This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

Flexible and Secure Data Transmission System based on Semi-Tensor Compressive Sensing in Wireless Body Area Networks

Lixiang Li, Lifei Liu, Haipeng Peng, Yixian Yang, and Shizhuo Cheng

***Abstract*—Wireless body area networks (WBANs) collect some physiological parameters of the human body. Each sensor uses limited energy to maximize its own life. There are three crucial problems including adaptiveness, energy and security in WBANs. In order to solve these problems, a flexible and secure data transmission system is proposed in this paper. The proposed scheme is composed of semi-tensor compressive sensing, hash function, Arnold scrambling and chaotic scrambling (SC-HAC). For the adaptiveness problem, our scheme uses semi-tensor compressive sensing to encrypt multiple signals with different dimensions. The chaotic sequence is applied to generate the semi-tensor measurement matrix. On the one hand, we only transmit a few chaotic parameters, which reduces the number of data storage and transmission. On the other hand, the size of the measurement matrix is small, and the computation overhead can be reduced. The security is considered by the proposed scheme which combines Arnold scrambling and Logistic scrambling to improve the encryption effect. Numerical simulations and security analyses are given to show that our scheme performs well. The total key space is approximately 2^{420}. The absolute value of adjacent pixel correlation is less than 0.004. Traditional compressive sensing method stores 524288 bytes, while the proposed scheme only stores 2048 bytes. When the compression ratio (CR) is less than 0.7, the peak signal to noise ratio (PSNR) of our scheme is obviously higher than those of other three schemes.**

***Index Terms*—wireless body area networks, semi-tensor compressive sensing, low-power consumption, security, image reconstruction.**

I. Introduction

WIRELESS body area networks (WBANs) are used to monitor patients remotely. Sensors are worn on the body or implanted in the skin to collect some physiological parameters of the human body, such as electrocardiogram (ECG), electroencephalogram (EEG), body temperature, blood pressure, respiration, body movement, and so on [1]. An evolution of pervasive healthcare has been presented from wearable sensors to smart implants [2]. The signals are sent to the data center through wireless transmission nodes. For instance, in Fig. 1, the sensor data are transmitted to the body sensor coordinator (such as computer, mobile phone etc.) by wireless communication technology (for examples, ZigBee, Bluetooth) in WBANs. ZigBee and Bluetooth have the characteristics of short distance and low power consumption. The signals are transmitted through the network to the terminal (for example, a hospital data center) [3]. Doctors can use these data to diagnose the patient without going to the patient's home or asking the patient to come to the hospital, which can save resources. Nowadays, WBANs are not only used in health monitoring, but also sport training, entertainment, military activities, etc. In terms of protocol framework, IEEE 802.15.4 and IEEE 802.15.6 are two security protocols related to WBANs, and the latter is designed specifically according to the characteristics of the network. These two protocols provide the international standard for security, reliability, low power consumption, and long distance transmission.

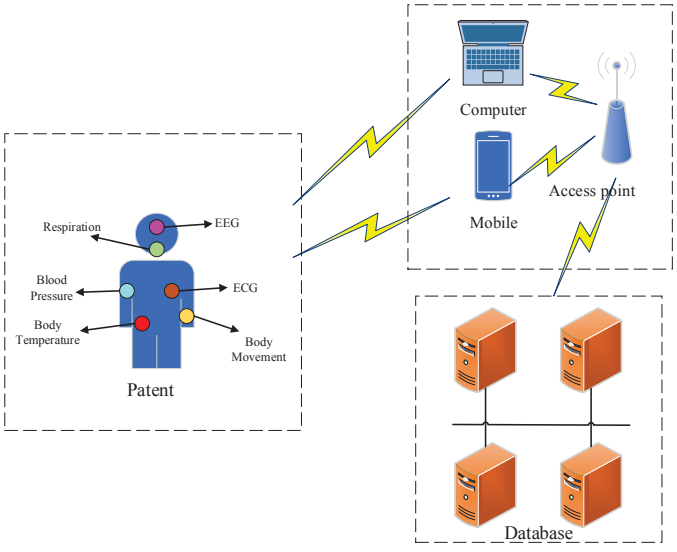


Fig. 1. Data transmission in wireless body area networks.

Adaptiveness is a key problem in wireless body area networks. The types of signals are different, such as ECG, EEG. And the sizes of signals are also different. Compressive sensing (CS) [4]–[6] can be applied to WBANs to measure signals. According to the rule of matrix multiplication in CS, the column number should equal to the signal length. To reduce the size of the measurement matrix, many works

Lixiang Li, Lifei Liu, Haipeng Peng, and Yixian Yang are with Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China, and National Engineering Laboratory for Disaster Backup and Recovery, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: lixiang@bupt.edu.cn; m15175291673@163.com; penghaipeng@bupt.edu.cn; yxyang@bupt.edu.cn) (Corresponding author: Lixiang Li).

Shizhuo Cheng is with the JangHo Architecture College, Northeastern University, Liaoning 110169, China (e-mail: chengshizhuo@mail.neu.edu.cn).

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

focused on reducing the row number of the measurement matrix [7] [8]. Besides, dividing the signals into blocks is another method to reduce the matrix size [9]. However, these methods need to design a specific measurement matrix for a specific size. In the proposed scheme, semi-tensor compressing sensing (STP-CS) is used to reduce the row and column numbers of the measurement matrix simultaneously, and it can measure various sizes by using the same measurement matrix.

Energy limitation is another important problem in WBANs, because the battery power is limited and the replacement is inconvenient. Each sensor uses limited energy to maximize its own life. Sensors are worn on the body or implanted in the skin in WBANs. For those sensors implanted in the skin, battery replacement and charging need to be completed by surgery. Energy harvesting is one solution, allowing the sensor to self-charge [10]. Other approaches to save energy include improving routing protocols or reducing data transmission size [11] [12].

However, existing schemes seldom focus on the security issue of medical health data. In terms of security in WBANs, the patient's health data are sensitive information. In wireless transmission, these health data are easily stolen by the eavesdropper, so it is very essential to implement good security [13] [14]. However, these two methods don't consider energy consumption [15] [16]. They are based on cryptography and analyze the security problem of health data. However the corresponding algorithms are relatively complex, and their running time is long. Therefore, it is important to design a simple and secure encryption scheme for energy saving.

In recent years, compressive sensing (CS) has been applied in wireless body area networks and image encryption. Due to the implementations of encryption and compression simultaneously, CS has attracted extensive attention [17]–[19]. Some schemes of image encryption were presented based on compressive sensing [20]–[22]. Zhang et al. [20] reviewed compressive sensing in the field of information security applications, including image, audio and video security, cloud computing security and 5G security. One disadvantage of compressive sensing is that it can not resist the chosen plaintext attack because of the lack of a diffusion mechanism.

In order to solve the above problems, including adaptiveness, energy and security issues in wireless body area networks, a flexible and secure data transmission system is proposed in this paper. The proposed scheme is composed of semi-tensor compressive sensing, hash function, Arnold scrambling, and chaotic scrambling (SC-HAC). Semi-tensor product compressive sensing (STP-CS) was proposed in [23], which demonstrated spark, coherence, and the restricted isometry property (RIP) theoretically. In our scheme, taking image data as an example, we analyze the transmission effect of data encryption in body area networks. The plain image conducts a sparse transformation, then uses Arnold scrambling and semi-tensor compressive sensing. The measurement matrix is generated by the chaotic system. Lastly, the chaotic permutation is used to get the cipher image. Our contributions are given as follows:

(1) We introduce the semi-tensor compressive sensing into wireless body area networks. Considering the diversity of signals, traditional methods need to design and store a large number of measurement matrices with different sizes. However, STP-CS overcomes the dimension restriction of matrix multiplication, and it can measure various sizes by using the same measurement matrix, reducing the number and size of measurement matrices drastically.

(2) The chaotic system has many good properties (i.e. pseudorandomness, ergodicity, and sensitivity to initial values), so it can be used to generate the semi-tensor measurement matrix. Moreover, the chaotic scrambling is used to improve the diffusion process, which enhancing the entire security in the proposed scheme.

The rest of this paper is organized as follows. In Section II, we introduce the related work. In Section III, we review basic knowledge, including compressive sensing, semi-tensor compressive sensing, Arnold scrambling, chaotic system, and several attacks. We propose the entire scheme and give the detailed process in Section IV. Section V presents experimental simulations under different conditions, and gives security analyses. The summary of this paper is given in Section VI.

II. Related work

There are a number of studies to solve the problem of energy consumption. Goudar et al. [10] used dielectric elastomers to analyze the minimized output of human energy harvesting. An energy-efficient MAC protocol was presented for WBANs and it was focused on healthcare applications [11]. In [12], a new algorithm was proposed for efficient routing in the network. The routing problem was considered as a multi-objective optimization problem. These methods dealt with the energy consumption problem by using energy harvesting or routing protocols, but did not refer to the security in WBANs. Compared with these methods, the proposed scheme reduces energy consumption by reducing data transmission size, and guarantees security in WBANs.

Some existing schemes can protect sensitive information. Ibaida et al. [13] proposed a wavelet-based steganography technique to protect patient confidential information. Abuadbba et al. [14] presented a walsh-hadamard-based 3-D steganography mechanism to protect sensitive data in point-of-care. Lee et al. [15] gave a secure and efficient key management scheme based on the elliptic curve to solve the security issue of body area networks. Wang et al. [16] presented a lightweight data storage strategy, which used secret sharing, erasing technique and algebraic signature. Compared with these methods, the proposed scheme not only protects sensitive information, but also reduces energy consumption and enhances adaptiveness.

Schemes based on compressive sensing have been studied in many articles. Zhang et al. [17] proposed compressive sensing of EEG signals to guarantee low energy consumption and cheap hardware. Imtiaz et al. [18] demonstrated that sensor nodes can extremely benefit from the use of CS. Liu et al. [19] discussed compressive sensing of multichannel EEG signals. A parallel image encryption scheme was proposed based on compressive sensing [21]. The scheme was composed of scrambling, mixing, S box and chaotic lattice XOR. Chen et al. [22] proposed a scheme of image encryption and compression based on the Kronecker CS and the elementary cellular

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

automaton of scrambling. Schemes based on compressive sensing realize encryption and compression simultaneously. The proposed scheme is based on STP-CS. STP-CS inherits the advantage of CS, and overcomes the dimension restriction of matrix multiplication.

III. Fundamental knowledge

In this section, we will introduce some fundamental knowledge about the proposed scheme SC-HAC, such as compressive sensing, semi-tensor compressive sensing, Arnold scrambling, chaotic system and several attacks.

A. Compressive Sensing

As a new research field of signal processing, compressive sensing [4]–[6] has attracted much attention. It can reduce the dimension of the signal and obtain a small number of observational values. The original sparse signal can be reconstructed from these observations. The basic sampling process is

$$y = \Phi x, \tag{1}$$

where $x \in R^N$, $\Phi \in R^{M\times N}$ and $y \in R^M$. It is essential to guarantee that x is the sparse signal (or to be sparse in some bases), y is the observation and Φ is the measurement matrix. If x can be represented in the sparse basis Ψ, then we have

$$x = \Psi\alpha, \tag{2}$$

where $\Psi \in R^{N\times N}$ and $\alpha \in R^N$. If there are K $(K \ll N)$ non-zero coefficients in α, which is said to be K sparse, then we get

$$y = \Phi\Psi\alpha = A\alpha, \tag{3}$$

where $A = \Phi\Psi$ is called the sensing matrix which satisfies the restricted isometry property (RIP). The RIP [24] is defined as

$$(1-\delta_k)\|\alpha\|_2 \le \|A\alpha\|_2 \le (1+\delta_k)\|\alpha\|_2, \tag{4}$$

where $\delta_k \in (0,1)$.

The measurement matrix can be a random matrix (Gaussian matrix, Bernoulli matrix, etc.), deterministic matrix (polynomial matrix, chaotic matrix, etc.), or structured random matrix (Toeplitz matrix, Hadamard matrix, etc.). The sparse bases can be discrete wavelet transform (DWT), discrete cosine transform (DCT), discrete fourier transform (DFT), and overcomplete dictionaries. The signal x is reconstructed from the observation y and the sensing matrix A, and this process can be described as

$$min\|\alpha\|_1 \quad s.t. \quad y = A\alpha. \tag{5}$$

In order to ensure the exact reconstruction of compressive sensing, M should satisfy the following relation

$$M \ge cKlog_2(N/K), \tag{6}$$

where c is a small constant.

The reconstruction algorithm of compressive sensing has many types, including matching pursuit (MP) [25], orthogonal matching pursuit (OMP) [26], regularized orthogonal matching pursuit (ROMP), basis pursuit algorithm (BP) [27], stagewise orthogonal matching pursuit (StOMP) [28] and CoSaMP [29].

B. Semi-tensor Compressive Sensing

The model of semi-tensor product compressive sensing (STP-CS) [23] is

$$y = A \ltimes x, \tag{7}$$

where $\ltimes$ represents the semi-tensor product [30]. $A \in R^{m\times n}$, $x \in R^p$, $y \in R^{mp/n}$ are the semi-tensor measurement matrix, the sparse signal (or to be sparse in some bases), and the observation, respectively, and $p = lcm(n,p)$. According to [23], [30], Equation (7) can be rewritten as

$$y = (A \otimes I_{p/n})x, \tag{8}$$

where $\otimes$ represents the tensor product. Equations (7-8) can be further rewritten as

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{\frac{mp}{n}} \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \ltimes \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}, \tag{9}$$

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{\frac{mp}{n}} \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & 0 & & a_{1n} & \cdots & 0 \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ 0 & \cdots & a_{11} & & 0 & \cdots & a_{1n} \\ & \vdots & & \ddots & & \vdots & \\ a_{m1} & \cdots & 0 & & a_{mn} & \cdots & 0 \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ 0 & \cdots & a_{m1} & & 0 & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}. \tag{10}$$

In Equation (10), the numbers of rows and columns in each block of the measurement matrix are both p/n, the total blocks are mn, the diagonal elements in the small block are a_{11}, a_{12}, $\cdots$, a_{mn}, and the remaining elements are 0. Equation (10) can also be expressed as follows

$$\begin{cases} y_1 = a_{11}x_1 + a_{12}x_{1+p/n} + \cdots + a_{1n}x_{p-2} \\ y_2 = a_{11}x_2 + a_{12}x_{2+p/n} + \cdots + a_{1n}x_{p-1} \\ \cdots \\ y_{\frac{mp}{n}} = a_{m1}x_3 + a_{m2}x_{3+p/n} + \cdots + a_{mn}x_p \end{cases}. \tag{11}$$

The reconstruction method of STP-CS is a parallel reconstruction method.

The semi-tensor measurement matrix A satisfies the following three theorems about spark, coherence, and the restricted isometry property.

Theorem 1 [23]. If $p = lcm(n,p)$, then spark$(A\otimes I_{p/n}) = spark(A)$.

Theorem 2 [23]. $\mu(A \otimes I_{p/n}) = \mu(A)$, where μ is a coherence coefficient.

Theorem 3 [23]. Suppose the matrix A satisfies the RIP with constant δ_k^A, then $\delta_k^{A\otimes I_{p/n}} = \delta_k^A$.

The semi-tensor product breaks the dimension restriction of matrix multiplication and realizes matrix multiplication in the case of $n \ne p$, so that the measurement matrix can be used to encrypt multiple signals with different dimensions.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

C. Arnold Scrambling

Arnold scrambling [31] is a commonly used algorithm for image scrambling. By changing the locations of pixels, the original image looks disorganized, and the mapping is one-to-one. The mapping of Arnold scrambling is given as

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 & k_2 \\ k_3 & k_2k_3+1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} (modN), \tag{12}$$

where (x_1, y_1) is the coordinate of the scrambled image, (x, y) is the coordinate of the original image, and N is the order of the image (the size of image is $N \times N$).

In Arnold scrambling, k_1 is the scrambling number, which controls the number of executions of Equation (12). k_1 doesn't appear in the equation. k_2 and k_3 are scrambling parameters chosen at random from positive integers that impact the output value.

D. Chaotic System

A chaotic system has the properties of pseudorandomness, ergodicity, and sensitivity to initial values, so it is often used in image encryption [32]–[34]. The Logistic system is presented as

$$x_{n+1} = ux_n(1-x_n), x_n \in (0,1), \tag{13}$$

where $x(n)$ is the Logistic chaotic sequence, x_n is the value of the nth iteration, x_{n+1} is the value of the $(n+1)$th iteration, and u is the control parameter, $u \in [3.57, 4]$.

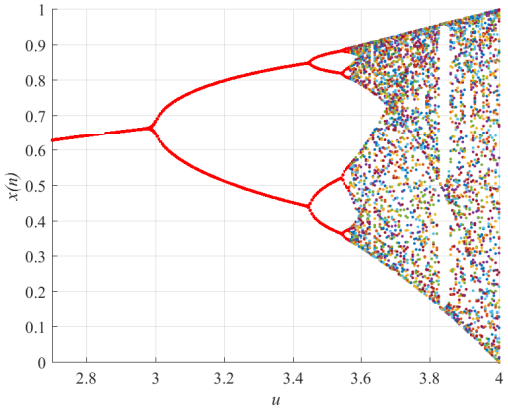


Fig. 2. Logistic bifurcation diagram [35].

Figure 2 is the bifurcation diagram of Logistic chaotic system. It shows the relationship between the final sequence $x(n)$ and the control parameter u. With the change of u, the distribution of $x(n)$ is divided into the periodic region (red section in Figure 2) and the chaotic region (colored section in Figure 2). Given a value of u in [0,3.57), $x(n)$ will converge to a stable periodic sequence when iterated on. Otherwise, the sequence fails to converge and appears chaotic.

The Tent system is given as follows

$$\begin{cases} z_{n+1} = z_n/k & 0 < z_n < k, \\ z_{n+1} = (1-z_n)/(1-k) & k < z_n < 1, \end{cases} \tag{14}$$

where $z(n)$ is the Tent chaotic sequence, z_n is the value of the nth iteration, z_{n+1} is the value of the $(n+1)$th iteration, and k is the control parameter, $k \in (0,1)$.

E. Threat Models

In this section, we introduce threat models [36] including the brute force attack, the statistical attack, the known plaintext attack and the chosen plaintext attack.

The brute force attack is an exhaustive search of the key to guess the correct key. This method is computationally expensive, and modern encryption algorithms are generally resistant to such attacks. For the statistical attack, the attacker explores the key by analyzing the statistical regularity of ciphertext and plaintext. The method of withstanding the statistical attacks tries to make the statistical characteristics of the ciphertext different from those of the plaintext. For the known plaintext attack, the attacker knows some plaintext and the corresponding ciphertext, thus the plaintext and the ciphertext are compared and analyzed. For the chosen plaintext attack, the attacker can choose some plaintext and get the corresponding ciphertext. By selecting special encrypted data for encryption, key information is obtained from the ciphertext for analysis.

IV. The Proposed Scheme

The details of our scheme (SC-HAC) are presented in this section. We give the key generation process, which uses the SHA-256 hash function. We introduce the encryption and compression process, which includes Arnold scrambling, semi-tensor compressive sensing, and chaotic scrambling. The decryption and decompression process is presented. We analyze the performance of our scheme.

A. Key Generation

The SHA-2 series includes the hash functions SHA-224, SHA-256, SHA-384, and SHA-512. Out of these, SHA-256 is the most widely used [37]. For the SHA-256 hash function, the length of the input is arbitrary, and the length of the output is 256 bits. In this paper, a plain image is used as the input, and the output of the hash function is treated as the key.

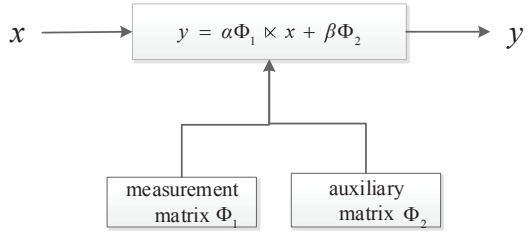


Fig. 3. The model of semi-tensor product compressive sensing.

Suppose the plain image is $P1$, and the SHA-256 hash function value of $P1$ is calculated to generate H with 256 bits. H is divided into three parts. We use the initial values n_0, a_0, b_0 and H to get k_1, k_2, k_3 for Arnold scrambling. The parameters n_0, a_0, b_0 are all positive integers. They are given by the sender. The parameter n_0 is used to generate Arnold scrambling number k_1. The parameters a_0 and b_0 are used to

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

generate Arnold scrambling parameters k_2 and k_3. The hash value H, and the keys k_1, k_2, k_3 are given as follows

$$\begin{aligned} H &= h_1, h_2, \cdots, h_{32}, \\ k_1 &= n_0 + h_1 \oplus h_2 \oplus \cdots \oplus h_{11}, \\ k_2 &= a_0 + h_{12} \oplus h_{13} \oplus \cdots \oplus h_{22}, \\ k_3 &= b_0 + h_{23} \oplus h_{24} \oplus \cdots \oplus h_{32}. \end{aligned} \tag{15}$$

When one pixel value of the plain image changes, the hash value becomes much different.

B. Encryption and Compression

The model of semi-tensor product compressive sensing (STP-CS) is given in Fig. 3, which shows the process of semi-tensor compressive sensing. In this model, the parameters α, β can be preset, Φ_1 is a chaotic measurement matrix with size $m \times n$, the compression ratio is $CR = m/n$, and the measurement matrix is generated by the Logistic chaotic sequence. The chaotic measurement matrix satisfies the restricted isometry property (RIP), which can guarantee exact recovery from compressive sensing [38]. In this paper, the initial value of Logistic system is k_4, and the parameter of Logistic system is 4. Then we sample the Logistic sequence. The initial sampling position is arbitrary. For convenience, it is set as 1. In order to reduce the correlation of the Logistic sequence, the sampling interval is set as 4 empirically. So the sequence s can be obtained. We can get the sequence t from the transformation $t = 1 - 2s$. The sequence t is reshaped to get the measurement matrix Φ_1, which can be expressed as follows

$$\Phi_1 = \sqrt{\frac{2}{m}} \begin{bmatrix} t_0 & \cdots & t_{m(n-1)} \\ t_1 & \cdots & t_{m(n-1)+1} \\ \vdots & \vdots & \vdots \\ t_{m-1} & t_{2m-1} & t_{mn-1} \end{bmatrix}, \tag{16}$$

where the coefficient $\sqrt{\frac{2}{m}}$ is used for the normalization. Φ_2 is the auxiliary matrix with size $mp/n \times q$, and it is generated by Tent chaotic system. The parameter of Tent system is 0.3, and the initial value of Tent system is k_5. We discard the former part of the generated sequence and sample the latter part. The initial sampling position is 1, and the sampling interval is 4. The sampling sequence is reshaped by the column priority to obtain the auxiliary matrix Φ_2.

Figure 4 shows the whole flow chart of our encryption and decryption algorithms in SC-HAC. The encryption process is presented as follows:

Step 1: According to the plain image $P1$, the keys k_1, k_2 and k_3 can be calculated.

Step 2: The plain image is $P1$, whose size is $p \times q$. The discrete wavelet transform (DWT) is performed to get $P2$, whose size is still the same as that of $P1$. DWT can only be performed on a square matrix.

Step 3: Arnold scrambling is performed on $P2$ to get $P3$ whose size is $p \times q$. The parameters are k_1, k_2, k_3, where k_1 is Arnold scrambling number, and k_2 and k_3 are Arnold scrambling parameters.

Step 4: $P3$ is considered as x. Performing a semi-tensor compressive sensing measurement, y can be obtained. y is denoted as $P4$ whose size is $mp/n \times q$, and $P4$ is the result of compression and encryption. In the model of $y = \alpha\Phi_1 \ltimes x + \beta\Phi_2$, when $n \neq p$, Φ_1, x can be calculated because we use semi-tensor compressive sensing. We take $n < p$ in the simulation experiments.

Step 5: Logistic chaotic scrambling is an application of Logistic chaotic system. First, we get the Logistic chaotic sequence w according to specific chaotic parameters. Then, we put the sequence w in ascending order to get the sequence v. The position of each element in the sequence v appearing in the original sequence w is denoted as the index sequence θ. That is, $\theta(j) = i$ for $v(j) = w(i)$, where i, j are integers and $1 \leq i, j \leq mp/n \times q$. Finally, according to the index sequence θ, we scramble $P4$ to get the final cipher image $P5$ with size $mp/n \times q$. That is, $P5(k') = P4(\theta(k'))$, where k' $(1 \leq k' \leq mp/n \times q)$ is the index. In the generation process of Logistic chaotic sequence, the control parameter is 4, and the chaotic initial value is k_6. The initial sampling position is 1, and the sampling interval is 4.

C. Decryption and Decompression

Decryption is the inverse process of encryption. In Fig. 4, the sender should transmit some data to the receiver for the decryption. They are k_1 (Arnold scrambling number), k_2 (Arnold scrambling parameter), k_3 (Arnold scrambling parameter), k_4 (the initial values of the measurement matrix), k_5 (the initial values of the auxiliary matrix), k_6 (the initial values of the scrambling sequence) and the cipher image $P5$. Compared with [39], which transmits the entire semi-tensor measurement matrix and the cipher image. Our scheme transmits only a few constants, which greatly reduces the transmission overhead.

The key transmission is an important part of any security scheme. A key distribution scheme was designed by using Elliptic Curve Cryptography in Wireless Sensor Networks [40]. Seo et al. [41] proposed a certificateless-effective key management protocol in dynamic Wireless Sensor Networks. As a supplement of original scheme, the keys can be transmitted in our scheme by digital envelope technology [42]–[44]. These digital envelopes employ RSA (Rivest Shamir Adleman) and ECC (Elliptic Curve Cryptography) asymmetric key technique to encrypt and decrypt the secret key.

The decryption process is presented as follows. Firstly, we use the key k_6 to generate the Logistic chaotic scrambling matrix. Secondly, the cipher image is multiplied by the inverse matrix of the scrambling matrix. Thirdly, we perform semi-tensor compressive sensing reconstruction algorithm by using OMP algorithm. Finally, we can restore the original image by Arnold recovery and the inverse discrete wavelet transform (IDWT). Considering snooping data, key sensitivity will be discussed in Section V-D. The proposed algorithm has the sensitivity to the decryption process. It means that the scheme prevents adversaries from decrypting the transmitted data with a key similar to the encryption key.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

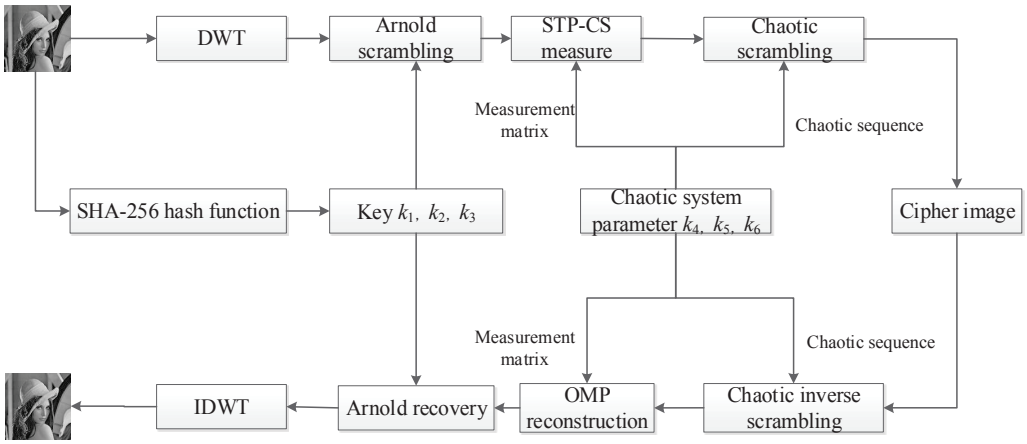


Fig. 4. The block diagram of the proposed scheme SC-HAC. The upper layer denotes the encryption and compression process, including SHA-256 hash function, Arnold scrambling, semi-tensor compressive sensing and chaotic scrambling. The middle layer denotes the transmission parameters. The lower layer denotes the decryption and decompression process.

D. Discussion

There are some advantages in the proposed scheme SC-HAC. The SHA-256 hash function is associated with the plain image. Hash values differ even if the original image is changed one pixel, so our scheme can resist the chosen plaintext attack and the known plaintext attack, which can compensate for the disadvantage of compressive sensing. Since compressive sensing is a linear transform, the encryption schemes based on CS struggle to resist the chosen/known plaintext attacks. By using semi-tensor product, the image encryption has brought new ideas. Because sensors collect data with different sizes, our scheme realizes multiple encryptions with a measurement matrix and breaks the dimension restriction of matrix multiplication, which can embody the adaptation and the flexibility of the proposed scheme. The chaotic system can be used to generate the semi-tensor measurement matrix, and its storage and transmission only require a few constants, which greatly reduces the amount of data in the network and improves the efficiency. The semi-tensor measurement matrix is smaller than the ordinary measurement matrix, so the computation overhead is relatively small.

V. Evaluation and Analyses

In this section, we conduct a series of simulation experiments. Experimental conditions include the simulation platform MATLAB R2016a, CPU 3.40GHz, memory 8GB, and 64-bit Microsoft windows 7. We set the initial values as $n_0 = 10$, $a_0 = 3$, $b_0 = 5$. Parameters n_0, a_0, b_0 are all positive integers. In the chaotic system, the initial values of the measurement matrix, the auxiliary matrix and the scrambling sequence are 0.2, 0.23 and 0.3, respectively. The initial values of the chaotic system should be in the range of $(0, 1)$. Three different initial values are to generate three different sequences. The reconstruction algorithm is the OMP algorithm.

A. Adaptiveness and Energy Saving

In wireless body area networks, the sizes of images are different. Traditional methods design a specific measurement matrix for a specific size, while our approach can measure various sizes by using the same measurement matrix. The experiment is designed for the dimension mismatching of the matrix, which shows that semi-tensor compressive sensing breaks the dimension restriction of matrix multiplication. Even when $n \neq p$, Φ_1 and x can also be multiplied. In the semi-tensor measurement $y = \alpha\Phi_1 \ltimes x + \beta\Phi_2$, according to the empirical value, we select the value of α as 0.001 and we select the value of β as 254. In the first row of Table I, the dimension of the semi-tensor measurement matrix is fixed to be 32×64, and the compression ratio is 0.5. In the second row of Table I, the dimension of the semi-tensor measurement matrix is selected as 64×128, and the compression ratio is 0.5.

TABLE I
THE QUALITY OF DECRYPTED IMAGES WITH DIFFERENT SIZES, AND THE SIZE OF THE MEASUREMENT MATRIX IS 32×64 (THE FIRST ROW IN CELLS), 64×128 (THE SECOND ROW IN CELLS). LARGER VALUES ARE BETTER HERE. (UNIT: DB)

image \ size	128×128	256×256	512×512	1024×1024
Lena	23.8371 23.8675	29.3458 29.5728	33.3173 33.6057	37.9660 38.2389
Pepper	23.6585 23.7683	29.9231 30.1148	32.5997 32.5481	38.2178 38.4805
House	26.5482 28.3438	32.1845 33.9554	37.8491 37.9960	40.6465 41.4877
Cameraman	23.4787 23.2923	28.0603 28.0443	32.4532 32.7850	37.4560 37.8333

Peak signal to noise ratio (PSNR) is an index used to evaluate the image quality. The greater the value of PSNR is, the better the image quality is. The smaller the mean square error (MSE) is, the smaller the difference between the original image and the reconstructed image is. That is

$$PSNR = 10\lg\left[\frac{M_1 \times N_1 \times 255^2}{\sum_{i=0}^{M_1-1}\sum_{j=0}^{N_1-1}(P(i,j)-Q(i,j))^2}\right], \tag{17}$$

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

$$MSE = \frac{1}{M_1 \times N_1} \sum_{i=0}^{M_1-1} \sum_{j=0}^{N_1-1} (P(i,j) - Q(i,j))^2, \quad (18)$$

where P and Q represent the original image and the cipher image, respectively, and M_1 and N_1 are the length and the width of the image. The relationship between PSNR and MSE is

$$PSNR = 10 \lg \frac{255^2}{MSE}. \quad (19)$$

Lena, Pepper, House and Cameraman images are tested, whose sizes are 128×128, 256×256, 512×512, 1024×1024, respectively. The column number of the semi-tensor measurement matrix is different from the row number of the image, thus we use the semi-tensor compressive sensing. As illustrated in Table I, for the same original image, the larger the size of the original image is, the higher the PSNR of the reconstructed image is. This is because when the size of the measurement matrix is fixed, the larger the image size is, the more the samples are. We have done numerical experiments when the sizes of measurement matrices are 16×32 and 8×16. The results of 16×32 are similar with those of Table I. The PSNR of 8×16 measurement matrix is 44.08 percent lower than that of 64×128 for 128×128 Lena. We can see that there is no significant difference between the first row of Table I and the second row of Table I with regard to PSNR of the decrypted image. Therefore, the proposed algorithm is suitable for the gray images with different sizes.

Next, we will discuss the energy consumption of our scheme. The energy consumption is dominated by wireless communication, and the communication energy is proportional to the volume of the data stream. Therefore, the energy model is given as follows [45]:

$$E = C \times M \times b, \quad (20)$$

where M is the sampling number, b is the bit resolution, and C is the energy per bit. The larger the M, the more energy consumption in WBANs. According to [45], we set $C = 3$ nJ/bit, $b = 16$. So we can calculate energy under different conditions. From Table II, we can see that different sampling rates cause different energy consumption. When $CR = 0.4$, the energy consumption is 0.315mJ $(1mJ = 1 \times 10^{-3}J)$, 1.258mJ, 5.033mJ, 20.133mJ with sizes 128×128, 256×256, 512×512, 1024×1024, respectively. But for the original data, the energy consumption is 0.786mJ, 3.146mJ, 12.583mJ, 50.332mJ, respectively. The case of $CR = 0.4$ saves 0.471mJ, 1.888mJ, 7.550mJ, 30.199mJ energy relative to the original data with sizes 128×128, 256×256, 512×512, 1024×1024. It's obvious that the former has more advantage. The higher the sampling number is, the more energy consumption is. Therefore, we can save energy by reducing the sampling number.

For storage costs, we select the original image with size 1024×1024. Other conditions remain unchanged, and the compression ratio is set to be 0.5. The dimension of the measurement matrix is 512×1024 for the traditional CS, which indicates that we should store 524288 bytes. Meanwhile, in our scheme, we choose the 32×64-dimensional measurement matrix which has 2048 bytes. Accordingly, one is in a level of hundreds of thousands, and another is in a level of thousands. The bigger the original image is, the greater the difference becomes. Thus, it is obvious that we are more competitive in this aspect. On the one hand, during the transmission, we only transmit a few parameters rather than many data, which increasingly reduces the storage and transmission overhead. On the other hand, the measurement matrix has smaller size, which saves the computation time in the generation process.

TABLE II
ENERGY CONSUMPTION UNDER DIFFERENT CONDITIONS (UNIT: MJ)

CR \ size	128×128	256×256	512×512	1024×1024
original data	0.786	3.146	12.583	50.332
CR=0.8	0.629	2.517	10.066	40.265
CR=0.6	0.472	1.887	7.550	30.199
CR=0.4	0.315	1.258	5.033	20.133

Brief summary: Our approach can measure signals with various sizes by using the same measurement matrix, and we give examples of 32×64 and 64×128 measurement matrices. The case of $CR = 0.4$ saves 0.471mJ, 1.888mJ, 7.550mJ, 30.199mJ energy relative to the original data with sizes 128×128, 256×256, 512×512, 1024×1024, respectively. Traditional compressive sensing method stores 524288 bytes, while the proposed scheme only stores 2048 bytes.

B. Recovery Effect

In this section, we discuss the recovery effect in our scheme. Figure 5 shows the simulation results of different plain images, including Lena (256×256), Pepper (256×256), House (256×256), Cameraman (256×256). The compression ratio is 0.5, and the dimension of the semi-tensor measurement matrix is 64×128. Figures 5(a1)-(d1), (a2)-(d2), (a3)-(d3) are the simulation results of original images, encrypted images and decrypted images, respectively. Figures 5(a4)-(d4) are the difference maps between the original images and the decrypted images. Figure 5(a4) is the difference map between Fig. 5(a1) and Fig. 5(a3). Figure 5(b4) is the difference map between Fig. 5(b1) and Fig. 5(b3). Figure 5(c4) is the difference map between Fig. 5(c1) and Fig. 5(c3). Figure 5(d4) is the difference map between Fig. 5(d1) and Fig. 5(d3). These maps show the amount of error per pixel between the two images. MSE values in Fig. 5 are 71.7468, 63.3282, 26.1540, 102.0122, respectively.

In Figures 5(a2)-(d2), encrypted images are similar to noise or texture, and useful information has been hidden in the image. The experimental results in Fig. 5 show that the PSNR of the reconstructed image is 29.5728dB for Lena (256×256), the PSNR of the reconstructed image is 30.1148dB for Pepper (256×256), the PSNR of the reconstructed image is 33.9554dB for House (256×256), and the PSNR of the reconstructed image is 28.0443dB for Cameraman (256×256). In terms of the above experimental results of four different plain images, the quality of the reconstructed image is good when the compression ratio is 0.5.

We analyze the relationship between PSNR and CR in SC-HAC. PSNR is peak signal to noise ratio, and CR is the

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

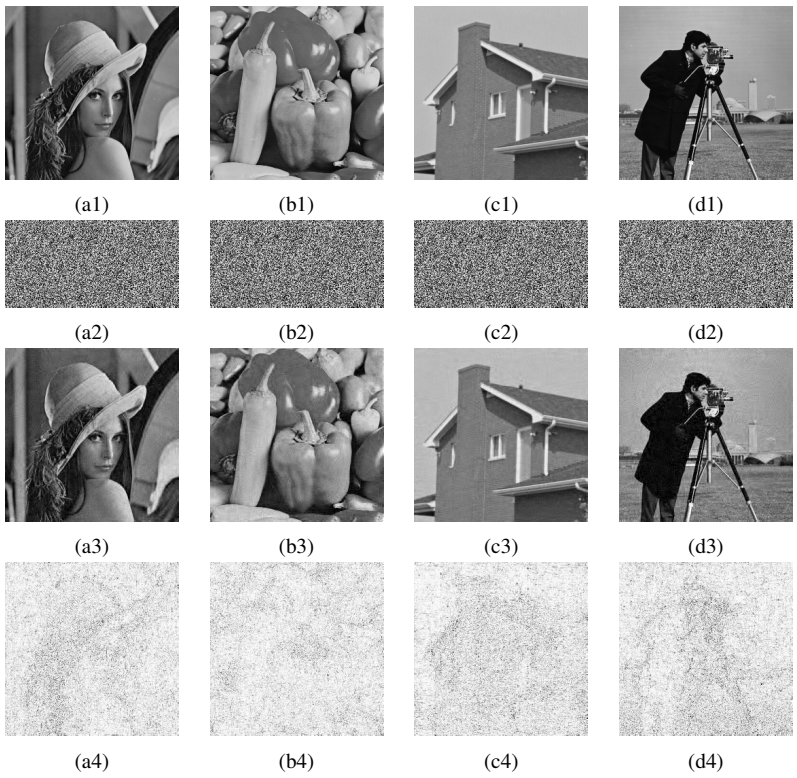


Fig. 5. Encryption and decryption images for Lena (256×256), Pepper (256×256), House (256×256), Cameraman (256×256). $(a1)-(d1)$, $(a2)-(d2)$ are the simulation results of original images, encrypted images respectively. (a3) The recovery image for Lena (256×256), PSNR=29.5728dB. (b3) The recovery image for Pepper (256×256), PSNR=30.1148dB. (c3) The recovery image for House (256×256), PSNR=33.9554dB. (d3) The recovery image for Cameraman (256×256), PSNR=28.0443dB. (a4) The difference map between (a1) and (a3), MSE=71.7468. (b4) The difference map between (b1) and (b3), MSE=63.3282. (c4) The difference map between (c1) and (c3), MSE=26.1540. (d4) The difference map between (d1) and (d3), MSE=102.0122.

compression ratio. In Fig. 6, different plain images are Lena, Pepper, House and Cameraman with size 256×256, and the horizontal axis is the compression ratio (CR) which varies from 0.3 to 1. Figure 6(a) represents that the column number of the semi-tensor measurement matrix is 128, so the row number is $128 \times CR$. Figure 6(b) represents that the column number of the semi-tensor measurement matrix is 64, so the row number is $64 \times CR$. For example, when $CR = 0.5$, the dimension of the measurement matrix is 64×128 in Fig. 6(a), and the dimension of the measurement matrix is 32×64 in Fig. 6(b). And the latter is more smaller than the former. The sizes of STP-CS measurement matrices are set as a quarter and one-sixteenth of that of the traditional CS, respectively.

In Fig. 6(a), the PSNR of House image is the highest, and the PSNR of Lena image is similar with that of Pepper image. When $CR = 0.8$, Lena and Pepper images have the same PSNR values. When $CR < 0.8$, Pepper image has more advantages. When $CR > 0.8$, the result is just opposite. When $CR = 1$, it means that there is no compression for the plain image, and the PSNRs of Lena, Pepper, House and Cameraman images are 36.1777dB, 35.8975dB, 38.1995dB and 35.4446dB, respectively. We can see that the images without compression are better than those with compression for the quality of the decrypted image. In Fig. 6(b), when $CR = 0.3$, the PSNR of House image is relatively low. However, the effects of other images are also available. From Fig. 6, we can draw a conclusion about different images. That is, with the increment of the compression ratio for the semi-tensor measurement matrix, the more the number of the samples is, the better the quality of the decrypted image is. For different plain images, the qualities of the corresponding decrypted images are also different.

Figure 7 shows the relationship between PSNR and CR for different sparse bases, such as DWT, DCT and DFT. The size of Pepper is 512×512. Figure 7(a) represents that the column number of the semi-tensor measurement matrix is 128, and its row number is $128 \times CR$. Figure 7(b) represents that the column number of the semi-tensor measurement matrix is 64, and its row number is $64 \times CR$. With the increment of the compression ratio, the number of samples increases, and the quality of the decrypted image becomes better. In Fig. 7(a), when CR ranges from 0.3 to 1, the quality of DWT reconstruction ranges from 29.4395dB to 35.4153dB, the quality of DCT reconstruction ranges from 25.4857dB to 34.5215dB, and the quality of DFT reconstruction ranges from 25.6729dB to 51.7959dB. The trends of these three curves are roughly the same as those in Fig. 7(b). On the whole, the DWT is better than the other two sparse bases. The effect of DFT is better when the compression ratio is higher. When $CR < 0.9$, the decryption quality of DWT is the highest in three sparse bases, and its effect is the best. Therefore, in other experiments, the sparse basis can be selected as DWT. In this experiment, the sizes of STP-CS measurement matrices are set as one-sixteenth and one-sixty-fourth of that of the traditional

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

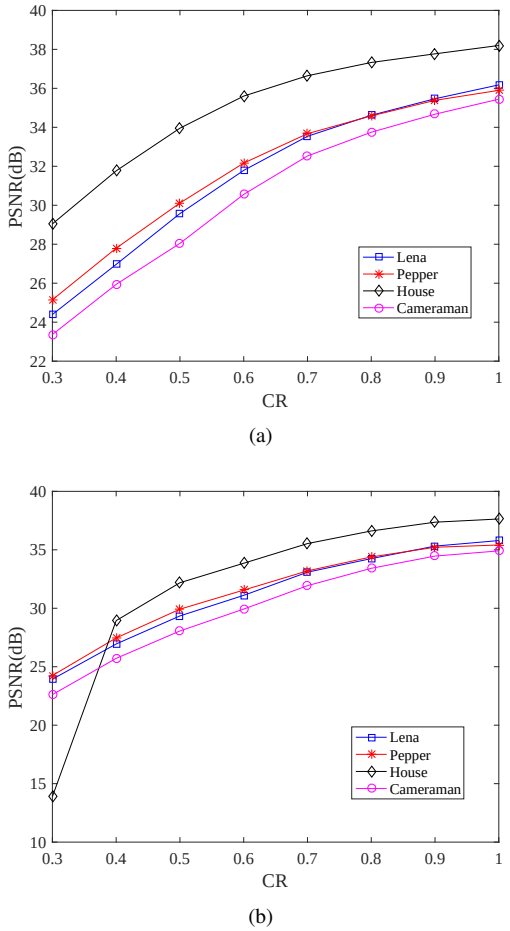


Fig. 6. The relationships between PSNR and CR for different plain images, where PSNR is peak signal to noise ratio, and CR is the compression ratio. The plain images are Lena, Pepper, House and Cameraman with size 256×256. The compression ratio varies from 0.3 to 1. (a) The column number of the semi-tensor measurement matrix is 128, and the row number is $128 \times CR$. (b) The column number of the semi-tensor measurement matrix is 64, and the row number is $64 \times CR$.

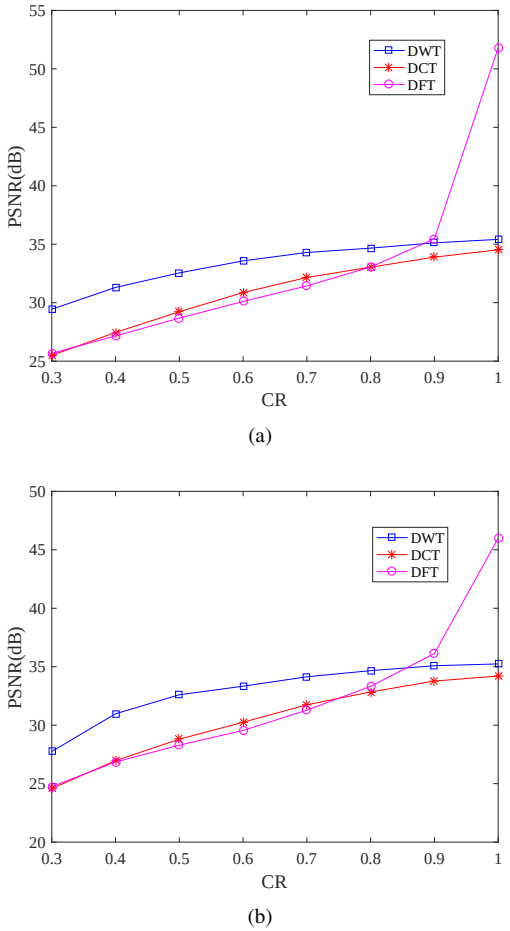


Fig. 7. The relationships between PSNR and CR for different sparse bases. The plain image is Pepper (512×512), and the sparse bases are DWT, DCT and DFT, respectively. The compression ratio varies from 0.3 to 1. (a) The column number of the semi-tensor measurement matrix is 128, and its row number is $128 \times CR$. (b) The column number of the semi-tensor measurement matrix is 64, and its row number is $64 \times CR$.

CS, respectively.

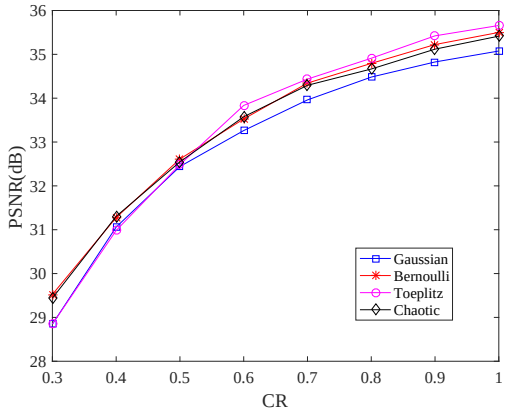
Pepper (512×512) is used in Fig. 8, where the horizontal axis is CR. Four measurement matrices (i.e. Gaussian, Bernoulli, Toeplitz and chaotic matrices) have influence on the quality of the decrypted image. Figure 8(a) represents that the column number of the semi-tensor measurement matrix is 128, and its row number is $128 \times CR$. Figure 8(b) represents that the column number of the semi-tensor measurement matrix is 64, and its row number is $64 \times CR$. From Fig. 8, we can see that the effects of four kinds of measurement matrices are probably similar, and the effect of Toeplitz matrix is slightly better. In general, the higher the compression ratio is, the more the number of samples is, and the better the image quality becomes. If the quality of the decrypted image is required to be 30dB, then CR is more than 0.4 for these four measurement matrices. In Fig. 8(a), when the compression ratio of the semi-tensor measurement matrix is 0.4, the quality of the decrypted image for Chaotic matrix is 31.3045dB, and the PSNRs of Gaussian, Bernoulli, Toeplitz measurement matrices are 31.0604dB, 31.2786dB, and 30.9829dB, respectively. In Fig. 8(b), when $CR = 0.3$, PSNR is relatively low for Gaussian measurement matrix. However, the results of other measurement matrices are good. The sizes of STP-CS measurement matrices are set as one-sixteenth and one-sixty-fourth of that of the traditional CS, respectively, which can show that semi-tensor compressive sensing reduces the transmission bandwidth and the storage space, and ensures the high quality of the decrypted image.

In Fig. 9, the plain image is Pepper (512×512), and the column number of the semi-tensor measurement matrix is 256. Figures $9(a)$, (b), (c) represent the cipher images with different compression ratios 0.25, 0.5 and 0.75, and Figures $9(d)$, (e), (f) represent the decrypted images with compression ratios 0.25, 0.5, 0.75. As the number of samples increases, the image quality is getting better and better. PSNRs of Figs. $9(d)$, (e) and (f) are 28.3421dB, 32.5067dB and 34.5396dB, respectively. Figures $9(g)$, (h), (i) show the difference maps between the original images and the decrypted images. MSE values of Figs. $9(g)$, (h) and (i) are 95.2518, 36.5100, 22.8621, respectively.

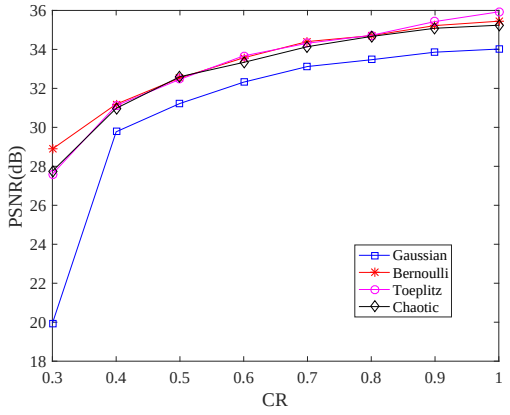
Brief summary: Figure 5 shows the difference map, PSNR

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

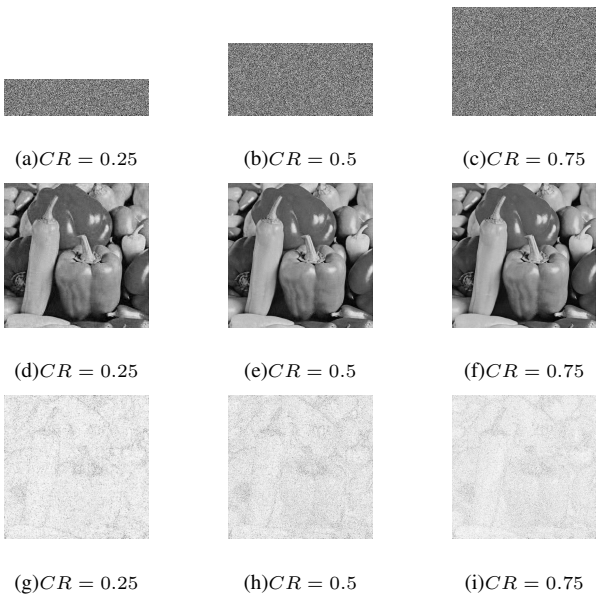


(a)



(b)

Fig. 8. The relationships between PSNR and CR for different measurement matrices, and the plain image is Pepper (512×512). Four measurement matrices are Gaussian, Bernoulli, Toeplitz, and chaotic matrices, respectively. The compression ratio varies from 0.3 to 1. (a) The column number of the semi-tensor measurement matrix is 128, and its row number is $128 \times CR$. (b) The column number of the semi-tensor measurement matrix is 64, and its row number is $64 \times CR$.



(a)$CR = 0.25$ (b)$CR = 0.5$ (c)$CR = 0.75$

(d)$CR = 0.25$ (e)$CR = 0.5$ (f)$CR = 0.75$

(g)$CR = 0.25$ (h)$CR = 0.5$ (i)$CR = 0.75$

Fig. 9. The original image is Pepper (512×512). (a)-(c) The encryption images. (d)-(f) The decryption images. (g)-(i) The difference maps.

and MSE between the original image and the recovery image. For House image, the PSNR can reach up to 33.9554dB. In Figs. 6-8, we analyze the relationship between PSNR and CR for different plain images, sparse bases and measurement matrices. On the whole, with the increment of the compression ratio for the measurement matrix, the higher the number of the samples is, the better the quality of the decrypted image is. For different plain images, the quality of the decrypted image is different. In Figure 6, Lena and Peppers images have similar PSNR, the PSNR of House is highest, and the PSNR of Cameraman is lowest. DWT is better than other sparse bases in the quality of the decrypted image. Four measurement matrices (i.e. Gaussian, Bernoulli, Toeplitz and chaotic matrices) have similar PSNR for the decrypted image. Figure 9 shows the recovery effect under different compression ratios. The higher the compression ratio is, the smaller MSE of the decrypted image is. When CR is 0.75, MSE is 22.8621 for Peppers image.

C. The Effect of Image Encryption

In this section, we analyze adjacent pixel correlation, histogram, information entropy and runtime. This paper randomly chooses 3000 adjacent pixels from the original image and the cipher image, including horizontal, vertical and diagonal directions. Mean, variance, covariance and correlation coefficient are given as follows

$$E(x) = \frac{1}{N_2}\sum_{i=1}^{N_2} x_i, \tag{21}$$

$$D(x) = \frac{1}{N_2}\sum_{i=1}^{N_2}(x_i - E(x_i))^2, \tag{22}$$

$$cov(x,y) = \frac{1}{N_2}\sum_{i=1}^{N_2}(x_i - E(x_i))(y_i - E(y_i)), \tag{23}$$

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{24}$$

where x and y represent the gray values of two adjacent pixels in three directions, and N_2 represents the number of pixels.

Figures 10-12 respectively represent horizontal, vertical and diagonal adjacent pixel correlations. The plain image is Lena (512×512). Figure 10(a), Figure 11(a), Figure 12(a) are adjacent pixel correlations of the original image in three directions. Figure 10(b), Figure 11(b), Figure 12(b) are results of the cipher images, and the dimensions of their measurement matrices are 256×512. Figure 10(c), Figure 11(c), Figure 12(c) are results of the cipher images, and the dimensions of their measurement matrices are 128×256. Figure 10(d), Figure 11(d), Figure 12(d) are results of the cipher images with measurement matrices 64×128. As is illustrated in Figs. 10-12, three directions of the original image have strong correlations, and linear correlations generally. Semi-tensor compressive sensing is uniform dispersion, and their cipher images can not be used to find the relevant information. The situation shows that different sizes of the semi-tensor measurement matrices

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

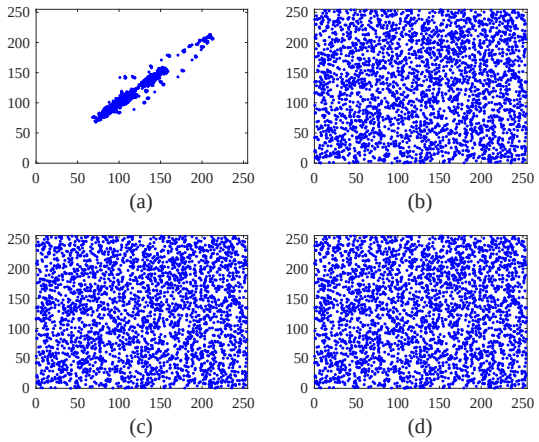


Fig. 10. Horizontal adjacent pixel correlation. (a) Lena (512×512) plain image. (b) (c) (d) the cipher images (the dimensions of the semi-tensor measurement matrices are 256×512, 128×256, and 64×128, respectively).

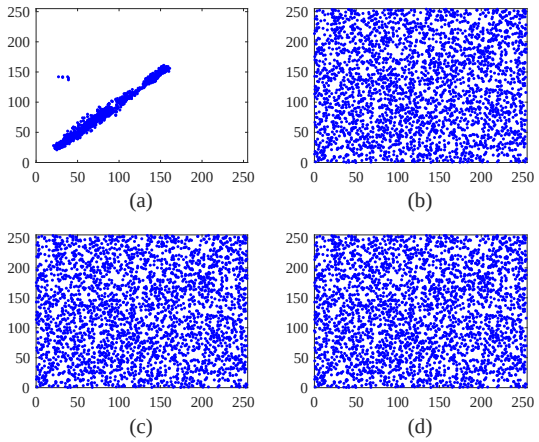


Fig. 11. Vertical adjacent pixel correlation. (a) Lena (512×512) plain image. (b) (c) (d) the cipher images (the dimensions of the semi-tensor measurement matrices are 256×512, 128×256, and 64×128, respectively).

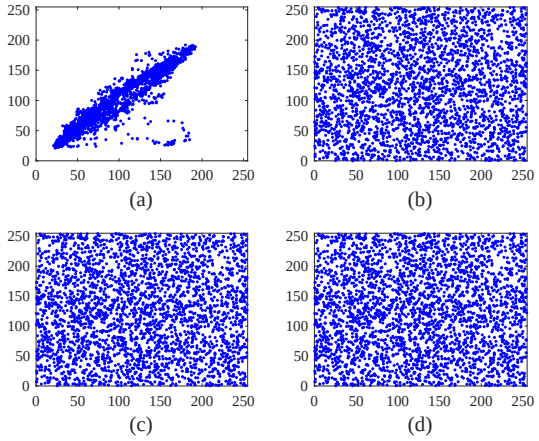


Fig. 12. Diagonal adjacent pixel correlation. (a) Lena (512×512) plain image. (b) (c) (d) the cipher images (the dimensions of the semi-tensor measurement matrices are 256×512, 128×256, and 64×128, respectively).

TABLE III
ADJACENT PIXEL CORRELATION WITH k_5=0.23

direction \ type	original image	Φ_1	Φ_2	Φ_3
horizontal	0.9839	-0.0035	-0.0036	-0.0035
vertical	0.9928	-0.0036	-0.0036	-0.0036
diagonal	0.9752	0.0006	0.0006	0.0006

TABLE IV
ADJACENT PIXEL CORRELATION WITH k_5=0.24

direction \ type	original image	Φ_1	Φ_2	Φ_3
horizontal	0.9839	-0.0028	-0.0028	-0.0029
vertical	0.9928	0.0004	0.0004	0.0004
diagonal	0.9752	-0.0009	-0.0009	-0.0009

all get better encryption effects. That is to say, there is little correlation between the adjacent pixels of the cipher images.

In Table III, the plain image is Lena (512×512), and Φ_1, Φ_2, Φ_3 represent the measurement matrices with dimensions 256×512, 128×256, and 64×128, respectively. It can be seen from Table III that the adjacent pixel correlations of the plain image are larger than 0.97, which is near 1, and this means strong correlation. The adjacent pixel correlations of cipher images are small under horizontal, vertical and diagonal directions, and they are less than 0.004 (to be close to 0), which indicates that the correlation is weak. The eavesdropper can not get useful information from the cipher image. We can see that the values in the same direction are similar, and they are affected by the auxiliary matrix. In Table III, the chaotic initial value k_5 of the auxiliary matrix is selected as 0.23. If this value is set to be 0.24, then Table IV is obtained. It's obvious that the values in the same direction have changed. Table V uses Lena (512×512), Pepper (512×512), House (512×512) and Cameraman (512×512), and the dimension of the semi-tensor measurement matrix is 64×128. Compared with the traditional compressive sensing, it is clear that our method has advantages in adjacent pixel correlation.

TABLE V
COMPARISON OF ADJACENT PIXEL CORRELATION

image \ type	direction	original image	traditional CS	SC-HAC
Lena	horizontal	0.9839	0.7655	-0.0035
	vertical	0.9928	-0.0452	-0.0036
	diagonal	0.9752	-0.0395	0.0006
Pepper	horizontal	0.9733	0.8164	-0.0035
	vertical	0.9764	-0.0137	-0.0036
	diagonal	0.9557	-0.0143	0.0006
House	horizontal	0.9867	0.8967	-0.0036
	vertical	0.9793	0.0013	-0.0036
	diagonal	0.9669	0.8796	-0.0036
Cameraman	horizontal	0.9666	0.8796	-0.0036
	vertical	0.9783	0.0530	-0.0036
	diagonal	0.9490	0.0421	0.0006

A histogram reflects the statistical characteristics of an image, representing the proportions of different gray values. Figure 13(a) is the histogram of Lena (512×512), and Figures 13(b)-(d) are the results of the semi-tensor measurement matrices with dimensions 256×512, 128×256, and 64×128, respectively. As can be seen from Fig. 13, the

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

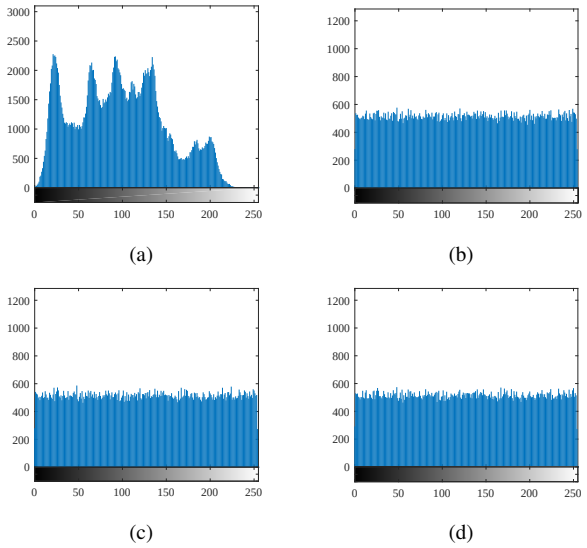


Fig. 13. Histogram. (a) Lena (512×512) is the plain image, (b) (c) (d) the cipher images (the dimensions of the semi-tensor measurement matrices are 256×512, 128×256, and 64×128, respectively).

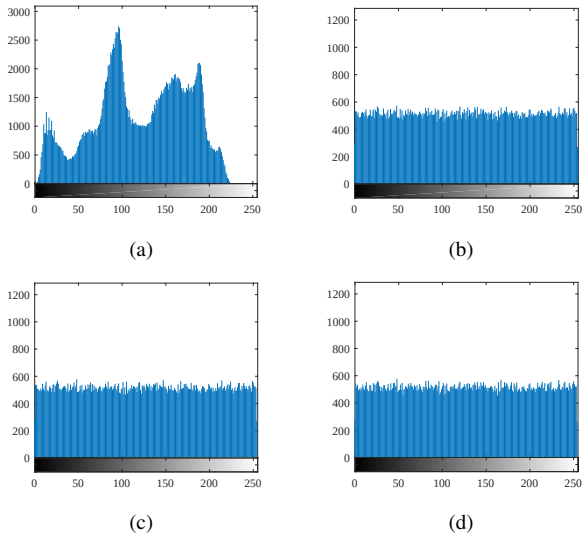


Fig. 14. Histogram. (a) Pepper (512×512) is the plain image, (b) (c) (d) the cipher images (the dimensions of the semi-tensor measurement matrices are 256×512, 128×256, and 64×128, respectively).

statistical features of plain images are obvious and uneven, the distribution of the encrypted images is more uniform, and the statistical feature is small. Figure 14 is the histogram of Pepper image. Since the histograms of House and Cameraman images are just like those of Lena and Pepper images, we do not list them here. In Figure 15, the dimensions of the semi-tensor measurement matrices are 1×2, 2×4, 4×8, and 8×16, respectively. Pepper (512×512) is the plain image. From Figure 15, we can see that the distribution is uniform. Even if the measurement matrix is small, the encryption effect is good. There are two reasons. The semi-tensor compressive sensing can encrypt image itself. Moreover, the chaotic scrambling improves the encryption effect.

Information entropy is an indicator used to reflect the

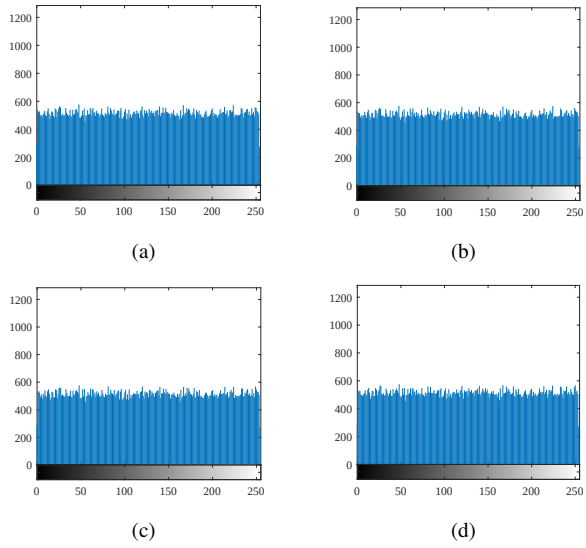


Fig. 15. Histogram. Pepper (512×512) is the plain image, (a) (b) (c) (d) the cipher images (the dimensions of the semi-tensor measurement matrices are 1×2, 2×4, 4×8, and 8×16, respectively).

TABLE VI
INFORMATION ENTROPY OF DIFFERENT MEASUREMENT MATRICES

type / image	original image	Φ_1	Φ_2	Φ_3
Lena	7.5736	7.9857	7.9856	7.9857
Pepper	7.5936	7.9856	7.9857	7.9858
House	6.4971	7.9860	7.9858	7.9859
Cameraman	6.9719	7.9856	7.9857	7.9859

uncertainty of information. Let G be an image with gray level L, x_i represents the pixel number of the ith gray level, and $p(x_i)$ indicates the probability value of the ith gray level. Moreover, we have $\sum_{i=1}^{L} p(x_i) = 1$, and the information entropy of G is expressed as

$$H(G) = -\sum_{i=1}^{L} p(x_i) \log_2 p(x_i). \tag{25}$$

Lena (512×512), Pepper (512×512), House (512×512), Cameraman (512×512) are chosen as the original images, and the sizes of the semi-tensor measurement matrices are $\Phi_1(256\times512)$, $\Phi_2(128\times256)$ and $\Phi_3(64\times128)$. Their values of information entropy are given in Table VI. The theoretical value of the information entropy is 8 for the encrypted image with 256 gray levels. Therefore, the closer the information entropy is to 8, the higher the security of the encrypted image is. We can see that the entropy of the encrypted image is improved. Compared with the original image, the entropy is closer to 8.

Table VII shows the performance of the algorithm with respect to the runtime. Pepper images are used with sizes 128×128, 256×256, 512×512, and 1024×1024, respectively. The dimensions of the semi-tensor measurement matrices are 32×64, 64×128, 128×256, and 256×512, respectively. From Table VII, we can see that the decryption time is more than the encryption time. This is because OMP reconstruction takes much time. As the image size increases, the encryption and

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

TABLE VII
ENCRYPTION AND DECRYPTION TIME FOR DIFFERENT SIZE IMAGES (UNIT:S)

size \ runtime	encryption time	decryption time	total time
128×128	0.3734	1.5385	1.9119
256×256	1.8236	8.1601	9.9837
512×512	1.8381	37.4248	39.2629
1024×1024	33.1258	234.8187	267.9445

decryption time increases. The sensors are only responsible for the encryption process, and the decryption process is completed by the terminal (for example, a hospital data center). Since the terminal has large computational device, we don't need to consider the computation cost. In the proposed scheme, the encryption time is 0.3734s, 1.8236s, 1.8381s, 33.1258s with sizes 128×128, 256×256, 512×512, and 1024×1024, respectively. Therefore, the encryption time is short. Of course, when the image size is 1024×1024, the time is a little long. After all, the image with size 1024×1024 has millions bytes to process.

For small-sized sensors, if image size is 256×256, then i) the encryption time is 1.8236s if input size is 256×256, ii) the encryption time is 1.4936s (i.e. 4×0.3734s) if input size is 128×128. As we all known, four times of size 128×128 has the same bytes with size 256×256. The encryption time 1.4936s is less than 1.8236s. So the optimal input size is 128×128 for small-sized sensors. For large-sized sensors, if image size is 1024×1024, then the encryption time is 23.8976s (i.e. 64×0.3734s), 29.1776 (i.e. 16×1.8236s), 7.3272 (i.e. 4×1.8381s), 33.1258s (i.e. 1×33.1258s) if input sizes are 128×128, 256×256, 512×512, 1024×1024, respectively. The shortest encryption time is 7.3272s. So the optimal input size is 512×512 for large-sized sensors.

Brief summary: The adjacent pixel correlations of cipher images are less than 0.004 under horizontal, vertical and diagonal directions. In terms of histogram, the distribution of the encrypted images is uniform. The information entropy of the encrypted images is closer to the theoretical value 8. In terms of the runtime, the optimal input size is 128×128 for small-sized sensors and 512×512 for large-sized sensors.

D. Security Analysis

Key space, key sensitivity, the brute force attack, the statistical attack, and the chosen/known plaintext attacks are discussed in this section. The key space of our algorithm is mainly related to the following aspects. Firstly, there are 37 types for wavelet sparse bases. Secondly, the output of SHA-256 hash function is 256 bits, so the key space is 2^{256}. Thirdly, if the difference between the initial values is less than 10^{-16}, then these two chaotic sequences could not be distinguished for Logistic and Tent chaotic systems. The key space of one chaotic system is 10^{16}, and the key space of three chaotic systems is 10^{48}. So the total key space is approximately 2^{420}, which is calculated by $37 \times 2^{256} \times 10^{48}$. Table VIII shows the sizes of key spaces for different algorithms. Compared with the relevant papers [46]–[48], the key space in our scheme is larger than those of other algorithms.

TABLE VIII
VALUES OF KEY SPACE FOR DIFFERENT ALGORITHMS

method	Ref. [46]	Ref. [47]	Ref. [48]	SC-HAC
key space	1×10^{48}	1×10^{90}	1×10^{37}	1×2^{420}

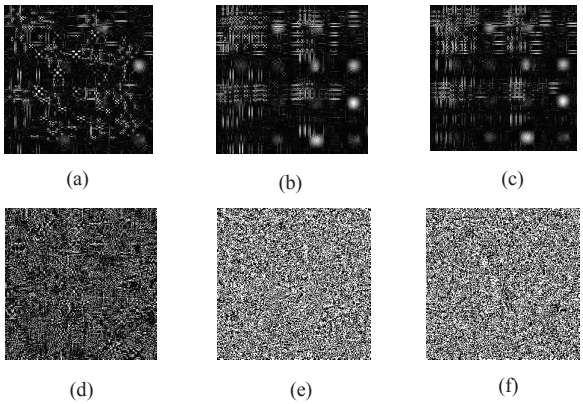


Fig. 16. Key sensitivity. The original image is Lena (256×256) in Figure 5(a1). (a), (b), (c), (d), (e) and (f) are the decrypted images with $k_1^{'}$, $k_2^{'}$, $k_3^{'}$, $k_4^{'}$, $k_5^{'}$ and $k_6^{'}$, respectively. According to Equation (15), one bit of h_1 is changed to generate the key $k_1^{'}$, one bit of h_{12} is changed to generate the key $k_2^{'}$, and one bit of h_{23} is changed to generate the key $k_3^{'}$. $k_4^{'} = k_4 + 10^{-16}$, $k_5^{'} = k_5 + 10^{-16}$, $k_6^{'} = k_6 + 10^{-16}$.

As we all known, the key sensitivity is important for image encryption. The plain image is Lena (256×256) in Figure 5(a1). Figures 16(a) - (f) are the decrypted images when the keys k_1, k_2, k_3, k_4, k_5 and k_6 are changed respectively. The changed keys are expressed as $k_1^{'}$, $k_2^{'}$, $k_3^{'}$, $k_4^{'}$, $k_5^{'}$ and $k_6^{'}$, respectively. According to Equation (15), one bit of h_1 is changed to generate the key $k_1^{'}$, one bit of h_{12} is changed to generate the key $k_2^{'}$, and one bit of h_{23} is changed to generate the key $k_3^{'}$. $k_4^{'} = k_4 + 10^{-16}$, $k_5^{'} = k_5 + 10^{-16}$, $k_6^{'} = k_6 + 10^{-16}$. The correct decryption image should be Lena image, and the incorrect decryption image is just like noise. The incorrect decryption image is disorganized, and we can not get any useful information about the original image, which shows that the proposed algorithm has the sensitivity to the decryption process. It means that the scheme prevents adversaries from decrypting the transmitted data with a key similar to the encryption key.

In the proposed scheme, attackers know the encryption algorithm, but they don't know the key. They use the intercepted data to recover the original data. The key space determines the ability of the algorithm to resist the brute force attack, and the greater the key space is, the stronger the ability to resist the brute force attack becomes. In the brute force attack, attackers know the encryption algorithm and the cipher image c_1, and they hope to get the corresponding original image p_1. Therefore, they try possible keys to decrypt the cipher images. They may try 2^{420} times at most. So the key space is large enough to resist the brute force attack in the proposed scheme.

In the statistical attack, attackers know the intercepted cipher image c_1 and the statistical regularities of original images (such as histogram, information entropy). They analyze the statistical regularities of the intercepted data, and compare

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

these two regularities. Thus, they expect to get the original image p_1. In the proposed scheme, the cipher images are uniform dispersion, and they can't be used to find the plaintext information, so our algorithm can resist the statistical attack.

The algorithm proposed in this paper can resist the chosen plaintext attack and the known plaintext attack. We add Logistic chaotic scrambling to improve the diffusion process. SHA-256 hash function is associated with the plain image, and it is used to generate the key parameters for the Arnold scrambling. Different plain images will have different hash function values, and hash values are all different even if the original image is changed with one pixel. So the algorithm is "one time, one key". In the chosen/known plaintext attacks, attackers know the encryption algorithm and the intercepted cipher image c_1, and some original images and corresponding cipher images such as $(p_2, c_2), (p_3, c_3), (p_4, c_4)$ (The original images and cipher images can be chosen in the chosen plaintext attack). They want to get the original image p_1. Because the algorithm is "one time, one key", the keys are updated all the time. Therefore, those original images and corresponding cipher images are useless to decrypt new cipher image.

Since compressive sensing is a linear transform, it lacks a diffusion mechanism. The encryption schemes based on compressive sensing [49] [50] are difficult to resist the chosen/known plaintext attacks. For example, selecting a set of plaintext sparse vectors with a sparsity of 1, the measurement matrix can be calculated so that this algorithm is cracked. In the proposed scheme, in order to resist the chosen/known plaintext attacks, Logistic chaotic scrambling is used to improve the diffusion process. And SHA-256 hash function is used to guarantee "one time, one key".

Brief summary: The total key space is approximately 2^{420}. The key sensitivity indicates that the scheme prevents adversaries from decrypting the transmitted data with a key similar to the encryption key. The proposed scheme can resist several attacks including the brute force attack, the statistical attack, the chosen plaintext attack and the known plaintext attack.

E. Comparison with Other Algorithms

The proposed scheme SC-HAC is compared with the existing compressive sensing (CS) methods, such as the traditional CS [49], [50], CS with partial hadamard matrix [51] and chaotic CS in body to body network (BBN) [48]. In Section V-C and Section V-D, we compare the adjacent pixel correlation and the key space, and analyze the comparison results. Now, we will compare the relationship between the compression ratio and the peak signal to noise ratio of the decrypted image. As shown in Fig. 17, the curve marked with the box is the result of the proposed scheme, and the size of semi-tensor measurement matrix is set to one-sixteenth of that in the traditional compressive sensing.

The original image is Pepper (512×512), the sparse basis of compressive sensing is DWT, and the reconstruction algorithm is OMP algorithm in these schemes. It is clear that our algorithm has obvious advantages when the sampling rate is low, and the quality of the decrypted images is higher. When $CR = 0.3$, the PSNR of our scheme has reached 29.4395dB,

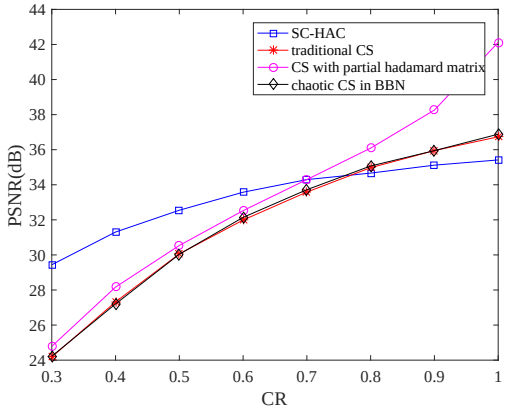


Fig. 17. Relationships between PSNR and CR for different schemes. When $CR < 0.7$, the PSNR of our scheme is obviously higher than those of other three schemes, and when $CR > 0.7$, the result of CS with partial hadamard matrix has the superiority.

and the PSNRs of the traditional CS, CS with partial hadamard matrix, chaotic compressive sensing in BBN are 24.1803dB, 24.7856dB, and 24.2210dB, respectively. When $CR < 0.7$, the PSNR of our scheme is obviously higher than those of other three schemes. In other words, when we compress a lot of data, the advantages of our scheme are obvious. When $CR > 0.7$, the result of CS with partial hadamard matrix is superior, and the result of our scheme is similar to those of other two schemes. At this time, encrypted data is compressed less, and the amount of data to be transmitted is still relatively large. In the process of compressive sensing, we certainly hope that the number of sampling is less, and the proposed scheme has more advantages when the number of sampling is less, which indicates that our scheme is better than other schemes. Simulation results demonstrate that the proposed method can achieve a good recovery effect, and outperform three previous methods when the sampling rate is low.

Brief summary: When CR is less than 0.7, the PSNR of our scheme is obviously higher than those other schemes. Numerical simulations and security analyses are given to show that our scheme performs well in encryption effect and recovery effect.

VI. CONCLUSION

The paper proposes a flexible and secure data transmission system based on semi-tensor compressive sensing in wireless body area networks. The plain image conducts DWT, then this image is transformed by Arnold scrambling. The scrambled image uses semi-tensor compressive sensing to compression and encryption, and Logistic chaotic scrambling is used to generate the cipher image. The proposed scheme is flexible and it can be used to measure the images with different sizes without adjusting the size of the measurement matrix, while other methods should change the size of the measurement matrix to fit the plain image. Because the measurement matrix is generated by the chaotic system, the transmission cost is greatly reduced by only transmitting several constants. In addition, the SHA-256 hash function resists the chosen

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

plaintext attack, and also compensates for the disadvantages of compressive sensing. By experimental simulations, we analyze the performance of the proposed encryption scheme from several perspectives. It can resist the brute force attack, the statistical attack, the known plaintext attack and the chosen plaintext attack. In the future, we consider using parallel compressive sensing instead of compressive sensing, enhancing the effect of encryption.

ACKNOWLEDGMENT

The authors would like to thank the editorial board and reviewers. This paper is supported by the National Key R&D Program of China (Grant no. 2016YFB0800602), and the National Natural Science Foundation of China (Grant nos. 61472045, 61771071, 61573067).

REFERENCES

[1] M. A. Hanson, H. C. Powell Jr, A. T. Barth, K. Ringgenberg, B. H. Calhoun, J. H. Aylor, and J. Lach, "Body area sensor networks: Challenges and opportunities," *Computer*, vol. 42, no. 1, pp. 58–65, Jan. 2009.

[2] J. Andreu-Perez, D. R. Leff, H. M. D. Ip, and G. Z. Yang, "From wearable sensors to smart implants-toward pervasive and personalized healthcare," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 12, pp. 2750–2762, Dec. 2015.

[3] J. Ko, J. H. Lim, Y. Chen, R. Musvaloiu-E, A. Terzis, G. M. Masson, T. Gao, W. Destler, L. Selavo, and R. P. Dutton, "Medisn: Medical emergency detection in sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 10, no. 1, pp. 11:1–11:29, Aug. 2010.

[4] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[5] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via l1 minimization," *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.

[6] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[7] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Transactions on Information Theory*, vol. 56, no. 4, pp. 1982–2001, Apr. 2010.

[8] P. Song, J. F. C. Mota, N. Deligiannis, and M. R. D. Rodrigues, "Measurement matrix design for compressive sensing with side information at the encoder," in *2016 IEEE Statistical Signal Processing Workshop (SSP)*, Jun. 2016, pp. 1–5.

[9] L. Gan, "Block compressed sensing of natural images," in *2007 15th International Conference on Digital Signal Processing*, Jul. 2007, pp. 403–406.

[10] V. Goudar, Z. Ren, P. Brochu, M. Potkonjak, and Q. Pei, "Optimizing the output of a human-powered energy harvesting system with miniaturization and integrated control," *IEEE Sensors Journal*, vol. 14, no. 7, pp. 2084–2091, Jul. 2014.

[11] O. Omeni, A. C. W. Wong, A. J. Burdett, and C. Toumazou, "Energy efficient medium access protocol for wireless medical body area sensor networks," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 2, no. 4, pp. 251–259, Dec 2008.

[12] R. Rajagopalan, "Energy efficient routing algorithm for patient monitoring in body sensor networks," in *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, Jun. 2016, pp. 141–146.

[13] A. Ibaida and I. Khalil, "Wavelet-based ecg steganography for protecting patient confidential information in point-of-care systems," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 12, pp. 3322–3330, Dec. 2013.

[14] A. Abuadbba and I. Khalil, "Walsh-hadamard-based 3-d steganography for protecting sensitive information in point-of-care," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 9, pp. 2186–2195, Sep. 2017.

[15] Y. S. Lee, E. Alasaarela, and H. Lee, "Secure key management scheme based on ecc algorithm for patient's medical information in healthcare system," in *The International Conference on Information Networking 2014 (ICOIN2014)*, Feb. 2014, pp. 453–457.

[16] Q. Wang, K. Ren, S. Yu, and W. Lou, "Dependable and secure sensor data storage with dynamic integrity assurance," *ACM Transactions on Sensor Networks (TOSN)*, vol. 8, no. 1, pp. 9:1–9:24, Aug. 2011.

[17] Z. Zhang, T. P. Jung, S. Makeig, and B. D. Rao, "Compressed sensing of eeg for wireless telemonitoring with low energy consumption and inexpensive hardware," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 1, pp. 221–224, Jan. 2013.

[18] S. A. Imtiaz, A. J. Casson, and E. Rodriguez-Villegas, "Compression in wearable sensor nodes: Impacts of node topology," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 4, pp. 1080–1090, Apr. 2014.

[19] Y. Liu, M. D. Vos, and S. V. Huffel, "Compressed sensing of multichannel eeg signals: The simultaneous cosparsity and low-rank optimization," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 8, pp. 2055–2061, Aug. 2015.

[20] Y. Zhang, L. Y. Zhang, J. Zhou, L. Liu, F. Chen, and X. He, "A review of compressive sensing in information security field," *IEEE Access*, vol. 4, pp. 2507–2519, 2016.

[21] R. Huang, K. Rhee, and S. Uchida, "A parallel image encryption method based on compressive sensing," *Multimedia Tools and Applications*, vol. 72, no. 1, pp. 71–93, Sep. 2014.

[22] T. Chen, M. Zhang, J. Wu, C. Yuen, and Y. Tong, "Image encryption and compression based on kronecker compressed sensing and elementary cellular automata scrambling," *Optics & Laser Technology*, vol. 84, pp. 118–133, Oct. 2016.

[23] D. Xie, H. Peng, L. Li, and Y. Yang, "Semi-tensor compressed sensing," *Digital Signal Processing*, vol. 58, pp. 85–92, Nov. 2016.

[24] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9-10, pp. 589–592, May. 2008.

[25] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[26] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[27] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM review*, vol. 43, no. 1, pp. 129–159, 2001.

[28] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1094–1121, Feb. 2012.

[29] D. Needell and J. A. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, May. 2009.

[30] D. Cheng and L. Zhang, "On semi-tensor product of matrices and its applications," *Acta Mathematicae Applicatae Sinica (English Series)*, vol. 19, no. 2, pp. 219–228, Jun 2003.

[31] L. Wu, J. Zhang, W. Deng, and D. He, "Arnold transformation algorithm and anti-arnold transformation algorithm," in *2009 First International Conference on Information Science and Engineering*, Dec. 2009, pp. 1164–1167.

[32] Q. Liu, P. Li, M. Zhang, Y. Sui, and H. Yang, "A novel image encryption algorithm based on chaos maps with markov properties," *Communications in Nonlinear Science and Numerical Simulation*, vol. 20, no. 2, pp. 506–515, Feb. 2015.

[33] Q. Zhang, L. Liu, and X. Wei, "Improved algorithm for image encryption based on dna encoding and multi-chaotic maps," *AEU-International Journal of Electronics and Communications*, vol. 68, no. 3, pp. 186–192, Mar. 2014.

[34] X. Wei, B. Wang, Q. Zhang, and C. Che, "Image encryption based on chaotic map and reversible integer wavelet transform," *Journal of Electrical Engineering*, vol. 65, no. 2, pp. 90–96, 2014.

[35] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, no. 5560, pp. 459–467, Jun. 1976.

[36] S. Li and X. Zheng, "Cryptanalysis of a chaotic image encryption method," in *2002 IEEE International Symposium on Circuits and Systems*, 2002, pp. 708–711.

[37] H. Gilbert and H. Handschuh, "Security analysis of sha-256 and sisters," in *International Workshop on Selected Areas in Cryptography*, 2004, pp. 175–193.

[38] L. Yu, J. P. Barbot, G. Zheng, and H. Sun, "Compressive sensing with chaotic sequence," *IEEE Signal Processing Letters*, vol. 17, no. 8, pp. 731–734, Aug. 2010.

[39] H. Peng, Y. Tian, and J. Kurths, "Semi tensor product compressive sensing for big data transmission in wireless sensor networks," *Mathematical Problems in Engineering*, vol. 2017, 2017.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2881129, IEEE Internet of Things Journal

[40] J. Louw, G. Niezen, T. D. Ramotsoela, and A. M. Abu-Mahfouz, "A key distribution scheme using elliptic curve cryptography in wireless sensor networks," in *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, Jul. 2016, pp. 1166–1170.
[41] S. H. Seo, J. Won, S. Sultana, and E. Bertino, "Effective key management in dynamic wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, pp. 371–383, Feb. 2015.
[42] R. Ganesan, M. Gobi, and K. Vivekanandan, "A novel digital envelope approach for a secure e-commerce channel," *International Journal of Network Security*, vol. 11, no. 3, pp. 121–127, Nov. 2010.
[43] R. Srinivasan, D. Vaidehi, J. Balaji, and S. Heema, "A single chip efficient fpga implementation of rsa and des for digital envelope scheme," *WSEAS Transactions on Communications*, vol. 3, no. 2, pp. 664–669, 2004.
[44] M. G. Rashed, S. Ullah, and R. Yasmin, "Secured message data transactions with a digital envelope (de)-a higher level cryptographic technique," in *International Conference on Engineering Research, Innovation and Education 2013*, Jan. 2013.
[45] A. Wang, F. Lin, Z. Jin, and W. Xu, "A configurable energy-efficient compressed sensing architecture with its application on body sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 1, pp. 15–27, Feb. 2016.
[46] Y. Zhang, B. Xu, and N. Zhou, "A novel image compression–encryption hybrid algorithm based on the analysis sparse representation," *Optics Communications*, vol. 392, pp. 223–233, Jun. 2017.
[47] X. Chai, Z. Gan, and M. Zhang, "A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion," *Multimedia Tools and Applications*, vol. 76, no. 14, pp. 15 561–15 585, Jul. 2017.
[48] H. Peng, Y. Tian, J. Kurths, L. Li, Y. Yang, and D. Wang, "Secure and energy-efficient data transmission system based on chaotic compressive sensing in body-to-body networks," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 11, no. 3, pp. 558–573, Jun. 2017.
[49] A. Orsdemir, H. O. Altun, G. Sharma, and M. F. Bocko, "On the security and robustness of encryption via compressed sensing," in *MILCOM 2008 - 2008 IEEE Military Communications Conference*, Nov. 2008, pp. 1–7.
[50] Y. Rachlin and D. Baron, "The secrecy of compressed sensing measurements," in *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, Sep. 2008, pp. 813–817.
[51] N. Zhou, A. Zhang, J. Wu, D. Pei, and Y. Yang, "Novel hybrid image compression–encryption algorithm based on compressive sensing," *Optik-International Journal for Light and Electron Optics*, vol. 125, no. 18, pp. 5075–5080, Sep. 2014.

Lixiang Li received the M.S. degree in circuit and system from Yanshan University, Qinhuangdao, China, in 2003, and the Ph.D. degree in signal and information processing from Beijing University of Posts and Telecommunications, Beijing, China, in 2006. She is currently a professor at the School of CyberSpace Security, Beijing University of Posts and Telecommunications, China. The winner of National Excellent Doctoral theses, the New Century Excellent Talents in University, the winner of Henry Folk Education Foundation, the winner of Hong Kong Scholar Award, the winner of Beijing Higher Education Program for Young Talents, the winner of Outstanding Youth Award of Chinese Association for Cryptology Research. Visiting Potsdam Institute for Climate Impact Research, Germany from July 2011 to June 2012. Engaged in the research of compressive sensing, complex networks, swarm intelligence and network security; having published more than 100 papers and a monograph.

Lifei Liu received the B.S.degree in computer science and technology from North China of Electric Power University, Hebei, China, in 2016. Now she is working towards the M.S degree in information security at Beijing University of Posts and Telecommunications, Beijing, China. Her major interests are semi-tensor product, compressive sensing and image encryption.

Haipeng Peng received the M.S. degree in system engineering from Shenyang University of Technology, Shenyang, China, in 2006, and the Ph.D. degree in signal and information processing from Beijing University of Posts and Telecommunications, Beijing, China, in 2010. He is currently a professor at the School of CyberSpace Security, Beijing University of Posts and Telecommunications, China. His research interests include compressive sensing, information security, network security, complex networks and control of dynamical systems. Dr. H. Peng is the co-author of 50 scientific papers.

Yixian Yang received the M.S. degree in applied mathematics in 1986 and the Ph.D. degree in electronics and communication systems in 1988 from Beijing University of Posts and Telecommunications, Beijing, China. He is the Managing Director of information security center, Beijing University of Posts and Telecommunications, Beijing, China. The Yangtze River scholar Program professor, National Outstanding Youth Fund winners, the National Teaching Masters. Major in coding and cryptography, information and network security, signal and information processing; having authored more than 40 national and provincial key scientific research project, published more than 300 high-level papers and 20 monographs.

Shizhuo Cheng received the M.S. degree in 2007 and the Ph.D. degree in 2013 from Harbin Institute of Technology. She was a visiting student in The University of Sheffield from 2009 to 2010. Since 2016, she is a postdoctoral fellow in The University of Virginia. She has held on 3 the national funds and 2 provincial and ministerial funds. She has published more than 30 papers and 2 academic monographs. She has edited (translated) 2 academic works and 1 invention patent.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.