IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 1, 2018, accepted June 1, 2018, date of publication June 5, 2018, date of current version June 29, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2844184

A Secure Compressive Sensing-Based Data Gathering System via Cloud Assistance

SUNG-HSIEN HSIEH[1,2], **TSUNG-HSUAN HUNG**[2,3], **CHUN-SHIEN LU**[2], **YU-CHI CHEN**[4], **AND SOO-CHANG PEI**[1], **(Fellow, IEEE)**

[1]Graduate Institute of Communication Engineering, National Taiwan University, Taipei 106, Taiwan
[2]Institute of Information Science, Academia Sinica, Taipei 115, Taiwan
[3]Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan
[4]Department of Computer Science and Engineering, Yuan Ze University, Taoyuan 320, Taiwan

Corresponding author: Chun-Shien Lu (lcs@iis.sinica.edu.tw)

This work was supported by the Ministry of Science and Technology, Taiwan, under Grants III 105-1629, MOST 104-2221-E-001-019-MY3, MOST 104-2221-E-001-030-MY3, and MOST 106-2218-E-155-008-MY3.

ABSTRACT Wireless sensors have been helpful and popular for gathering information, in particular in harsh environments. Due to the limit of computation power and energy, compressive sensing has attracted considerable attention in achieving simultaneous sensing and compression of data on the sensor/encoder side with cheap cost. Nevertheless, along with the increase of the data size, the computation overhead for decoding becomes unaffordable on the user/decoder side. To overcome this problem, by taking advantage of resourceful cloud, it is helpful to leverage the overhead. In this paper, we propose a cloud-assisted compressive sensing-based data gathering system with security assurance. Our system, involving three parties of sensor, cloud, and user, possesses several advantages. First, in terms of security, for any two data that are sparse in certain transformed domain, their corresponding ciphertexts are indistinguishable on the cloud side. Second, to avoid the communication bottleneck between the user and cloud, the sensor can encrypt data individually such that, once the cloud receives encrypted data from sensor, it can immediately carry out its task without requesting any information from the user. Third, we show that, even though the cloud knows the permuted support information of data, the security never is sacrificed. Meanwhile, the compression rate can be reduced further. Theoretical and empirical results demonstrate that our system is cost effective and privacy guaranteed and that it possesses acceptable reconstruction quality.

INDEX TERMS Compressive sensing, cloud assistance, security, data gathering, encryption.

I. INTRODUCTION

A. BACKGROUND

Wireless sensors for monitoring and gathering data have grown rapidly since the cost of sensors is low and they are easy to set up [1], [2]. For example, there are many applications, including healthcare [3], traffic control [4], and military area surveillance [5], which widely utilize wireless sensors. Nevertheless, the bottleneck of sensors is resource-constrained due to the limit of energy and floating-point operations per second (FLOPS). Thus, how to reduce the computation and communication overhead is a critical issue for the use of wireless sensors. Conventional approaches [6], [7] usually compress data before transmission on sensors. Such a solution is efficient to reduce the communication cost.

On the other hand, privacy assurance is another concern, especially when data are privacy-sensitive. To this end, the security must be ensured from the beginning (*e.g.*, sensor). A trend is to compress and encrypt data simultaneously based on chaotic map [8], [9]. Nevertheless, they suffer the problem of ciphertext expansion in that the length of ciphertext becomes longer than that of plaintext, resulting in higher communication overhead.

Compressive sensing (CS) recently has been an emerging and feasible solution to the aforementioned problems. The framework of CS is composed of fast sensing/encoder and slow recovery/decoder. On the encoder side, the original signal is sensed and compressed simultaneously via a sensing matrix to obtain measurements or the so-called measurement vector. In contrast, the decoder suffers from the overhead of reconstructing a high-dimensional original signal from its corresponding low-dimensional measurement vector, as this procedure is usually time-consuming.

To meet the requirement of preserving secrecy, the sensing matrix is considered as a key in [10], where perfect secrecy defined by Shannon [11] is not achievable but computational secrecy can be guaranteed. Such a framework only considers two participants, sensor and user. Considering the fact that the decoding process in CS is time-consuming, the overhead may not be acceptable, especially for large-scale data.

B. RELATED WORK

We classify the existing works for CS-based data gathering based on two characteristics: i) with or without cloud assistance and ii) know or do not know input signals on the sensor. Specifically, with cloud assistance, the task with significant overhead on the decoder side will be carried out by a cloud instead of a user. Along with the setting of cloud-assistance, an accompanying challenge is that a sensing matrix no longer is considered as a key; otherwise, the cloud cannot carry out decoding. In addition, CS requires that the input signal be unknown on the encoder because compressing and sensing are done simultaneously. On the contrary, if the input signal is known, it means the sampling rate on the encoder cannot break out Nyquist rate.

1) KNOW INPUT SIGNAL BUT WITHOUT CLOUD ASSISTANCE

In the literature, to enhance the security and privacy, Qi *et al.* [12] proposed a hybrid system using 8-bit integer chaotic block encryption and message authentication codes (MACs)-based hashing. Qi *et al.* [13] used pseudo-random permutation (PRP) and symmetric encryption to encrypt measurements. Similarly, Xie *et al.* [14] exploited homomorphic encryption to encrypt input signals such that the privacy is preserved. The aforementioned methods, however, did not consider cloud assistance and required to know the input signals. Hu *et al.* [15] proposed two statistical inference attacks for non-cloud assisted systems such that the sensing matrix, treated as the key, may be estimated. Thus, traditional CS-based methods may suffer from information leakage under these attacks.

2) DO NOT KNOW INPUT SIGNAL BUT WITH CLOUD ASSISTANCE

Wang *et al.* [16], [17] first proposed a cloud-assisted system based on CS. Their key idea is that the original data are one-to-one mapped into random data, which are recovered on the cloud by using linear programming (LP). Then, the user can reconstruct data efficiently by inverse mapping. The authors claimed that any two ciphertexts are indistinguishable in terms of statistical distance. On the contrary, Hung *et al.* [18] did not use the encryption scheme as in [16] and [17]. Thus, they returned to use convex programming on the cloud because the length of reconstructed signals by linear programming is twice longer than that by convex programming, leading to more storage cost. In addition, their encryption aims to permute the sparse data. Zhang *et al.* [19] extended this framework for multiple input signals and proposed using Arnold Transform to scramble the positions of multiple input signals. Xue *et al.* [20] also aims to a cloud-assisted system, where the ciphertext still includes statistical information such that the user can calculate average, sum, and standard deviation of the plaintext without decryption. However, from the viewpoint of security, statistical information is easily leaked to the attacker. To prevent the cloud from reconstructing the plaintext, the user also must pay communication cost for sending an encrypted matrix to the cloud. The aforementioned methods do not assume to know the input signals.

3) KNOW INPUT SIGNAL WITH CLOUD ASSISTANCE

Zhang *et al.* [21] assumed that the data to be sensed are known to the sensor for achieving higher security. They also proposed another work [22] considering two kinds of clouds, private cloud and public cloud. Each image is partitioned into a small set of sensitive data and a large set of insensitive data, which are securely stored in the private cloud and the public cloud, respectively. The process of partition, however, involves the knowledge of input signal.

C. CONTRIBUTION

In this paper, our system follows a similar framework to [17] with sensor, cloud, and user. The prerequisite is that data are unknown for any participants and possess sparsity in a certain domain. To reduce the cost on the sensor side, we first use CS to simultaneously sense, encrypt, and compress data before sending the compressed and encrypted data to the cloud for signal recovery. Then, the reconstructed but still encrypted data is stored on the cloud and sent to user for decryption when the user issues a query. Among them, the cloud is responsible for the time-consuming signal recovery task.

We summarize the contributions of our system and compare it with [17] as follows.

(1) The secure mechanism in Wang *et al.*'s system requires transmitting extra information from the user to the cloud once the cloud wants to reconstruct encrypted data. On the contrary, in our system, when the cloud receives the data from the sensor, the cloud can immediately start to reconstruct and store the data. No communication cost for extra information is required between the user and cloud. In other words, when the user issues a query, it only needs to transmit the query message and the cloud can immediately return the reconstructed data without spending any waiting time for reconstruction, whereas [17] does.

(2) We present a new concept of sparsity-based data encryption. The encryption scheme is designed to add another sparse random data to the original data for hiding information on the sensor side. When data are sparse, our system achieves security in that, given any two plaintexts, their corresponding ciphertexts are

indistinguishable in terms of statistical distance. We further study how sparse is enough for our system.

(3) In (2), the encrypted data become more non-sparse (corresponding to increase of non-zero entries) than its original counterpart. In the context of CS, the reconstruction quality, in general, degrades along with the increase of non-zero entries in the encrypted data. To overcome this problem, we show that ℓ_1-minimization with the information of support set, which is transmitted from the sensor to the cloud, can maintain the reconstruction quality without losing security. Since the cardinality of support set is related to the sparsity that is less than the number of measurements, this encryption scheme possesses low computation, storage, and communication costs on the sensor side.

D. ORGANIZATION OF THIS PAPER

After introducing the background of the cloud-assisted compressive sensing-based data gathering problem, the preliminary is described in Sec. II to make this paper self-contained. In Sec. III, the model formulation, including system model, threat mode, and overview of the proposed compressed sensing with encryption in communication (CSEiC) system, is described. Then, we specifically describe CSEiC, including performance and security analyses, in Sec. IV. Finally, simulation results are given in Sec. V to verify our method. In Appendix, we present an attack and formally prove that Wang *et al.*'s method [17] is not as secure as they claimed.

II. PRELIMINARY

Compressive sensing [23] includes two components: encoder and decoder. On the encoder side, given a K-sparse signal $x \in \mathbb{R}^N$ (having K non-zero entries), the corresponding measurement vector can be obtained via sensing matrix $A \in \mathbb{R}^{M\times N}$ as $y = Ax$ with $K \leq M < N$. If the encoding phase is contaminated by noise, namely $y = Ax + e$, with $e \in \mathbb{R}^M$ being Gaussian random noise, ℓ_1-minimization [23]–[25] is an efficient tool to reconstruct x on the decoder side:

$$\hat{x} = \arg\min_{\bar{x}} \|\bar{x}\|_{\ell_1} \quad s.t. \ \|y - A\bar{x}\|_{\ell_2} \leq \epsilon, \tag{1}$$

where $\epsilon \geq 0$ is often set to $\|e\|_{\ell_2}$.

Restricted isometric property (RIP), involving a sufficient condition of sparse signal recovery in the context of CS, is the base of our proofs in this paper and is described as follows.

Lemma 1 (RIP): Let $A \in \mathbb{R}^{M\times N}$. Suppose that there exists a constant $\delta_{|I|} < 1$ such that, for any $x \in \mathbb{R}^{|I|}$ and any $I \subset \Omega = \{1, 2, .., N\}$,

$$(1 - \delta_{|I|})\|x\|_2^2 \leq \|A_I x\|_2^2 \leq (1 + \delta_{|I|})\|x\|_2^2 \tag{2}$$

holds, where A_I is a matrix formed by columns of A with indices belonging to I.

The matrix A is said to satisfy the $|I|$-restricted isometry property with restricted isometry constant (RIC) $\delta_{|I|}$.

Lemma 2 [24]: If the sensing matrix A satisfies the RIP of both orders K_1 and K_2, then $\delta_{K_1} \leq \delta_{K_2}$ for any $K_1 \leq K_2$.

Lemma 3 (Consequences of RIP [26]): Let $I_1, I_2 \subset \Omega$ be two disjoint sets ($I_1 \cap I_2 = \emptyset$). If $\delta_{|I_1|+|I_2|} < 1$, then

$$\|(A_{I_1})^T A_{I_2} x\|_2 \leq \delta_{|I_1|+|I_2|}\|x\|_2$$

holds for any x.

For example, when $\delta_{2K} \leq 0.414$ [26], the decoder achieves perfect recovery with $\hat{x} = x$. In addition, a random matrix is known to satisfy $\delta_{dK} < t$ with high probability provided one chooses $M = O\left(\frac{dK}{t^2}\log\frac{N}{K}\right)$ [27], where d and t are parameters relevant to M in the context of compressive sensing. For example, if the sufficient condition of perfect recovery is $\delta_{2K} \leq 0.414$, it means that $d = 2$ and $t = 0.414$.

III. MODEL FORMALIZATION

In this section, before introducing the proposed CSEiC system, we first define the system model and the threat model.

A. SYSTEM MODEL

We introduce the model of our system, called *compressed sensing with encryption in communication* (CSEiC), with three parties: *sensor* $\mathcal{S}$, *cloud* $\mathcal{C}$, and *user* $\mathcal{U}$. We keep using the notations x, A, and y as the original data, sensing matrix, and measurement vector, respectively, as in the previous sections. In our system, which is depicted in Fig. 1, the sensing matrix A is public and will be used by $\mathcal{S}$ and $\mathcal{C}$. First, $\mathcal{S}$ needs to do compression, sensing, and encryption using a one-time secret key sk and A, then sends encrypted measurement y' (ciphertext of y) to $\mathcal{C}$. After receiving y', $\mathcal{C}$ will use y' and A to solve an ℓ_1-minimization problem to obtain and store encrypted data $\hat{z}$, which is a ciphertext of reconstructed signal z and will be sent to $\mathcal{U}$ if $\mathcal{U}$ issues a query. Upon receiving $\hat{z}$, $\mathcal{U}$ can decrypt $\hat{z}$ using sk to obtain the recovered data z.

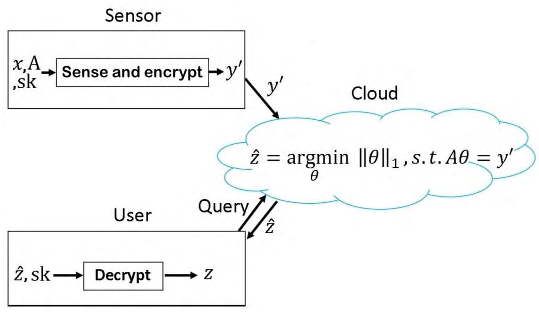

FIGURE 1. Flowchart of our system.

Since the goal of our system is to guarantee the security of outsourcing computations in $\mathcal{C}$, the input and output of the optimization problem are encrypted for protection purposes. The optimization problem (ℓ_1-minimization) is public, which means the computation is known to each party.

We take the healthcare system mentioned in [17] as a real example. The sensor accounts for collecting various raw data about healthcare. To lower the computation overheads with respect to the sensed data, the sensor in our system will send compressed and encrypted data samples to the cloud with privacy preserved. Cloud is responsible for providing data retrieval for users. Here the user might be a healthcare workstation operated by a physician in a hospital.

B. THREAT MODEL

As mentioned, our goal is to guarantee the security of input and output in the system. In this paper, we only assume that only $\mathcal{C}$ is adversary and that it is semi-honest, where it will do the computation honestly but be curious about the input and output. The $\mathcal{C}$ can only know information from A, $\hat{z}$, and y'. The adversary considered in this paper is a kind of *ciphertext-only attack*.

C. OVERVIEWS OF CSEIC SYSTEM

CSEiC is composed of four algorithms: Keygen, ECS, $\mathsf{ERecovery}$, and DCS. The formal definition of CSEiC is as follows.

Definition 1: With a parameter pp depending on κ and a sensing matrix A depending on pp, which are public, CSEiC is a tuple of probabilistic polynomial time (PPT) algorithms, denoted by CSEiC={Keygen, ECS, $\mathsf{ERecovery}$, DCS}.

- $(\mathsf{sk}) \leftarrow \mathsf{Keygen}(1^\kappa, pp)$: takes as input of security parameter κ and pp, and outputs a secret key sk for encryption on the sensor and user sides.
- $(y') \leftarrow \mathsf{ECS}(A, pp, \mathsf{sk}, x)$: takes A, pp, and sk to sense, compress, and encrypt the original signal x, and outputs a ciphertext y'.
- $(\hat{z}) = \mathsf{ERecovery}(A, y')$: takes A to recover/decompress the data y' sent out from the sensor, and outputs a ciphertext $\hat{z}$.
- $(z) = \mathsf{DCS}(pp, \mathsf{sk}, \hat{z})$: takes pp and sk to decrypt $\hat{z}$, and outputs a reconstructed signal z.

There are two main characteristics of CSEiC.

Free-Error Correctness: For original data x, if $\mathsf{ERecovery}$ (A, y') outputs decryptable ciphertext $\hat{z}$, then we have

$$Pr[\mathsf{DCS}(pp, \mathsf{sk}, \hat{z}) = x] = 1.$$

Security: CSEC is κ-secure if, for any two K-sparse data x_0 and x_1, there exists a negligible function negl such that

$$SD(\Phi_0, \Phi_1) \leq \mathsf{negl}(\kappa),$$

where $SD(\Phi_0, \Phi_1)$ measures the statistical distance between two tasks, Φ_0 and Φ_1,[1] $\Phi_b = (y'_b, A, \hat{z}_b)$, $(\hat{z}_b) = \mathsf{ERecovery}(A, y'_b)$, $(y'_b) \leftarrow \mathsf{ECS}(A, pp, \mathsf{sk}_b, x_b)$, and $(\mathsf{sk}_b) \leftarrow \mathsf{Keygen}(1^\kappa, pp)$ for $b \in \{0, 1\}$.

Since both pp and A are fixed, they can be used in multiple different tasks. We remark that A does not need to change at all times. This property reduces the communication costs of $\mathcal{S}$ and $\mathcal{U}$, and leads to better efficiency than [17].

In our system, the inherent security of CSEiC enables a powerful guarantee in that the PPT adversary distinguishes both tasks, Φ_0 and Φ_1, with probability less than $\mathsf{negl}(\kappa)$ for any two K-sparse signals. Thus, $(y', A, \hat{z})$ observed by $\mathcal{C}$ does not leak any information of x.

IV. THE CSEIC SYSTEM

In this section, we first introduce the construction of CSEiC in IV-A. Then, we analyze the correctness and security of CSEiC in IV-B. Finally, we compare the cost of CSEiC with that of [17].

Remarks, notations, and assumptions:

- $\mathbb{PM}$ is a distribution of random permutation matrix.
- Original signal x is in an analog form on the sensor side, but is digital on the user side.
- x is supposed to be K-sparse with $K \leq M \leq N$ to meet the requirements in CS and $x[i] \in [-L, L]$ for all i's.
- Given a vector u, we denote $u[i]$ as i-th entry of u. $abs(\cdot)$ denotes an element-wise absolute function.
- $N = poly(\kappa)$, $M = poly(\kappa)$, and $K = poly(\kappa)$.
- Given a distribution $\mathbb{D}$, $H \leftarrow \mathbb{D}$ denotes H is sample from $\mathbb{D}$.
- $H \overset{\$}{\leftarrow} \mathbb{D}^{R \times C}$ denotes each entry of H is an i.i.d. sample from $\mathbb{D}$, and H is an $R \times C$ matrix.
- $\mathbb{N}(0, \sigma^2)$ denotes a normal distribution with mean 0 and variance σ^2.
- $\mathbb{U}(D)$ denotes a uniform distribution with range $[-D, D]$.
- We set $pp = (M, N, K)$ and $A \overset{\$}{\leftarrow} \mathbb{N}(0, \frac{1}{M})^{M \times N}$.

A. CONSTRUCTION OF CSEIC

In this section, we describe the detailed construction of CSEiC. The formal descriptions are shown in Algorithms 1, 2, 3, and 4. Before getting into detail, we will first summarize our encryption idea. Suppose $y = APx$, where the permutation matrix P is used to protect the positions of non-zero entries of x against ciphertext-only attack (CoA), as in [18], if we consider Px to be a ciphertext. Furthermore, the real positions are unknown, meaning attackers cannot reconstruct the plaintext x perfectly. Px, however, still leaks information about the values of x. To overcome this problem, an intuitive way is to design P as a generalized permutation matrix, where each diagonal entry is drawn from a standard normal distribution. For this, let (i, j) be a pair, where i/j denotes the index after/before permutation. Then, we have

$$(Px)[i] \sim \mathbb{N}\left(0, x[j]^2\right).$$

Note that Px, however, still is not statistically independent of x. Actually, the system [17] has the same security breach with the aforementioned method, which will be proved formally in Appendix. In the following, we present another solution to address this problem.

Inspired by the facts that conventional encryption adds a random vector to Px to hide the values of Px and that random projection in compressive sensing is linear, we first construct a random vector u', where each entry is drawn

[1] There are many different measures about statistical distance. We adopt the general form defined in (8) in Sec. IV-B.

from i.i.d. uniform distribution over $[-D, D]$. Let Σ be a diagonal matrix, where the diagonal entry is either 0 or 1, depending on the support set S, and let $u = P^{-1}u'$. Although Px is unknown, we have $y = APx$. Thus, instead of directly computing $Px - \Sigma u'$, we subtract y by $A\Sigma u'$ to generate y', which is sent to the cloud, for the goal of protecting x as:

$$y' = y - A\Sigma u' = APx - A\Sigma u' = A\left(Px - \Sigma Pu\right) = Ax', \tag{3}$$

where

$$x' = Px - \Sigma Pu. \tag{4}$$

Note that this "minus" computation in (3) helps us hide the information of non-zero entries in x thanks to x'.

Intuitively, if the output $\hat{z}$ of ERecovery is equal to x', then DCS outputs $z = x$ by $z = P^{-1}\left(\hat{z} + \Sigma Pu'\right)$. The analysis will be presented in the next subsection.

With this idea, the four algorithms are described as follows.

- Keygen (Algorithm 1): It uses M, N, and pp to create a one-time vector u' and one-time permutation matrix P in order to avoid the security breach, where the attacker accumulates enough plaintexts and the corresponding ciphertexts to estimate u' and P via regression.
- ECS (Algorithm 2): The original measurement vector y without encryption is sensed via $y = APx$. We detect the support S of Px via collecting the indices of the first $2K$ largest entries of $abs(A^Ty)$. Finally, the ciphertext is transmitted, according to (3), to cloud as:

$$y' = y - A\Sigma u'. \tag{5}$$

- ERecovery (Algorithm 3): It uses y' and A to solve an ℓ_1-minimization problem and obtain ciphertext $\hat{z}$.
- DCS (Algorithm 4): It uses u' and P to decrypt $\hat{z}$ as z.

It should be noted that we generate u' and P randomly for the encryption each time. In other words, u' and P can be used as one-time key. In practice, we always need plenty of secret keys to encrypt signals, where the secret keys must be stored in advance on both the sender and user sides. Storing these keys seems to consume a lot of storage, where such a situation also happens in Wang *et al.*'s method. In fact, we can avoid the excessive use of storage by classical methods in that a master secret key is pre-shared to generate plenty of u' and P on both the sender and user sides by means of pseudorandom function, which is one of well-studied cryptographic primitives [28] in achieving computational security.

Algorithm 1 Keygen

```
Input: $1^\kappa$, $pp$.
    Parse $pp = (M, N, K)$;
    $u' \xleftarrow{\$} \mathbb{U}(D)^N$ with $D = 2^\kappa$;
    $P \xleftarrow{\$} \mathbb{PM}$.
Output: $\mathsf{sk} = (u', P)$.
```

Algorithm 2 ECS

```
Input: $A$, $pp$, $\mathsf{sk}$, $x$.
    Parse $\mathsf{sk} = (u', P)$;
    Parse $pp = (M, N, K)$;
    Sensing $y = APx$;
    $r = abs(A^Ty)$;
    Set $S$ as a set of collecting indices of the first $2K$ largest entries of $r$;
    Set $\Sigma[j,j] = \begin{cases} 1, & \text{if } j \in S \\ 0, & \text{otherwise} \end{cases}$, where $\Sigma \in \mathbb{R}^{N\times N}$;
    Set $y' = y - A\Sigma u'$.
Output: $y'$.
```

Algorithm 3 ERecovery

```
Input: $A$, $y'$.
    Obtain $\hat{z}$ by solving the following problem:
        $\arg\min_{\bar{x}} \|\bar{x}\|_1 \quad s.t. \quad y' = A\bar{x}.$
Output: $\hat{z}$.
```

Algorithm 4 DCS

```
Input: $pp$, $\mathsf{sk}$, $\hat{z}$.
    Parse $\mathsf{sk} = (u', P)$;
    Set $S$ as a set of collecting indices of the first $2K$ largest entries of $abs(\hat{z})$;
    Set $\Sigma[j,j] = \begin{cases} 1, & \text{if } j \in S \\ 0, & \text{otherwise} \end{cases}$, where $\Sigma \in \mathbb{R}^{N\times N}$;
    $z = P^{-1}\left(\hat{z} + \Sigma Pu'\right)$.
Output: $z$.
```

B. ANALYSIS

1) FREE-ERROR CORRECTNESS

In this section, we show that, if $\hat{z} = x'$, P is invertible and both S's on the sensor and the cloud are the same, then $z = x$, which will achieve free-error correctness. Since P is a permutation matrix, it will be invertible.

To show the exact recovery $\hat{z} = x'$ on the cloud, if x' is K'-sparse and $M = O\left(K' \log \frac{N}{K'}\right)$ due to perfect recovery of CS, then the output of decoder is x'. Since $x' = Px - \Sigma Pu$, x is K-sparse, and the diagonal entries of Σ only have $2K$ non-zero entries, x' must be less than or equal to $3K$. Thus, M is still $O\left(K \log \frac{N}{K}\right)$.

According to the algorithm, $\hat{z}$ is the signal x added with a $2K$-sparse random vector, whose the support set S on the sensor is decided by finding the first $2K$ largest entries of $abs(A^Ty)$. When S includes the real support set of x (discussed in Theorem 2 in the next section), it means that the support set of $\hat{z}$ must be equal to S on the sensor. Thus, by finding the first $2K$ largest entries of $abs(\hat{z})$, we can ensure that both S's on the sensor and the user are the same. Consequently, free-error correctness holds.

2) SECURITY

Here, we discuss the security of our system. First, we show that distinguishing any two tasks, Φ_0, and Φ_1, is equivalent to distinguishing the statistical distance between corresponding ciphertext, $\hat{z}_0$ and $\hat{z}_1$. We can derive:

$$\begin{aligned} SD(\Phi_0, \Phi_1) &= SD((y'_0, A, \hat{z}_0), (y'_1, A, \hat{z}_1)) \\ &= SD((A\hat{z}_0, A, \hat{z}_0), ((A\hat{z}_1, A, \hat{z}_1))) \\ &\leq SD((A, \hat{z}_0), (A, \hat{z}_1)) \\ &= SD(\hat{z}_0, \hat{z}_1). \end{aligned}$$

The second equality is based on $y'_b = A\hat{z}_b$ for $b \in \{0, 1\}$. The inequality is derived because $A\hat{z}_b$ is a function of $\hat{z}_b$ for $b \in \{0, 1\}$ and $SD(f(X), f(Y)) \leq SD(X, Y)$ for any random variable X and Y with a deterministic function f. The last equality follows from the fact that A is independent of $\hat{z}_b$ and A is identical in two tasks.

As described in Sec. IV-B.1), $\hat{z} = x'$ is ensured due to free-error correctness. By (4), we derive that $\hat{z} = Px - \Sigma Pu$, which is further considered as a random vector with:

$$\hat{z}[i] \sim \begin{cases} x[j] - \mathbb{U}(D), & \text{if } i \in S \\ x[j], & \text{if } (Px)[i] \neq 0 \text{ and } i \notin S \\ 0, & \text{if } (Px)[i] = 0 \text{ and } i \notin S. \end{cases} \tag{6}$$

Ideally, if $\{i | (Px)[i] \neq 0\} \subset S$, namely perfect support detection of Px, then (6) is further reduced to

$$\hat{z}[i] \sim \begin{cases} x[j] - \mathbb{U}(D), & \text{if } i \in S \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Based on these observations, we show the statistical distance between $\hat{z}_0$ and $\hat{z}_1$ is bounded by a negligible function as follows.

Theorem 1: Suppose Φ_0 and Φ_1 are any two tasks and the corresponding original signals x_0 and x_1 are K-sparse. If perfect support detection with $|S| = cK$ holds, then

$$SD(\Phi_0, \Phi_1) \leq \frac{cKL}{D}.$$

Proof: Suppose S_b contains the support set of permuted signal $P_b x_b$, for b=0 or 1. Let $\hat{z}_b = P_b x_b - \Sigma_b u'_b$ be considered as two multivariate random variables, where, for all non-zero entries, the positions are distributed uniformly over $[1, N]$ and the values follow a uniform distribution with ranges different from those in (7). Since $\hat{z}_b[i] = 0$ for all $i \in S_b$ with probability 0, we make an assumption that the range of all uniform random variables does not contain zero. Namely, $\hat{z}_b[i]$ uniformly distributes over $[x_b[j] - D, x_b[j] + D] - \{0\}$ for all $i \in S_b$. In other words, the sparsity of $\hat{z}_b$ is cK, which simplifies our subsequent proof. This assumption does not change statistical distance, defined as:

$$SD(\hat{z}_0, \hat{z}_1) = \frac{1}{2} \int_{\mathbb{R}^N} \left| f_{\hat{z}_0}(v) - f_{\hat{z}_1}(v) \right| dV, \tag{8}$$

where $f_{\hat{z}_0}$ and $f_{\hat{z}_1}$ are probability density functions (p.d.f.) of $\hat{z}_0$ and $\hat{z}_1$, respectively, $v = [v_1; v_2; \ldots; v_N]$, and $V = |dv_1 \times dv_2 \times \ldots \times dv_N|$.

For a cK-sparse vector, there are $\binom{N}{cK}$ support sets. Let $\omega_1, \ldots, \omega_{\binom{N}{cK}}$ be all possible sets collecting cK indices from range $[1, 2, \ldots, N]$, and let $\mathsf{sp}(\hat{z})$ generate the support set of $\hat{z}$. We define

$$\Omega_i = \left\{ v \mid v \in \mathbb{R}^N \text{ and } \mathsf{sp}(v) = \omega_i \right\}.$$

We have $f_{\hat{z}_0}(v) = f_{\hat{z}_1}(v) = 0$ if $v \notin \Omega_i$ for all i's. Since $\Omega_i \bigcap \Omega_j = \emptyset$ for any $i \neq j$, we can derive:

$$SD(\hat{z}_0, \hat{z}_1) = \frac{1}{2} \sum_{i=1}^{\binom{N}{cK}} \int_{\Omega_i} \left| f_{\hat{z}_0}(v) - f_{\hat{z}_1}(v) \right| dV. \tag{9}$$

In addition, we can derive:

$$\begin{aligned} &\int_{\Omega_i} \left| f_{\hat{z}_0}(v) - f_{\hat{z}_1}(v) \right| dV \\ &= \frac{1}{\binom{N}{cK}} \int_{\Omega_i} \left| \sum_{j=1}^{\binom{N}{cK}} f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_j\right) \right. \\ &\quad \left. - \sum_{j=1}^{\binom{N}{cK}} f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_j\right) \right| dV \\ &= \frac{1}{\binom{N}{cK}} \int_{\Omega_i} \left| f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_i\right) \right. \\ &\quad \left. - f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_i\right) \right| dV. \end{aligned} \tag{10}$$

In (10), the first equality relies on the conditional probability

$$f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_j\right) \mathsf{Pr}\left(\mathsf{sp}(\hat{z}_0) = \omega_j\right)$$

and $\mathsf{Pr}\left(\mathsf{sp}(\hat{z}_0) = \omega_j\right) = 1/\binom{N}{cK}$ for any j. The second equality holds because, if $\mathsf{sp}(v) \neq \omega_j$ for $v \in \Omega_i$ and $i \neq j$, then $f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_j\right) = 0$ and $f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_j\right) = 0$.

Given ω_i for any i, the number of permutations of cK non-zero entries is $cK!$. Let $\pi^i_1, \ldots, \pi^i_{cK!}$ be all possible orders and let $\mathsf{o}_{\omega_i}(\hat{z})$ output the order of $\hat{z}$. Then,

$$\begin{aligned} (10) &= \frac{1}{cK!\binom{N}{cK}} \int_{\Omega_i} \left| \sum_{j=1}^{cK!} f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi^i_j\right) \right. \\ &\quad \left. - \sum_{j=1}^{cK!} f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi^i_j\right) \right| dV \\ &\leq \frac{1}{cK!\binom{N}{cK}} \sum_{j=1}^{cK!} \int_{\Omega_i} \left| f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi^i_j\right) \right. \\ &\quad \left. - f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi^i_j\right) \right| dV. \end{aligned} \tag{11}$$

Furthermore, we can derive:

$$\int_{\Omega_i} \left| f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0)=\omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0)=\pi_j^i\right) \right. \\ \left. - f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1)=\omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0)=\pi_j^i\right)\right| dV \le \frac{2cKL}{D}. \tag{12}$$

According to (9), (10), (11), and (12), we finally induce:

$$SD(\hat{z}_0, \hat{z}_1) \le \frac{1}{2cK!\binom{N}{cK}} \sum_{i=1}^{\binom{N}{cK}} \sum_{j=1}^{cK!} \frac{2cKL}{D} = \frac{cKL}{D}. \tag{13}$$

We complete this proof. □

Following (13), by setting a constant c, $L = poly(\kappa)$, and $D = 2^{\kappa}$, and letting $\mathsf{negl}(\kappa) = \frac{cKL}{D}$, CSEiC meets the security claimed in Definition 1.

So far, if $\{i \mid (Px)[i] \neq 0\} \subset S$ (Namely, perfect support detection) holds, the proposed system is secure. Next, we discuss and derive the sufficient condition of perfect support detection of Px. We extend Lemma 3 to the following lemma.

Lemma 4: Let I_1 and $I_2 \subset \Omega$ be two sets with $|I_1 \cap I_2| \le \gamma$, $\gamma \in [0, K-1]$. If $\delta_{|I_1|+|I_2|} < 1$, then

$$\|(A_{I_1})^T A_{I_2} x\|_2^2 \\ \le \delta_K^2 \|x_V\|_2^2 + \left(1+\delta_{|U|}\right)^2 \|x_U\|_2^2 + \delta_{|I_1|+|I_2|}^2 \|x\|_2^2 \tag{14}$$

holds for any x, where U is the set collecting indices of the first γ largest entries of $abs(x)$ and $V = I_2 \setminus U$.

Proof: We start from the case $|I_1 \cap I_2| = c$ and $|U| = c$ with $c \le \gamma$. Let U' and V' satisfy $I_1 \cap I_2 = U'$ with $|U'| \le c$ and $V' = I_2 \setminus U'$. Then, we have:

$$\begin{aligned} &\|(A_{I_1})^T A_{I_2} x\|_2^2 \\ &= \|(A_{U'})^T A_{I_2} x\|_2^2 + \|(A_{I_1 \setminus U'})^T A_{I_2} x\|_2^2 \\ &\le \|(A_{U'})^T A_{I_2} x\|_2^2 + \delta_{|I_1|+|I_2|-c}^2 \|x\|_2^2 \\ &\le \|(A_{U'})^T A_{V'} x_{V'}\|_2^2 + \|(A_{U'})^T A_{U'} x_{U'}\|_2^2 + \delta_{|I_1|+|I_2|}^2 \|x\|_2^2 \\ &\le \delta_{|I_2|}^2 \|x_{V'}\|_2^2 + (1+\delta_c)^2 \|x_{U'}\|_2^2 + \delta_{|I_1|+|I_2|}^2 \|x\|_2^2 \\ &\le \delta_K^2 \|x_V\|_2^2 + (1+\delta_c)^2 \|x_U\|_2^2 + \delta_{|I_1|+|I_2|}^2 \|x\|_2^2. \end{aligned} \tag{15}$$

Since $\delta_{|I_2|}^2 \le (1+\delta_c)^2$ holds for any U' and V', $c = \gamma$ will lead to the largest upper bound of $\|(A_{I_1})^T A_{I_2} x\|_2^2$. Note that the last inequality in (15) holds because $|U'| \le |U|$ and U is the set collecting indices of the first γ largest entries of $abs(x)$. □

Theorem 2: Let $x \in \mathbb{R}^N$ be K-sparse, let $A \in \mathbb{R}^{M \times N}$ with $y = Ax$, let U be the set collecting indices of the first γ largest entries of $abs(x)$, let S be a set collecting indices of first ρ entries of $abs(A^T y)$, let $\tau = \min(\rho, K)$, and let $T = \{i \mid x[i] \neq 0\}$ be the ground truth. Suppose A satisfies K-RIP with δ_K. For any $\gamma \in [0, K-1]$, if

$$(2K-\tau)\delta_{\rho+K}^2 + 2\left(\tau + K\frac{\|x_U\|_2^2}{\|x_T\|_2^2}\right)\delta_{\rho+K} < \tau - K\frac{\|x_U\|_2^2}{\|x_T\|_2^2},$$

then $|S \cap T| \ge \gamma + 1$.

Proof: Let A_S be the submatrix formed by the columns of A with indices belonging to S, and let a_i be i'th column of A. Based on Step 5 in Algorithm 2, S has the following property:

$$\|A_S^T y\|_2 = \max_{|I|=\tau} \sqrt{\sum_{i \in I} |a_i^T y|^2}.$$

Then, we can derive:

$$\begin{aligned} \frac{1}{\tau}\|A_S^T y\|_2^2 &= \max_{|I|=\tau} \frac{1}{\tau} \sum_{i \in I} |a_i^T y|^2 \ge \frac{1}{|T|} \sum_{i \in T} |a_i^T y|^2 \\ &= \frac{1}{K}\|A_T^T y\|_2^2 = \frac{1}{K}\|A_T^T A_T x_T\|_2^2. \end{aligned} \tag{16}$$

By Lemma 1, we have

$$\|A_S^T y\|_2^2 \ge \frac{\tau}{K}\|(A_T)^T A_T x_T\|_2^2 \ge \frac{\tau}{K}(1-\delta_K)^2 \|x_T\|_2^2. \tag{17}$$

On the other hand, when the chosen support includes partially correct indices (*i.e.*, $|S \cap T = U| \le \gamma$), we have

$$\begin{aligned} \|A_S^T y\|_2^2 &= \|(A_S)^T A_T x_T\|_2^2 \\ &\le \delta_K^2 \|x_V\|_2^2 + \left(1+\delta_\gamma\right)^2 \|x_U\|_2^2 + \delta_{\rho+K}^2 \|x_T\|_2^2, \end{aligned} \tag{18}$$

where $V = T \setminus U$ and the inequality follows from Lemma 4. This inequality contradicts (17) if

$$\begin{aligned} &\delta_K^2 \|x_V\|_2^2 + \left(1+\delta_\gamma\right)^2 \|x_U\|_2^2 + \delta_{\rho+K}^2 \|x_T\|_2^2 \\ &\quad < \frac{\tau}{K}(1-\delta_K)^2 \|x_T\|_2^2 \\ &\Rightarrow \delta_{\rho+K}^2 \|x_V\|_2^2 + \left(1+\delta_{\rho+K}\right)^2 \|x_U\|_2^2 + \delta_{\rho+K}^2 \|x_T\|_2^2 \\ &\quad < \frac{\tau}{K}(1-\delta_{\rho+K})^2 \|x_T\|_2^2 \\ &\Rightarrow (2K-\tau)\delta_{\rho+K}^2 + 2\left(\tau + K\frac{\|x_U\|_2^2}{\|x_T\|_2^2}\right)\delta_{\rho+K} \\ &\quad < \tau - K\frac{\|x_U\|_2^2}{\|x_T\|_2^2} \end{aligned}$$

We complete this proof. □

By Theorem 2, letting $\gamma = K-1$, we can induce the following corollary for perfect support detection.

Corollary 1: Following the assumption and notation in Theorem 2, let $\rho = 2K$, $\gamma = K-1$, and $\eta = \min_{i \in T} \frac{x[i]^2}{\|x\|_2^2}$, if

$$\delta_{3K} < \frac{\eta}{5},$$

then $T \subset S$.

From Corollary 1, we know that a random matrix satisfies $\delta_{3K} < \frac{\eta}{5}$ with high probability, provided one chooses $M = O\left(\frac{K}{\eta^2}\log\frac{N}{K}\right)$ [27]. Thus, we can achieve security with a larger number of measurements than $M = O\left(K\log\frac{N}{K}\right)$, which is required for perfect recovery in CS. To guarantee perfect support detection, we can sense x with more measurements but only transmit M measurements to the cloud without needing extra computation cost on the sensor. For example, if $A \in \mathbb{R}^{\mathbb{N}\times\mathbb{N}}$ is a circulant matrix or discrete Fourier transform matrix, N measurement can be sensed within $O(N \log N)$ operations via fast Fourier transform (FFT). Nevertheless, this strategy requires the sampling rate on the sensor meet the Nyquist rate. Moreover, the cloud still requires more measurements due to $2K$-sparse ciphertext x'. We further study how to reduce the number of measurements M in Sec. IV-C.

a: Remarks

In summary, compared with Wang *et al.*'s system [17], even though the cloud side (which also can be treated as a semi-honest adversary) is allowed to probe information based on the knowledge of A, y', and $\hat{z}$, our system still is secure. In contrast, Wang *et al.* tried to hide information of Px in terms of one-to-one linear transformation. In essence, however, solving linear programming is equivalent to reconstructing Px. In other words, the security of Wang *et al.*'s system, in fact, is not as high as their claim that any of two ciphertexts are indistinguishable in terms of statistical distance. We show that their system can only protect information about the positions of non-zero entries of data under certain attacks, as formally proved in Appendix. Nevertheless, CSEiC encrypts x by actually adding a random vector to itself, as shown in (4), such that the cloud no longer aims at reconstructing Px. This exactly states the main difference between CSEiC and [17], and why the former possesses higher security.

C. REDUCTION OF MEASUREMENTS

Recall that our method works well, depending on perfect reconstruction of x' on the cloud, as discussed in Sec. IV-B. In this section, we discuss how to reduce the number M of measurements under fixed reconstruction quality on the cloud, without losing any security.

Recall that the cloud will obtain $\hat{z} = x'$. Thus, the cloud can obtain the support set S of x'. Since the information about S is included in x', it means

$$SD\left((\hat{z}_0, S_0), (\hat{z_1}, S_1)\right) = SD\left(\hat{z}_0, \hat{z_1}\right).$$

Since we have proved that our system is secure without revealing $\hat{z}$ in Theorem 1, if the sensor sends S to the cloud, the proposed method is still secure.

A fascinating thing is that Wang and Yin [29] proposed a CS decoder where, with the help of the support set S, solving weighted ℓ_1-minimization instead of ℓ_1-minimization is effective to improve the performance. Thus, we can change the decoder procedure on the cloud as:

$$\hat{x} = \arg\min_{\bar{x}} \|W\bar{x}\|_{\ell_1} \quad s.t.\ \|y - A\bar{x}\|_{\ell_2} \leq \epsilon, \tag{19}$$

where W is a diagonal matrix with $W[i,i] = 0$ for $i \in S$ and $W[i,i] = 1$ for $i \notin S$.

We will demonstrate in the simulations that this decoder can reduce the number of measurements while maintaining reconstruction quality.

D. EXTENSION TO DATA BEING SPARSE IN TRANSFORMED DOMAIN

Up to now, previous discussions have been based on the assumption that x is exactly K-sparse in the time domain. Nevertheless, most natural data may not be sparse in the time/spatial domain. On the contrary, if x exhibits certain sparsity in a transformed domain Ψ such that $s = \Psi^T x$, then we can simply modify Step 3 in Algorithm 2 as:

$$y = AP\Psi^T x, \tag{20}$$

and modify Step 4 in Algorithm 3 as

$$z = \Psi P^{-1}\left(\hat{z} + \Sigma P u'\right). \tag{21}$$

Under this circumstance, our system now is considered as encrypting K-sparse vector s by keeping the first K largest coefficients instead of x.

In compressive sensing, there exist many studies [30]–[32], called "dictionary learning," that aim to find Ψ to achieve signal sparsity in a transformed domain. In addition to dictionary learning, it is also well-known that transforms like discrete cosine transform (DCT) or discrete wavelet transform (DWT) can sparsely represent signals well and are commonly adopted. Note that the reconstruction quality is related to RIP of A (neither Ψ nor $A\Psi$) because signals are transformed into the corresponding coefficients in the sensor such that the cloud still uses A (not Ψ nor $A\Psi$) for ℓ_1-minimization. In practice, A can be chosen as a Gaussian random matrix, which has already been proved to satisfy RIP with high probability [33].

E. COMPUTATION COMPLEXITY AND COMMUNICATION COST

We mainly focus on the computation and communication cost [34] on the sensor and user sides since the cloud does not necessarily care about energy consumption.

On the sensor side, Step 7 of Algorithm 2 dominates the whole cost. Specifically, it requires $O(MN)$ operations to multiply an $N \times 1$ vector by an $M \times N$ sensing matrix A. As mentioned in Sec. IV-B, there are existing works about fast sensing matrix design [35], [36] such that the cost is reduced to $O(N \log N)$. If data are sparse in a transformed domain, we require extra computation cost of matrix-by-vector multiplications $\Psi^T x$ in (20). In the worst case, $\Psi^T x$ requires $O(N^2)$. Nevertheless, transformed domains like discrete cosine transform (DCT) and wavelet transform can be speeded up such that Ψx costs $O(N \log N)$ operations. In addition, the communication cost for y' is $O(M)$. If we want to send S to cloud, it costs extra $O(K)$. Since $K \leq M$, the total cost is still $O(M)$.

On the user side, the cost is dominated by Step 4 of Algorithm 4. Since P is a permutation matrix, P^{-1} can be computed in $O(N)$ operations and $\hat{z} + \Sigma P u'$ costs $O(N)$ operations. In addition, if data are sparse in a transformed domain, it requires multiplying a matrix Ψ by a vector $P^{-1}\left(\hat{z} + \Sigma P u'\right)$. The computation cost depends on Ψ, as discussed in the last paragraph.

TABLE 1. Comparison with existing system.

	Wang [17]	Our system
Cloud storage cost (in Bytes)	$O(2N)$	$O(N)$
Sensor bandwidth (in Hz)	$O(M)$	$O(M)$
Sensor computation (in Operations)	$O(MN)$	$O(MN) + [\Psi]$
User bandwidth (in Hz)	$O(MN)$	Zero
User computation (in Operations)	$O(N^{\theta})$ $2 < \theta < 3$	$O(N) + [\Psi]$
Privacy protection	Positions only	Positions and values

Finally, ℓ_1-minimization on the cloud side requires $O(N^3)$ operations.[2] Thus, the cloud burdens the main computation cost.

Table 1 shows the comparison with [17], where $[\Psi]$ represents the computation cost of matrix-by-vector multiplications. Note that only the cloud requires storing the ciphertext for responding the query from user. The "User bandwidth" denotes the number of bits sent from the user to the cloud per data x. In contrast, Wang *et al.*'s system [17] requires that the user upload an $M \times N$ decoding matrix to cloud. Also note that, since the measurement vector y' of data is, in fact, required for participating in generating the decoding matrix (see Appendix), user cannot upload the decoding matrix to cloud before querying. The result is that Wang *et al.*'s system must wait for transmission of the decoding matrix once the user delivers a query. In addition, it may be too heavy to be affordable since the user usually is assumed to have upload speed slower than download speed. If user bandwidth is zero, it means that the cloud does not need to wait for the data sent from the user.

As for user computation, Wang *et al.*'s system requires $O(N^{\theta})$ with $2 < \theta < 3$ due to matrix-by-matrix multiplications. Nevertheless, the cost of our system is bounded by $O(N) + [\Psi]$, which is faster than Wang *et al.*'s system.

V. SIMULATIONS

The simulations were conducted in a MATLAB environment with an Intel CPU Q6600 at 3.40 GHz and 4 GB RAM under Microsoft Win7 (32 bits). We used CVX package for implementing ℓ_1-minimization. On the cloud, Wang *et al.*'s method [17] used linear programming, which has the same performance as ℓ_1-minimization but suffers higher computation cost due to complex constraints for encryption, as described in Appendix. Thus, we simply consider ℓ_1-minimization as the baseline for comparison here.

In our settings, all test images were with size of 640×640 (for "Brain") or 640×640 (for "Lena"). They were divided into 32×32 blocks, where each block was sensed, encrypted, and decrypted independently because the original images are too large to be efficiently dealt with. The sensing matrix A was drawn from i.i.d. normal variables with $\mathbb{N}\left(0, \frac{1}{M}\right)$. In addition, considering images are usually not sparse in the time domain, we sparsify all blocks of images by discrete cosine transform (DCT).

Three experiments were conducted. First, we validate the effectiveness of weighted ℓ_1-minimization and support detection. Second, since our method encrypts the plaintext by actually modifying its values such that the reconstruction quality is changed accordingly, we compare the reconstruction quality between the proposed method and the baseline under the same set of parameters, including N, M, and K. Meanwhile, we show the corresponding encrypted images that illustrate the effectiveness of privacy guarantee. Third, we show the computation costs of encryption and decryption on the sensor, cloud, and user sides, respectively. Furthermore, because we require doing the support detection on the sensor side, we need to decide K. Since the natural images are only approximately K-sparse in the DCT domain, we merely set $K = \frac{M}{16}$ in the second and third experiments.

A. EFFECTIVENESS OF WEIGHTED ℓ_1-MINIMIZATION AND SUPPORT DETECTION

This experiment was conducted with $N = 200$, $K = 1, \ldots, 100$, and $M = 1, \ldots, 100$ via the following procedure.

1) Construct K-sparse binary signal x and $A \in \mathbb{R}^{M \times N}$.
2) Obtain $y = APx$, $y' = Ax'$, and S via Algorithm 2 with $x' = Px - \Sigma Pu$ in (3).
3) For signal recovery, we explore three kinds of decoders:
   a) (Decoder 1) Given y, run ℓ_1-minimization to output $\hat{x}$ by (1).
   b) (Decoder 2) Given y', run ℓ_1-minimization to output $\hat{z}$ by (1).
   c) (Decoder 3) Given y' and S, run weighted ℓ_1-minimization to output $\hat{z}$ by (19).
4) Declare success if $\|\hat{x} - x\|_2 \leq 10^{-5}$ for Decoder 1 or $\|\hat{z} - x'\|_2 \leq 10^{-5}$ for Decoders 2 and 3.

It should be noted that Decoder 1 is considered as a baseline to reconstruct K-sparse vectors via ℓ_1-minimization but Decoders 2 and 3 require reconstructing $2K$-sparse vectors because of encryption in our system. Fig. 2 shows the successful probabilities of three decoders, respectively. Although Decoder 1 in Fig. 2(a) outperforms the other two under fixed K and an increase of $\frac{M}{N}$, CS usually adopts smaller measurement rates (*e.g.*, $\frac{M}{N} \leq 0.25$ ($M \leq 50$ in this case)) for real applications. Under this circumstance, Decoder 3 in Fig. 2(c) exhibits comparable performance with Decoder 1. As expected, Decoder 2 gets the worst performance due to $2K$ sparsity.

Finally, Fig. 2(d) shows the successful probability of perfect support detection with $\{i| \, (Px) \, [i] \neq 0\} \subset S$. The results reveal the number of measurements required for support detection is larger than that of perfect reconstruction in Fig. 2(c). Since both perfect support detection and perfect reconstruction must succeed in our system, support detection becomes the bottleneck of requiring more measurements. This problem, however, can be overcome by sensing

[2] The computation complexities in CS depend on different sparse recovery algorithms. $O(N^3)$ mentioned here is just for Basis Pursuit [25], [37].

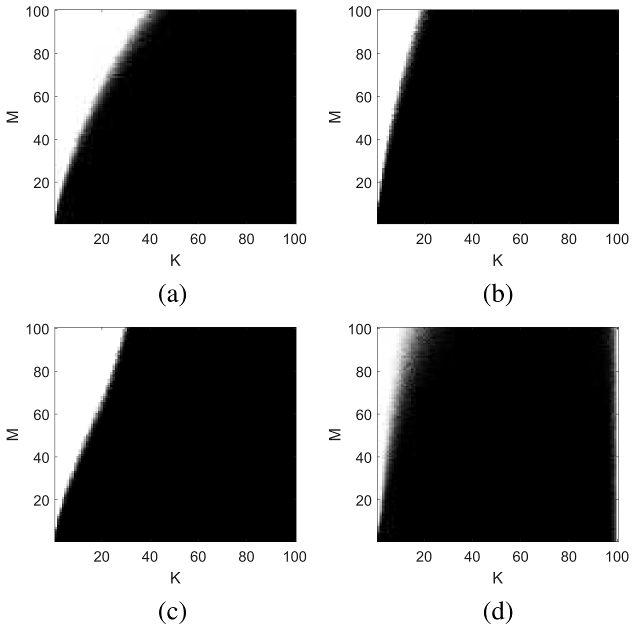

FIGURE 2. Probability of perfect reconstruction for (a)(b)(c) under $N = 200$, where the white region denotes prob. 1 and the black region means prob. 0. (a) ℓ_1-minimization for K-sparse signal; (b) ℓ_1-minimization for $2K$-sparse signal; and (c) weighted ℓ_1-minimization for $2K$-sparse signal. (d) probability of perfect support detection.

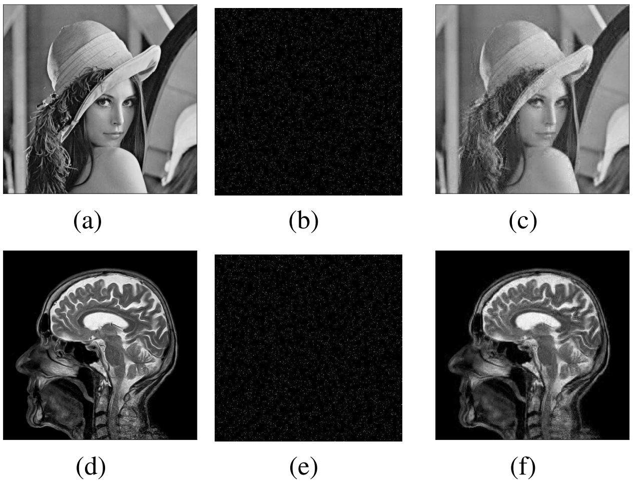

FIGURE 3. Illustration of effectiveness for privacy guarantee for two test images under $\frac{M}{N} = 0.25$. (a)(d): Sensed images on the sensor; (b)(e) Encrypted images on the cloud; (c)(f) Decrypted images on the user.

more measurements but only transmitting M measurements, as discussed in Sec. IV-B.

B. VALIDATION FOR REAL IMAGES

In this section, we focus on the reconstruction quality comparison between the baseline and our system. Fig. 3 demonstrates the effectiveness of privacy guarantee for two images with $\frac{M}{N} = 0.25$. Specifically, Fig. 3(a) and Fig. 3(d) show the original images sampled on the sensor side. Fig. 3(b) and Fig. 3(e) show the corresponding encrypted images on the cloud side, and Fig. 3(c) and Fig. 3(f) show the corresponding reconstructed and decrypted image on the user side.

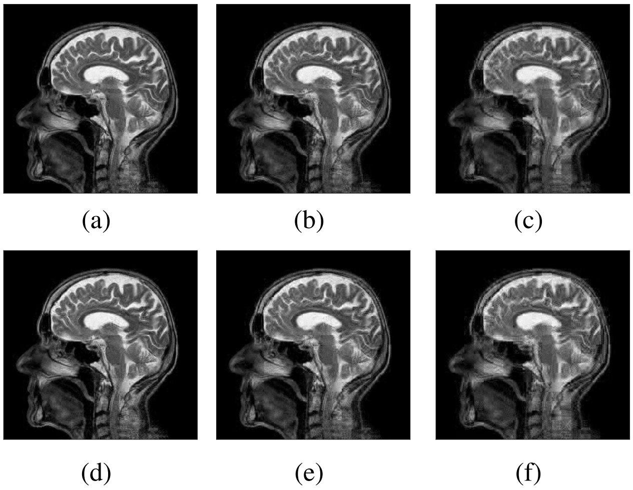

FIGURE 4. Comparison between Wang *et al.*'s system ((a)~(c)) and our proposed scheme ((d)~(f)) in terms of reconstructed images in PSNR for baseline with measurement rates $\frac{M}{N} = 0.25$, 0.1875, and 0.125, respectively.

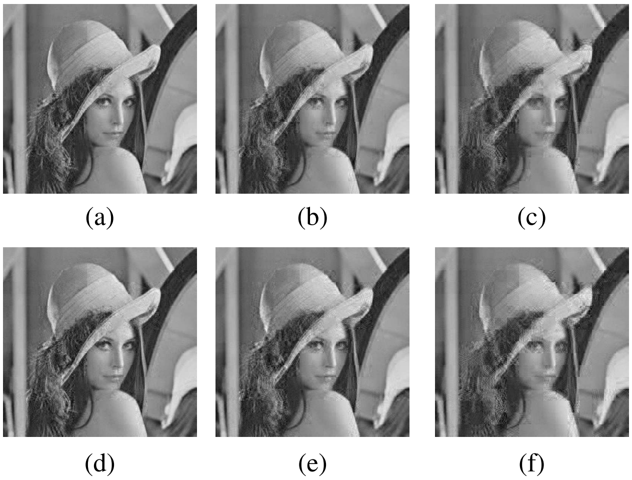

FIGURE 5. The same setting as in Fig. 4.

It is interesting and important to note that the top-left areas of Fig. 3(d) have pixels of zero gray-level. Nevertheless, the corresponding encrypted blocks in Fig. 3(e) still are indistinguishable from other blocks originally with non-zero values. This illustrates that the proposed system actually achieves the security described in Definition 1.

As for reconstruction quality, which is estimated by peak signal-to-noise ratio (PSNR), under a range of measurement rates, experimental results indicate that our system does not obviously degrade the visual quality (less than 0.6dB degradation, compared with the baseline) for different images. More specifically, Fig. 4 and Fig. 5 demonstrate the comparison between our system and Wang *et al.*'s system [17] in terms of visual reconstruction quality. It is concluded that our method exhibits comparable visual results with

Wang *et al.*'s system but benefits from other properties discussed in Sec. IV-E.

On the other hand, since wireless communication usually suffers from additive Gaussian noise, the Step 7 in ECS (Algorithm 2) is modified as:

$$y' = y - A\Sigma u' + e,$$

where e is an additive white Gaussian noise with zero mean and variance σ_n. We verify the proposed algorithm under $\sigma_n = 5, 10, 25$ and show the results in Figs. 6 and 7. It is observed that both our system and Wang *et al.*'s system can be said to be robust against noises because the PSNR values in Figs. 6 and 7 are slightly lower than the corresponding ones in Figs. 4 and 5.

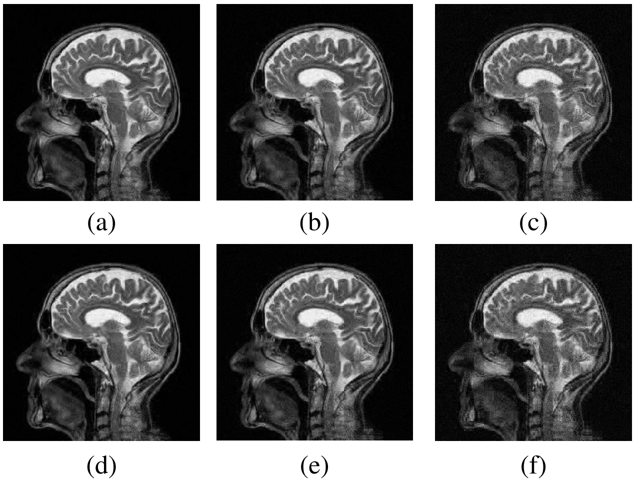
(a) (b) (c)
(d) (e) (f)

FIGURE 6. Comparison between Wang *et al.*'s system ((a)~(c)) and our proposed scheme ((d)~(f)) under the noisy interference with measurement rates $\frac{M}{N} = 0.25$. (a)(d) $\sigma_n = 5$. (b)(e) $\sigma_n = 10$. (c)(f) $\sigma_n = 20$.

C. EFFICIENCY EVALUATION

In this section, we follow the same comparison found in [17]. Specifically, we compare the computation cost of ℓ_1-minimization without cloud assistance. Thus, the user needs to burden the cost of ℓ_1-minimization. In contrast, our proposed system only needs to consider the sensor cost and user cost because we do not worry about the resourceful cloud. Therefore, we use the criterion "Asymmetric Speedup," [17] by dividing the cost in conventional system by that in our system, to evaluate computational savings.

Table 2 shows the results. Our proposed system always achieves around 245× speedup. In fact, it can be expected that, if the block size increases, the speedup becomes higher too. The results confirm that the cloud actually burdens the main overhead more than the other two parties. In addition, two different images, under the same measurement rate, share nearly asymmetric speedup because each block has the same fixed size, implying the need of the same computation cost.

On the contrary, Wang *et al.*'s system achieves 7× speedup (see [17, Table 2]) because the user side requires the matrix-by-matrix multiplication, costing $O(N^{\theta})$ with $2 < \theta < 3$. In contrast, our method costs $O(N) + [\Psi]$ operations, where Ψ is a DCT matrix that can be efficiently speeded up via fast Fourier transform (FFT). Thus, in terms of the computation cost on the user side, our system spends lower than Wang *et al.*'s system.

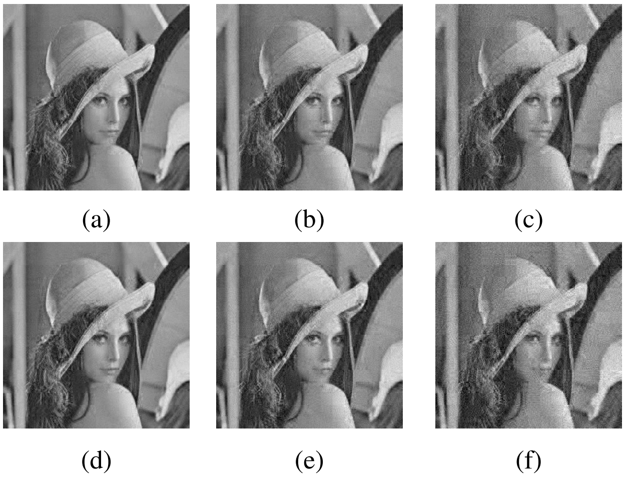
(a) (b) (c)
(d) (e) (f)

FIGURE 7. The same setting as in Fig. 6.

TABLE 2. Comparison of computation cost under different measurement rates, where the 3*rd* − 5*th* rows show results for "Lena" and the 6*th* − 8*th* rows are for "Brain".

$\frac{M}{N}$	Conv. System	Proposed System		Speedup
	t_{ori} (sec)	t_{sensor} (sec)	t_{user} (sec)	$\frac{t_{ori}}{t_{sensor}+t_{user}}$
0.25	1183.1	1.52	0.36	629 ×
0.1875	641.3	1.23	0.38	398 ×
0.125	324.7	0.92	0.37	251 ×
0.25	1738.6	2.2	0.48	648 ×
0.1875	1022.9	1.91	0.45	433 ×
0.125	525.3	1.58	0.46	245 ×

VI. CONCLUSIONS

In this paper, we have presented a cloud-assisted compressive sensing-based data gathering system. We show that ciphertexts on the cloud side are statistically indistinguishable to achieve privacy assurance. Our method is also cost-effective in terms of communication between the user and cloud such that, once the cloud receives the encrypted data from sensor, it can immediately carry out the intended task. We further show that ℓ_1-minimization with the information of support set can maintain the reconstruction quality without breaching security. We provide both theoretical analyses and empirical results to verify our system.

APPENDIX

ATTACK ON WANG *et al.*'s Method

In [17], the main idea is to replace ℓ_1-minimization for sparse signal recovery with transformed linear programming (LP), resulting in the reconstructed signal (ciphertext) on the cloud side approximating a Gaussian random signal for protection of the corresponding plaintext. Due to the transformation, the length of ciphertext on the cloud becomes $2N$ instead of N. In addition, the authors assume that the cloud is a semi-honest adversary. Here, we present an attack on Wang *et al.*'s system to prove that they cannot achieve the security they claimed.

First, we introduce Wang *et al.*'s security definition as follows.

Definition 2 (Security Definition on [17, P. 4]): Wang *et al.*'s transform scheme is κ-secure if, over the random choice, the secret key sk with the security parameter κ satisfies:

$$\forall \Phi_0, \Phi_1 \in \mathcal{S} : SD(\mathsf{Trans}(\mathsf{sk}, \Phi_0), \mathsf{Trans}(\mathsf{sk}, \Phi_1)) \le \mathsf{negl}(\kappa),$$

where $\mathcal{S}$ denotes the set of all the LP problems and Trans is a query transformation algorithm that takes as input the secret key sk and the original LP problem Φ and outputs the transformed problem $\Phi_b^{\mathsf{sk}} = \mathsf{Trans}(\mathsf{sk}, \Phi_b)$ for $b \in \{0, 1\}$.

Then, the notations and assumptions in Wang *et al.*'s system are defined as follows (some notations are redefined and only used in this section).

Remarks, notations, and assumptions:

- x is supposed to be K-sparse with $K < M < N$ to meet the requirements in CS and $x[i] \in [-L, L]$ for all i's with $L = poly(\kappa)$.
- $A \xleftarrow{\$} \mathbb{N}(0, \frac{1}{M})^{M \times N}$.
- f is a measurement vector with $f = Ax$.
- $Q \in \mathbb{R}^{M \times M}$ is a random invertible matrix.
- $1 \in \mathbb{R}^{2N}$ is a one vector.
- $W \in \mathbb{R}^{2N \times 2N}$ such that $1^T W = 1^T$.
- $D \in \mathbb{R}^{2N \times 2N}$ is a generalization permutation matrix and has positive non-zero elements.
- $r \xleftarrow{\$} \mathbb{U}(2^\kappa)^{2N}$.

Second, we describe the protocol of Wang *et al.*'s system. On the sensor side, the original signal $x \in \mathbb{R}^N$ is sampled via $f = Ax$ and $\hat{f} = Qf$ is transmitted to cloud. The user also transmits two matrices $\hat{\Lambda}$ and $\hat{D}$, where $\hat{\Lambda} = Q\Lambda W$, $\Lambda = [A, -A]$, $\hat{D} = DW - \lambda\Lambda W$, $\lambda \in \mathbb{R}^{2N \times M}$ is a matrix such that $Dr - \lambda(f + \Lambda r) = 0$. Note that λ is dependent on f, implying λ must be generated after the user has received $\hat{f}$ from the cloud. This is the reason we claim Wang *et al.*'s system cannot avoid the delay of recovery in Sec. IV-E because it fails to transmit $\hat{D}$ in advance.

After receiving the data from both the sensor and user, the cloud can solve linear programming (LP) as:

$$\min_z \hat{c}z \quad s.t.\ \hat{\Lambda}z = \hat{f},\ \hat{D}z \ge 0, \tag{22}$$

where $\hat{c} = 1^T W$. The solver will return the ciphertext z with $x' = Wz - r$, where $x' = \left[x^+, x^-\right]$, $x^+[i] = \max(x[i], 0)$, and $x^-[i] = \max(-x[i], 0)$ for all i's.

To achieve the security defined above, the authors show the following theorem.

Theorem 3 [17, Th. 4.1]: If we pick random vectors r and r^*, where each entry in r and r^* is sampled from the uniform distribution with range $[-2^\kappa, 2^\kappa]$, then the statistical distance (SD) satisfies:

$$SD(x' + r, r^*) \le \mathsf{negl}(\kappa),$$

where $\mathsf{negl}(\kappa)$ is a negligible function.

By Theorem 3, both $z = W^{-1}\left(x' + r\right)$ and $z^* = W^{-1}r^*$ are also statistically indistinguishable such that $SD(z, z^*) \le \mathsf{negl}(\kappa)$. Since z and z^* are indistinguishable on the cloud, if we switch z with z^*, then z does not reveal x'. In sum, by this idea, they prove that, for any two transformed LP problems Φ_0^{sk} and Φ_1^{sk} solved via (22), we get:

$$SD(\Phi_0^{\mathsf{sk}}, \Phi_1^{\mathsf{sk}}) \le \mathsf{negl}(\kappa).$$

Finally, we propose our attack on Wang *et al.*'s system with the goal of formally proving that $SD(\Phi_0^{\mathsf{sk}}, \Phi_1^{\mathsf{sk}})$ is larger than a threshold, which means Φ_0^{sk} and Φ_1^{sk} are not statistically indistinguishable according to Theorem 3. Thus, Wang *et al.*'s system does not meet the security definition they claimed. Since the cloud is a semi-honest adversary, it can use the information owned by itself to explore the information from the ciphertext z. Recall that the cloud has $\hat{D}$, z, and $\hat{\Lambda}$. We propose an attack such that the cloud can obtain Dx' by calculating $\hat{D}z$ as:

$$\begin{aligned} \hat{D}z &= DWz - \lambda\Lambda Wz - Dr \\ &= DWz - \lambda\Lambda\left(x' + r\right) \\ &= DWz - \lambda\Lambda x' - \lambda\Lambda r \\ &= DWz - \lambda\Lambda x' - Dr\ + \lambda f \\ &= D(Wz - r) = Dx'. \end{aligned}$$

The second equality uses the property $x' + r = Wz$ and the fourth equality uses $f = \Lambda x'$ and $Dr - \lambda(f + \Lambda r) = 0$. The result implies that the cloud can know the permuted x'.

Under this attack, r is removed. Then, we prove in Theorem 4 that there exists a pair of plaintexts, x_0' and x_1', such that the corresponding ciphertexts, z_0 and z_1, are no longer statistically indistinguishable, which violates Theorem 3. Since Wang *et al.*'s system never explicitly defines D,[3] we assume that all non-zero elements of D are uniformly distributed within the range of $[0, 1]$.

Theorem 4 (Insecurity of [17] by Our Attack): Let $\Phi_b^{\mathsf{sk}} = (\hat{\Lambda}_b, \hat{D}_b, z_b, \hat{D}_b z_b)$ for $b \in \{0, 1\}$ represent all information revealed to the cloud. Suppose C_0 and C_1 are positive numbers with $C_1 > C_0$. Then, there exists a pair of K-sparse plaintexts, x_0' and x_1' with $x_b'[i] \in \{0, C_b\}$ for $b \in \{0, 1\}$, such that

$$SD(\Phi_0^{\mathsf{sk}}, \Phi_1^{\mathsf{sk}}) \ge \frac{(C_1 - C_0)}{C_1}.$$

Proof: Suppose S_b denotes the support set of permuted signal $D_b x_b'$, for $b \in \{0, 1\}$. Let $\hat{z}_b = D_b x_b'$ be considered as two multivariate random variables, where, for all non-zero entries, the positions are uniformly distributed over $[1, N]$

[3] The paper mentions "We assume D has positive non-zero elements" and "randomly choosing D".

and the values follow a uniform distribution within the range of $[0, C_b]$. The sparsity of $\hat{z}_b$ is K. Since $\hat{z}_b[i] = 0$ for all $i \in S_b$ with probability 0, we make an assumption that the range of all uniform random variables does not contain zero. Let $f_{\hat{z}_0}$ and $f_{\hat{z}_1}$ be the probability density functions (p.d.f.) of $\hat{z}_0$ and $\hat{z}_1$, respectively. Based on the same definition of statistical distance in (8), we have

$$\begin{aligned} SD(\Phi_0^{\mathsf{sk}}, \Phi_1^{\mathsf{sk}}) &= SD((\hat{\Lambda}_0, \hat{D}_0, z_0, \hat{D}_0 z_0), (\hat{\Lambda}_1, \hat{D}_1, z_1, \hat{D}_1 z_1)) \\ &\geq SD(\hat{D}_0 z_0, \hat{D}_1 z_1) \\ &= SD(D_0 x_0', D_1 x_1') \\ &= SD(\hat{z}_0, \hat{z}_1). \end{aligned}$$

For a K-sparse vector, the number of possible support sets is $\binom{2N}{K}$. Let $\omega_1, \ldots, \omega_{\binom{2N}{K}}$ be all possible sets collecting K indices from range $[1, 2, \ldots, N]$ and let $\mathsf{sp}(\hat{z})$ denote the support set of $\hat{z}$. We define

$$\Omega_i = \left\{ v \mid v \in \mathbb{R}^{2N} \text{ and } \mathsf{sp}(v) = \omega_i \right\}.$$

We have $f_{\hat{z}_0}(v) = f_{\hat{z}_1}(v) = 0$ if $v \notin \Omega_i$ for all i's. Since $\Omega_i \bigcap \Omega_j = \emptyset$ for any $i \neq j$, we can derive:

$$SD(\hat{z}_0, \hat{z}_1) = \frac{1}{2} \sum_{i=1}^{\binom{2N}{K}} \int_{\Omega_i} \left| f_{\hat{z}_0}(v) - f_{\hat{z}_1}(v) \right| dV. \tag{23}$$

In addition, we can also derive:

$$\begin{aligned} &\int_{\Omega_i} \left| f_{\hat{z}_0}(v) - f_{\hat{z}_1}(v) \right| dV \\ &\quad = \frac{1}{\binom{N}{K}} \int_{\Omega_i} \Big| f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_i\right) \\ &\quad\quad - f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_i\right) \Big| dV. \end{aligned} \tag{24}$$

We skip the detailed derivations in (24) as they are the same as (10).

Given ω_i for any i, the number of permutations for K nonzero entries is $K!$. Let $\pi_1^i, \ldots, \pi_{K!}^i$ be all possible orders and let $\mathsf{o}_{\omega_i}(\hat{z})$ output the order of $\hat{z}$. Then,

$$\begin{aligned} (24) &= \frac{1}{K!\binom{2N}{K}} \int_{\Omega_i} \Bigg| \sum_{j=1}^{K!} f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi_j^i\right) \\ &\quad - \sum_{j=1}^{K!} f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi_j^i\right) \Bigg| dV \\ &= \frac{1}{K!\binom{2N}{K}} \sum_{j=1}^{K!} \int_{\Omega_i} \Big| f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi_j^i\right) \\ &\quad - f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi_j^i\right) \Big| dV. \end{aligned} \tag{25}$$

Note $f_{\hat{z}_b}\left(v \mid \mathsf{sp}(\hat{z}_b) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_b) = \pi_j^i\right)$'s, for $b = 0, 1$, have the same distribution; thus, the second equality holds.

Furthermore, we can derive:

$$\begin{aligned} &\int_{\Omega_i} \Big| f_{\hat{z}_0}\left(v \mid \mathsf{sp}(\hat{z}_0) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi_j^i\right) \\ &\quad - f_{\hat{z}_1}\left(v \mid \mathsf{sp}(\hat{z}_1) = \omega_i, \mathsf{o}_{\omega_i}(\hat{z}_0) = \pi_j^i\right) \Big| dV = \frac{2(C_1 - C_0)}{C_1}. \end{aligned} \tag{26}$$

According to (23), (24), (25), and (26), we finally induce:

$$SD(\hat{z}_0, \hat{z}_1) = \frac{1}{2K!\binom{2N}{K}} \sum_{i=1}^{\binom{2N}{K}} \sum_{j=1}^{K!} \frac{2(C_1 - C_0)}{C_1} = \frac{(C_1 - C_0)}{C_1}. \tag{27}$$

We complete this proof. □

Consequently, since two transformed LP problems are not statistically indistinguishable, the original objective of protecting data by LP fails. In addition, to solve LP, the user needs to transmit extra information to cloud, consuming extra communication cost. The cloud also must wait until the data transmitted from the user have been completely received.

REFERENCES

[1] M. Leinonen, M. Codreanu, and M. Juntti, "Distributed correlated data gathering in wireless sensor networks via compressed sensing," in *Proc. Asilomar Conf. Signals, Syst. Comput.*, 2013, pp. 418–422.

[2] S. Padalkar, A. Korlekar, and U. Pacharaney, "Data gathering in wireless sensor network for energy efficiency with and without compressive sensing at sensor node," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Apr. 2016, pp. 1356–1359.

[3] J. Ko, C. Lu, M. B. Srivastava, J. A. Stankovic, A. Terzis, and M. Welsh, "Wireless sensor networks for healthcare," *Proc. IEEE*, vol. 98, no. 11, pp. 1947–1960, Nov. 2010.

[4] C. Wenjie, C. Lifeng, C. Zhanglong, and T. Shiliang, "A realtime dynamic traffic control system based on wireless sensor network," in *Proc. Int. Conf. Parallel Process. Workshops (ICPPW)*, 2005, pp. 258–264.

[5] N. Javaid, S. Faisal, Z. A. Khan, D. Nayab, and M. Zahid, "Measuring fatigue of soldiers in wireless body area sensor networks," in *Proc. 8th Int. Conf. Broadband Wireless Comput., Commun. Appl.*, 2013, pp. 227–231.

[6] N. Kimura and S. Latifi, "A survey on data compression in wireless sensor networks," in *Proc. Int. Conf. Inf. Technol., Coding Comput. (ITCC)*, 2005, pp. 8–13.

[7] T. Srisooksai, K. Keamarungsi, P. Lamsrichan, and K. Araki, "Practical data compression in wireless sensor networks: A survey," *J. Netw. Comput. Appl.*, vol. 35, no. 1, pp. 37–59, Jan. 2012.

[8] Y. Zhang, D. Xiao, W. Wen, H. Nan, and M. Su, "Secure binary arithmetic coding based on digitalized modified logistic map and linear feedback shift register," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 27, nos. 1–3, pp. 22–29, 2015.

[9] Y. Zhang, D. Xiao, K.-W. Wong, J. Zhou, S. Bai, and M. Su, "Perturbation meets key-based interval splitting arithmetic coding: Security enhancement and chaos generalization," *Secur. Commun. Netw.*, vol. 9, no. 1, pp. 43–53, 2016.

[10] Y. Rachlin and D. Baron, "The secrecy of compressed sensing measurements," in *Proc. 46th Annu. Allerton Conf. Commun., Control, Comput.*, 2008, pp. 813–817.

[11] 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.

[12] J. Qi, X. Hu, Y. Ma, and Y. Sun, "A hybrid security and compressive sensing-based sensor data gathering scheme," *IEEE Access*, vol. 3, pp. 718–724, 2015.

[13] S. Qi, Z. Li, and Y. Liu, "Achieving private, scalable, and precise data collection in wireless sensor networks," in *Proc. IEEE Int. Conf. Parallel Distrib. Syst.*, Dec. 2012, pp. 14–21.

[14] K. Xie *et al.*, "An efficient privacy-preserving compressive data gathering scheme in WSNs," *Inf. Sci.*, vol. 390, pp. 82–94, Jun. 2017.

[15] P. Hu, K. Xing, X. Cheng, H. Wei, and H. Zhu, "Information leaks out: Attacks and countermeasures on compressive data gathering in wireless sensor networks," in *Proc. IEEE INFOCOM*, May 2014, pp. 1258–1266.

[16] C. Wang, B. Zhang, K. Ren, and J. M. Roveda, "Privacy-assured outsourcing of image reconstruction service in cloud," *IEEE Trans. Emerg. Topics Comput.*, vol. 1, no. 1, pp. 166–177, Jun. 2013.

[17] C. Wang, B. Zhang, K. Ren, J. M. Roveda, C. W. Chen, and Z. Xu, "A privacy-aware cloud-assisted healthcare monitoring system via compressive sensing," in *Proc. IEEE INFOCOM*, Apr. 2014, pp. 2130–2138.

[18] T.-H. Hung, S.-H. Hsieh, and C.-S. Lu, "Privacy-preserving data collection and recovery of compressive sensing," in *Proc. IEEE ChinaSIP*, Jul. 2015, pp. 473–477.

[19] Y. Zhang *et al.*, "Low-cost and confidentiality-preserving data acquisition for Internet of multimedia things," *IEEE Internet Things J.*, to be published.

[20] W. Xue, C. Luo, G. Lan, R. Rana, W. Hu, and A. Seneviratne, "Kryptein: A compressive-sensing-based encryption scheme for the Internet of Things," in *Proc. Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2017, pp. 169–180.

[21] Y. Zhang, J. Zhou, L. Y. Zhang, F. Chen, and X. Lei, "Support-set-assured parallel outsourcing of sparse reconstruction service for compressive sensing in multi-clouds," in *Proc. Int. Symp. Secur. Privacy Social Netw. Big Data (SocialSec)*, 2015, pp. 1–6.

[22] Y. Zhang, H. Huang, Y. Xiang, L. Y. Zhang, and X. He, "Harnessing the hybrid cloud for secure big image data service," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1380–1388, Oct. 2017.

[23] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[24] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[25] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1999.

[26] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comp. Rendus Math.*, vol. 346, nos. 9–10, pp. 589–592, May 2008.

[27] P. Jain, A. Tewari, and I. S. Dhillon, "Orthogonal matching pursuit with replacement," in *Proc. Neural Inf. Process. Syst.*, 2011, pp. 1215–1223.

[28] O. Goldreich, *Foundations of Cryptography*, vol. 1. Cambridge, U.K.: Cambridge Univ. Press, 2001.

[29] Y. Wang and W. Yin, "Sparse signal reconstruction via iterative support detection," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 462–491, 2010.

[30] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[31] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 689–696.

[32] S. Ravishankar and Y. Bresler, "Learning sparsifying transforms," *IEEE Trans. Signal Process.*, vol. 61, no. 5, pp. 1072–1086, Mar. 2013.

[33] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approx.*, vol. 28, no. 3, pp. 253–263, Dec. 2008.

[34] M. Shoaib, N. K. Jha, and N. Verma, "A compressed-domain processor for seizure detection to simultaneously reduce computation and communication energy," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2012, pp. 1–4.

[35] T. T. Do, L. Gan, N. H. Nguyen, and T. D. Tran, "Fast and efficient compressive sensing using structurally random matrices," *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 139–154, Jan. 2012.

[36] F. Yu, S. Kumar, Y. Gong, and S.-F. Chang, "Circulant binary embedding," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. II-946–II-954.

[37] A. Y. Yang, A. Ganesh, Z. Zhou, S. S. Sastry, and Y. Ma. (2010). "Fast L1-minimization algorithms for robust face recognition." [Online]. Available: https://arxiv.org/abs/1007.3753

SUNG-HSIEN HSIEH received the B.S. and M.S. degrees in computer science from National Cheng Kung University, Tainan, Taiwan, in 2008 and 2010, respectively. He is currently pursuing the Ph.D. degree in communication engineering from National Taiwan University, Taipei, Taiwan. His research interests include compressive sensing, signal processing, and machine learning.

TSUNG-HSUAN HUNG received the M.S. degree in mathematical modeling and scientific computing, National Chiao Tung University, Hsinchu, Taiwan, in 2014. He is currently pursuing the Ph.D. degree with the Department of Information Engineering, National Taiwan University, Taipei, Taiwan. His research interests include cryptography, and Internet of Things security.

CHUN-SHIEN LU received the Ph.D. degree in electrical engineering from National Cheng-Kung University, Tainan, Taiwan, in 1998. He joined the Institute of Information Science, Academia Sinica, Taiwan, as a Post-Doctoral Fellow for his military service. Since 2013, he has been a Research Fellow. His current research interests include compressed sensing and sparse representation, multimedia signal processing, and security and privacy preserving in multimedia and sensor network.

Dr. Lu is currently the Deputy Director of the Research Center for Information Technology Innovation, Academia Sinica. He has been serving as a Technical Committee Member of the Multimedia Systems and Applications Technical Committee, IEEE Circuits and Systems Society, since 2007. Since 2012, he has been serving as a Technical Committee Member of the Communications and Information Systems Security (CIS-TC), IEEE Communications Society. He also serves as the Area Chair of ICASSP 2012, ICASSP 2013, ICIP 2013, and ICME 2018. He received the Ta-You Wu Memorial Award from the National Science Council in 2007 and the National Invention and Creation Award in 2004. He has been an Associate Editor of the IEEE Transactions on Image Processing since 2018.

YU-CHI CHEN received the B.S., M.S., and Ph.D. degrees from the Department of Computer Science and Engineering, National Chung-Hsing University, Taiwan, in 2008, 2009, and 2014, respectively. In 2013, he was a Visiting Scholar with the Department of Electrical Engineering, University of Washington. He was a Post-Doctoral Fellow with the Institute of Information Science, Academia Sinica, Taiwan, from 2014 to 2017. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Yuan Ze University, Taiwan. His research interests include cryptography and information security.

SOO-CHANG PEI (F'00) received the Ph.D. degree from the University of California, Santa Barbara, in 1975. He has been a Professor with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, since 1984. His current research interests include digital signal processing, image processing, optical information processing, and laser holography. He became an IEEE fellow in 2000 for his contributions to the development of digital eigenfilter designs, color image coding, and signal compressions, and to the electrical engineering education in Taiwan. He was a recipient of the Distinguished Research Award from the National Science Council from 1990 to 1998, the Academic Achievement Award in Engineering from the Ministry of Education in 1998, the Pan Wen-Yuan Distinguished Research Award in 2002, and the National Chair Professor Awards from the Ministry of Education in 2002 and 2008.