11] A. Bracken and P. Poranhage, “Efficient generalized signcryption based on ECC,” Int. J. Cryptogr. Inf. Secur., vol. 5, no. 2, pp. 1–13, 2015.

[12] L. Pang, M. Wei, and H. Li, “Efficient and anonymous certificateless multi-message and multi-receiver signcryption scheme based on ECC,” IEEE Access, vol. 7, pp. 24511–24526, 2019.

[13] M. Kumar and P. Gupta, “An efficient and authentication signcryption scheme based on elliptic curves,” MATHEMATIKA, vol. 35, no. 1, pp. 1–11, Apr. 2019.

[14] C.-H. Tsai and P.-C. Su, “An ECC-based blind signcryption scheme for multiple digital documents,” Secur. Commun. Netw., vol. 2017, pp. 1–14, 2017.

[15] A. Shamir, “How to share a secret,” Commun. ACM, vol. 22, no. 11, pp. 612–631, 1979.

[16] L. Liu, A.-H. Wang, C.-C. Chang, Z.-H. Li, and J. Liu, “A lossy secret color image sharing scheme with small shadow and error-resilient capability,” J. Inf. Hiding Multimedia Signal Process., vol. 6, no. 2, pp. 246–253, 2015.

[17] K. M. Faroum, “Design of a new efficient and secure multi-secret images sharing scheme,” Multimedia Tools Appl., vol. 76, no. 5, pp. 6247–6261, Mar. 2017.

[18] Y. Xiang, D. Xiao, H. Wang, and X. Li, “A secure image tampering detection and self-recovery scheme using P0B number system over cloud,” Signal Process., vol. 162, pp. 282–295, Sep. 2019.

[19] J. Xie, L. Li, H. Peng, and Y. Yang, “A secure and efficient scalable secret image sharing scheme with flexible shadow sizes,” PLoS ONE, vol. 12, no. 1, 2017, Art. no. e0168674.

[20] X. Li, X. Meng, Y. Wang, X. Yang, Y. Yin, X. Peng, W. He, G. Dong, and H. Chen, “Secret shared multiple-image encryption based on row scanning compressive ghost imaging and phase retrieval in the fresnel domain,” Opt. Lasers Eng., vol. 96, pp. 7–16, Sep. 2017.

[21] T. Bhattacharjee, S. P. Maity, and S. R. Islam, “Hierarchical secret image sharing scheme in compressed sensing,” Signal Process., Image Commun., vol. 61, pp. 21–32, Feb. 2018.

[22] M. Elad, “Optimized projections for compressed sensing,” IEEE Trans. Signal Process., vol. 55, no. 12, pp. 5695–5702, Dec. 2007.

[23] Y. Zhang, J. Zhou, F. Chen, L. Y. Zhang, K.-W. Wong, X. He, and D. Xiao, “Embedding cryptographic features in compressive sensing,” Neurocomputing, vol. 205, pp. 472–480, Sep. 2016.

[24] Y. C. Eldar and G. Kutyniok, Compressed Sensing: Theory and Applications. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[25] H. Fang, S. A. Vorobyov, H. Jiang, and O. Taheri, “Permutation meets parallel compressed sensing: How to relax restricted isometry property for 2D sparse signals,” IEEE Trans. Signal Process., vol. 62, no. 1, pp. 196–210, Jan. 2014.
DIANJUN LU received the B.S. and M.S. degrees from Qinghai Normal University, Xining, China. From 2011 to 2012, he visited the Department of Statistics, University of California, Riverside, USA, as a Visiting Scholar. He is currently a Full Professor with the School of Mathematics and Statistics, Qinghai Normal University, Qinghai, China. His research interests include cryptography and information security. So far, he has published more than 40 academic journal articles.

MEI PENG received the M.S. degree from Chongqing Normal University, Chongqing, China. She is currently a Lecturer with the School of Computer Engineering, Yangtze Normal University. Her research interests include computer software, stochastic system analysis, and image processing.

***