Information Sciences 387 (2017) 132–145

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A Compressive Sensing based privacy preserving outsourcing of image storage and identity authentication service in cloud

Guiqiang Hu [a], Di Xiao [a,*], Tao Xiang [a], Sen Bai [b], Yushu Zhang [c]

[a] Key Laboratory of Dependable Service Computing in Cyber Physical Society (Chongqing University) of Ministry of Education, College of Computer Science, Chongqing University, Chongqing 400044, China
[b] Department of Information Engineering, Chongqing Communication Institute, Chongqing 400035, China
[c] School of Electronics and Information Engineering, Southwest University, Chongqing 400715, China

ARTICLE INFO

Article history:
Received 21 September 2015
Revised 20 July 2016
Accepted 18 September 2016
Available online 19 September 2016

Keywords:
Compressive sensing
Cloud security
Privacy preserving
Image storage
Watermark detection in the encrypted domain

ABSTRACT

The theory of Compressive Sensing (CS) enables the compact storage of image datasets which are exponentially generated today. In this application, the high computational complexity CS reconstruction process is considered to be outsourced to the cloud for its abundant computing and storage resources. Although it is promising, how to protect data privacy and simultaneously maintain management of the image remains challenging. To address the challenge, we propose a novel outsourced image reconstruction and identity authentication service in cloud, which integrates the techniques of signal processing in the CS domain and computation outsourcing. In our system, the image CS samples are outsourced to cloud for reduced storage. For privacy, the scheme ensures the cloud to securely reconstruct image without revealing the underlying content. For management, whether the cloud determines to supply the reconstruction service is depending on the identity authentication result. Theoretical analysis and empirical evaluations show a satisfactory security performance and low computational complexity of the proposed system. Besides, experimental results also confirm the feasibility of identity authentication in the CS domain.

1. Introduction

In recent years, the Compressive Sensing (CS) paradigm [5,7,11] brings a lot of benefits to image transmission and processing, due to the compressive sampling process that captures signals at a sub-Nyquist rate. Specially, by performing CS sampling on the existing image dataset, the storage cost can be saved. However, the high computational complexity of the CS reconstruction makes it unrealistic for the processing on the resource constrained device such as a smart phone. Fortunately, the growing trend of computation outsourcing to the public cloud [8,12,16,32] provides a new avenue to solve this problem. Hence, with the help of cloud's abundant computing and storage resources, a practical image storage and reconstruction service for resource constrained device under CS framework could be realized.

1.1. Challenges

Although being promising, this image outsourcing plan brings in challenges. On one hand, to be a secure outsourcing protocol, it should fulfill some basic requirements, such as the correctness, privacy protection and efficiency requirement that

* Corresponding author. Fax: +86 23 6510 4570.
E-mail address: xiaodi_cqu@hotmail.com (D. Xiao).

http://dx.doi.org/10.1016/j.ins.2016.09.045
0020-0255/

were suggested in [2,3,19,30]. Namely, the proposed scheme is required to achieve privacy preserving with low complexity, while ensuring that the original solution can be successfully decrypted from the cloud's output.

On the other hand, from the perspective of cloud management, it is a sticky business for the cloud server to distinguish the data if the content is privacy protected. In some scenarios, the cloud server may desire to resolve the rightful ownership of the data or the legality of the user to avoid some malicious uploading and downloading. However, it is hard to realize if the cloud derives no information from the privacy protected data. In other words, for such scenario, there requires an intersection between the data management and the privacy preserving.

1.2. Motivations

To address the above challenges, we propose Cloud-assisted Authenticated Image Storage Service (CAISS), a novel outsourced image reconstruction and identity authentication service in cloud. Firstly, a lightweight transformation is performed on the CS reconstruction problem to achieve sufficient security and low complexity. In more detail, the CS samples of image are directly outsourced to the cloud based on the fact that the CS transformation can be viewed as a symmetric cipher with computational security if the secret sensing matrix is unknown to the adversary [21,23]. Secondly, to achieve the goal of cloud management under image privacy assurance, we intuitively plan to perform an identity authentication on the CS samples. More precisely, if we can resolve the identity of the user by carrying out a detection in the CS domain, the cloud management under data privacy assurance would be reliable. Tactfully, the work of [9] explores that some signal processing such as detection can be solved in the CS domain. Inspired by these facts, we intend to integrate the technique of signal processing in the CS domain into the computation outsourcing for the purpose of image CS reconstruction and identity authentication.

1.3. Contributions

We regard our main contributions as follows:

1. We implement a system to provide image storage and CS reconstruction service in cloud, while maintaining privacy assurance and management of the images. This could be viewed as an extension of computational outsourcing application.
2. The design of low complexity encryption of the proposed outsourcing scheme is useful for practical application, since significant speedup and less resource consuming are gained.
3. We show by theoretical analysis and experimental evaluation that the proposed scheme is efficient and effective. Experiments are also conducted to verify the feasibility of identity authentication in the CS domain.

The rest of the paper is organized as follows: In Section 2, the related work is introduced. Then, the system architecture and preliminaries are described in Section 3. Section 4 gives the detailed mechanism design. Sections 5 and 6 give some related analysis and performance evaluation. Finally, some conclusions are drawn in Section 7.

2. Related work

A line of research related to our work is the secure computation outsourcing to the public cloud for storage and computations. Till now, many protocols have been designed for secure outsourcing. For example, Lei et al. proposed a number of protocols for secure outsourcing scientific computations, such as solving matrix multiplication, matrix inversion computation [16,17]. However, designs of these schemes can hardly apply to other systems because of their high specificity. For generalization, Zhang et al. proposed a general computation outsourcing scheme for inverting homomorphic functions with computation disequilibrium [32]. But it is not suitable for our system because of the property of inverting homomorphic functions. Apart from that, some new proposals [26–28] are closely related to our work. In [26] and [27], Wang et al. proposed a secure outsourcing of linear programming scheme and a privacy assured outsourcing of image reconstruction service in cloud, respectively. Moreover, a cloud assisted outsourcing scheme for healthcare video monitoring was proposed in [28]. Although the basic framework of our system is similar to these works to some extent, we extend the image outsourcing system to a broader scenario with low complexity. Besides, the work proposed by Divekar et al. has also inspired us, where the CS samples of image is suggested to be stored for datasets storage reduction [10]. However, their work does not take security into consideration.

Another research area that are related to our work is the signal processing in the CS domain. It has been shown that many signal processing algorithms performed in the CS domain have close performance as performed in the original domain [9,18,29]. Davenport et al. [9] have taken some initial steps towards a general framework called compressive signal processing, which shows fundamental signal processing problems such as detection, classification, estimation and filtering can be solved in the CS domain. Wang et al. proposed a CS based privacy preserving framework for collaborative data mining and signal processing using secure multiparty computation (MPC) protocol in which the signal processing is performed in the CS domain [29]. However, their protocol is based on the Paillier [22] public key system, which is significantly different from our work.

3. Problem statement

Before presenting CAISS design, we introduce some preliminaries about CS and linear correlation in the random projection domain.

3.1. Preliminaries

3.1.1. Compressive sensing

Compressive Sensing (CS) is a new signal sampling theory telling us that we can exactly recover the original signals through few measurements less than Shannon sampling rate if signal is sparse or compressible. The CS framework includes sampling process in the encoder side and reconstruction process in the decoder side. The sampling process is a non-adaptive linear projection, which preserves the information of an $n \times 1$ signal $\mathbf{x}$ with only $m \ll n$ measurements.

The sampling process is done by multiplying an $m \times n$ measurement matrix $\mathbf{\Phi}$ to $\mathbf{x}$ that derives an $m \times 1$ sample vector $\mathbf{y} = \mathbf{\Phi x}$, where $\mathbf{\Phi}$ satisfies the restricted isometry property (RIP) of a certain order [4]. If $\mathbf{x}$ is not sparse itself, it may be represented as a sparse signal in some orthonormal basis $\mathbf{\Psi}$ via $\mathbf{s} = \mathbf{\Psi}^{-1}\mathbf{x}$, where $\mathbf{\Psi}^{-1}$ is the inverse of $\mathbf{\Psi}$. Then, the sampling process can be directly performed on the sparse signal $\mathbf{s}$ via $\mathbf{y} = \mathbf{\Phi s}$. In this way, the signal $\mathbf{s}$ can be reconstructed from the measurement $\mathbf{y}$ by solving an l_1 minimization problem as (1). After obtaining $\hat{\mathbf{s}}$, the original signal can be recovered via $\hat{\mathbf{x}} = \mathbf{\Psi}\hat{\mathbf{s}}$.

$$\hat{\mathbf{s}} = \arg\min_{\mathbf{s}\in\mathbb{R}^n} \|\mathbf{s}\|_1 \ \ s.t.\ \mathbf{y} = \mathbf{\Phi s}. \tag{1}$$

It is worth noting that the l_1 minimization reconstruction problem (1) is equivalent to a linear programming (LP) problem [6] as

$$\min \mathbf{1}^T \cdot \mathbf{q}\ s.t.\ \mathbf{y} = \mathbf{\Phi s}, -\mathbf{q} \le \mathbf{s} \le \mathbf{q}, \tag{2}$$

where $\mathbf{q}$ is an $n \times 1$ vector with positive real variables. If we denote $\mathbf{s} + \mathbf{q} = 2\mathbf{u}$, $\mathbf{q} - \mathbf{s} = 2\mathbf{v}$, where $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, then it holds $\mathbf{s} = \mathbf{u} - \mathbf{v}$ and $\mathbf{q} = \mathbf{u} + \mathbf{v}$. Thus, the problem (2) gets a form as

$$\min \mathbf{1}^T \cdot \mathbf{u} + \mathbf{1}^T \cdot \mathbf{v}\ s.t.\ \mathbf{y} = \mathbf{\Phi}(\mathbf{u} - \mathbf{v}), \mathbf{u} \ge \mathbf{0}, \mathbf{v} \ge \mathbf{0}.$$

Furthermore, we can rewrite the above LP problem in a more standard form by letting $\mathbf{r} = [\mathbf{u}^T, \mathbf{v}^T]^T \in \mathbb{R}^{2n}$, $\mathbf{F} = [\mathbf{\Phi}, -\mathbf{\Phi}] \in \mathbb{R}^{m\times 2n}$, that is

$$\min \mathbf{1}^T \cdot \mathbf{r}\ s.t.\ \mathbf{y} = \mathbf{F} \cdot \mathbf{r}, \mathbf{r} \ge \mathbf{0}. \tag{3}$$

We denote this problem as $\Gamma = (\mathbf{F}, \mathbf{y}, \mathbf{1}^T)$.

3.1.2. Linear correlation in the random projection domain

The theoretical basis of identity authentication in CAISS is based on the early work of data mining in the random projection domain which can be considered as a generalized signal processing in the CS domain [18]. In [18], Liu et al. gave a lemma about linear correlation in the random projection domain. In particular, we rewrite it (lemma 5.4 in [18]) to be fit for the CS domain.

Lemma 1. *[18] : Let $\mathbf{X} = \{X_i\}$, $\mathbf{W} = \{W_i\}$ be two vectors in $\mathbb{R}^n$. Let $\mathbf{\Phi}_{m\times n}$ be am $\times$ n random matrix. Each entry of $\mathbf{\Phi}_{m\times n}$ is independent and identically chosen from Gaussian distribution with mean zero and variance σ^2. Further, let $\mathbf{Y} = \frac{1}{\sqrt{m}\sigma}\mathbf{\Phi}_{m\times n}\mathbf{X}$, and $\mathbf{Z} = \frac{1}{\sqrt{m}\sigma}\mathbf{\Phi}_{m\times n}\mathbf{W}$. Then: $E[Y^T Z] = X^T W$.*

The above lemma infers that the watermark could be "statistically visible" in the CS domain. The work shows CS transformation not only scrambles the original data to protect the privacy but also preserves the linear correlation and Euclidean distance. Motivated by this property, we present a statistical watermark detection scheme to verify the feasibility of identity authentication in the CS domain (more details see Section 4.3).

3.2. System architecture

We assume a semi-trusted cloud as the adversary in CAISS throughout this paper, i.e., the cloud performs the reconstruction service honestly, but is curious in learning content of the client's data. The clients in CAISS involve two entities: the data owner and the end user, which are assumed to mobile devices with only limited computational resources. Note that the watermark that carries the identity, copyright or authorization information is embedded into the image before outsourcing. The detail of embedding scheme will be given in Section 4.3. Thanks to this watermark system, the identity of the end user can be verified even without knowing the actual content of the image, as long as the watermark can be detected in the CS domain.

Fig. 1 demonstrates a potential identity authentication scenario under the CAISS architecture. The basic parties in the CAISS include: the data owner, the cloud and the end user. Firstly, the data owner acquires the CS sample of watermarked

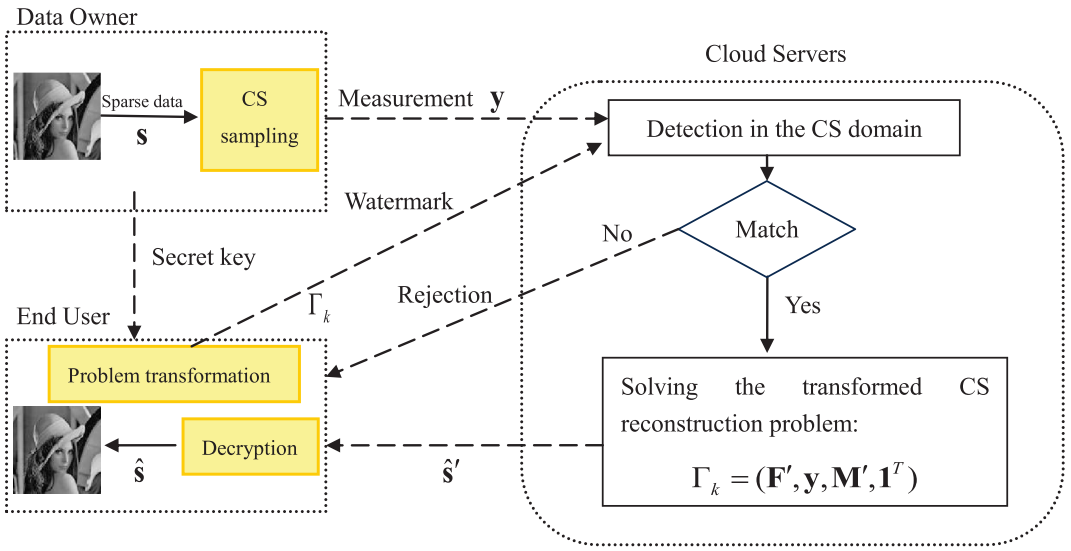

Fig. 1. The CAISS architecture.

image data and uploads it directly to the cloud for storage and further processing. Secondly, the end user sends a CS transformed watermark to the cloud as an access request of data. Meanwhile, the end user transforms the CS reconstruction problem to an encrypted version problem Γ_k and outsources it to the cloud. Thirdly, the cloud detects existence of the claimed watermark in the CS samples after receiving the request from the end user. If the watermark matches, the cloud would solves Γ_k and output an encrypted sparse data $\hat{\mathbf{s}}'$. Otherwise, it refuses to provide the reconstruction service. That is to say, the cloud reconstructs the images just for the authorized end user. In this way, and the aim of management is achieved by preventing the meaningless computation and malicious access of unauthorized users. Finally, the end user recovers the image by decryption and inverse sparse representation.

3.3. Design goals

Our design goals for CAISS consist of the following: (1) CAISS should provide protection on the private content of the recovered images. (2) The cloud should effectively perform the image reconstruction service over the encrypted problem, and the encrypted image can be correctly decrypted by the end user. (3) CAISS should save computation cost to the data owner and the end user, while keeping the complexity of encryption and decryption process as low as possible. (4) The watermark detection in the CS domain should be feasible.

4. The CAISS design

4.1. Problem transformation

The problem transformation should be oriented to the CS reconstruction algorithm, since the mechanisms vary in different kinds of decoding algorithms. In CAISS, we choose to reconstruct the image by solving a LP problem.

The purpose of transformation is to encrypt the original LP problem (3) $\Gamma = (\mathbf{F}, \mathbf{y}, \mathbf{1}^T)$ into a new optimization problem that shares the same structure as the former. In CAISS, we do not encrypt the measurement vector $\mathbf{y}$ any more as it can be considered as cipher text if adversary has no access to the sensing matrix $\mathbf{F}$.

For privacy protection of the output $\mathbf{r}$, the data owner should outsource an encrypted version measurement matrix of $\mathbf{F}$ to the cloud. Therefore, we use a linear matrix transformation to encrypt $\mathbf{F}$. By letting $\mathbf{F}' = \mathbf{F} \cdot \mathbf{M}$, where $\mathbf{M}$ is an $2n \times 2n$ encryption matrix, the decoding problem Γ can be transformed into a new optimization problem as

$$\min \mathbf{1}^T \cdot (\mathbf{M} \cdot \mathbf{e}) \ s.t. \ \mathbf{y} = \mathbf{F} \cdot \mathbf{M} \cdot \mathbf{e}, \mathbf{M} \cdot \mathbf{e} \geq \mathbf{0}. \tag{4}$$

In this way, we can hide output vector $\mathbf{r}$ by the measurement matrix encryption, since $\mathbf{e} = \mathbf{M}^{-1} \cdot \mathbf{r}$, where $\mathbf{M}^{-1}$ is the inverse matrix of $\mathbf{M}$.

It is worth noting that the key idea of CS decoding is to find the sparsest solution of the linear presentation. To ensure $\mathbf{e}$ is the sparsest solution of the transformed optimization problem, the encryption matrix $\mathbf{M}$ should satisfy the condition: the product magnitude of the objective function which is multiplied by $\mathbf{M}$ should be the same as that of the original problem. Furthermore, $\mathbf{M}$ should be invertible.

Note that there is no inequality constraint privacy protection in (4), and this can be achieved by multiplying a positive definite matrix $\mathbf{H}$ to both sides of the inequality constraint. Because for a positive definite matrix $\mathbf{H}$, $\mathbf{H}\cdot\mathbf{M}\cdot\mathbf{e}\geq\mathbf{0}$ is equivalent to $\mathbf{M}\cdot\mathbf{e}\geq\mathbf{0}$. Hence, the fully protected problem becomes

$$\min \mathbf{1}^T\cdot(\mathbf{M}\cdot\mathbf{e})\ s.t.\ \mathbf{y}=\mathbf{F}\cdot\mathbf{M}\cdot\mathbf{e},\mathbf{H}\cdot\mathbf{M}\cdot\mathbf{e}\geq\mathbf{0}. \tag{5}$$

Furthermore, for the objective function of the LP problem, if we enable the term $\mathbf{1}^T\cdot\mathbf{M}$ to be equal to $\mathbf{1}^T$, the problem (5) can be transformed into

$$\min \mathbf{1}^T\cdot\mathbf{e}\ s.t.\ \mathbf{y}=\mathbf{F}'\cdot\mathbf{e},\mathbf{M}'\cdot\mathbf{e}\geq\mathbf{0}, \tag{6}$$

where $\mathbf{F}'=\mathbf{F}\cdot\mathbf{M}$ and $\mathbf{M}'=\mathbf{H}\cdot\mathbf{M}$. We denote the problem (6) as $\Gamma_k=(\mathbf{F}',\mathbf{y},\mathbf{M}',\mathbf{1}^T)$. Note that the output of Γ_k is $\mathbf{e}=[\mathbf{u}'^T,\mathbf{v}'^T]^T\in\mathbb{R}^{2n}$. Thus, we can obtain an encrypted sparse data $\hat{\mathbf{s}}'$ via $\hat{\mathbf{s}}'=\mathbf{u}'-\mathbf{v}'$.

4.2. Encryption matrix implementation

To ensure that the original signal can not be obtained from the reconstructed signal in the cloud, we exploit random permutation matrix $\mathbf{P}$ and diagonal matrix $\mathbf{R}$ with $2n$ random entries to implement the function of encryption matrix $\mathbf{M}$ by letting $\mathbf{M}=\mathbf{R}\cdot\mathbf{P}$.

We use $\pi(i)=i'$ to denote a random permutation function, where $i=1,\ldots,2n$. It can be written as Cauchy's two-line notation $\begin{pmatrix}x_1 & \cdots & x_{2n}\\ x_{\pi(1)} & \cdots & x_{\pi(2n)}\end{pmatrix}$. We can generate a permutation π by choosing uniformly at random from the set $\{1,\ldots,2n\}$. Let $p_{i,j}$ denote the entry of ith row and jth column in matrix $\mathbf{P}$, where i and j are indexed from 1 to $2n$. We can implement a permutation matrix as $p_{i,j}=\begin{cases}1, & i=\pi(j),\\ 0, & others.\end{cases}$. According to this definition, every row or column in $\mathbf{P}$ has exactly one 1 and the rest of the values are 0s. It is an invertible matrix, and its inverse matrix $\mathbf{P}^{-1}$ equals to its transform $\mathbf{P}^T$.

On the other hand, to make sure $\mathbf{1}^T\cdot\mathbf{M}=\mathbf{1}^T\cdot\mathbf{R}\cdot\mathbf{P}$ is equal to $\mathbf{1}^T$, we implement the diagonal matrix $\mathbf{R}$ by two steps. Firstly, we generate the $2n$ diagonal entries a_i at random, where $i=1,\ldots,2n$. Secondly, for each column, we obtain the other $2n$ entries $1-a_i$, $i=1,\ldots,2n$. Thus, the resultant $\mathbf{R}$ holds the form as $\begin{bmatrix}a_1 & \cdots & 1-a_{2n}\\ 1-a_1 & \ddots & 0\\ \vdots & & \vdots\\ 0 & \cdots & a_{2n}\end{bmatrix}$. In practice, we can also choose a full rank random matrix to implement the encryption matrix, which are denoted as $\mathbf{R}'=\begin{bmatrix}b_{1,1} & \cdots & b_{1,2n}\\ \vdots & \ddots & \vdots\\ b_{2n,1} & \cdots & b_{2n,2n}\end{bmatrix}$. In this case, for each column, the $2n-1$ random entries in each column are obtained firstly. Then, we get the $2n$-th entry by letting $b_{2n,j}=1-\sum_{i=1}^{2n-1}b_{ij}$, where $i=1,\ldots,2n$; $j=1,\ldots,2n$. In the remainder of this paper, the diagonal matrix $\mathbf{R}$ with $2n$ random entries is adopted in CAISS unless special statement. Intuitively, it would be more efficient if the diagonal matrix $\mathbf{R}$ with $2n$ random entries is chosen rather than the full rank random $\mathbf{R}'$ with $(2n-1)\times 2n$ random entries. We will verify this assumption by experiment in Section 6.2.

Similarly, we exploit another random permutation matrix $\mathbf{P}_2$ and diagonal matrix $\mathbf{D}$ with all positive random entries to implement the function of encryption matrix $\mathbf{H}$ by letting $\mathbf{H}=\mathbf{D}\cdot\mathbf{P}_2$. Note that $\mathbf{D}$ is positive definite since it has a diagonal form with all positive entries. We can draw the conclusion that $\mathbf{H}$ is also a positive definite matrix, since $\mathbf{P}_2$ just permutes the columns in $\mathbf{D}$. Therefore, the designed encryption matrix $\mathbf{M}$ and $\mathbf{H}$ satisfy the conditions described in Section 4.1.

4.3. Watermark system details

In CAISS, watermark sequence $\{W_i\}\in\{-1,1\}$, $(i=1,\ldots,n)$ is embedded in the frequency domain of image. At first, a set of features $\mathbf{L}=\{L_i\}$ (e.g., some subbands of the discrete wavelet transform (DWT) coefficients or some channels of the discrete cosine transform (DCT) coefficients) are selected. Then, $\{W_i\}$ is embedded into the feature set $\{L_i\}$ with embedding strength α. The embedding process can be expressed as $X_i=L_i+\alpha W_i$, where $\{X_i\}$ is the test feature set embedded with watermark. As the test feature set $\{X_i\}$ may be correlated with the watermark $\{W_i\}$, detection can be accomplished via the hypothesis testing:

$H_0: X_i=L_i$ not contain the claimed watermark

$H_1: X_i=L_i+\alpha W_i$ contain the claimed watermark

Note that we exploit the statistical detection model proposed in [31] to resolve the rightful ownership of images. Namely, whether the test image contains the claimed watermark is determined by comparing the test statistic q with a threshold T_q. In more detail, the test statistic q is defined as

$$q=\frac{\sum_{i=1}^{n}X_iW_i}{V_c\sqrt{n}}=\frac{M_c\sqrt{n}}{V_c}, \tag{7}$$

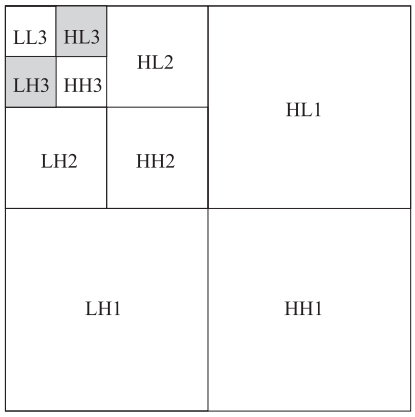

Fig. 2. The selective subbands for watermark embedding: LH3 and HL3 parts of 3-level DWT coefficients.

where n is the size of the test feature set $\{X_i\}$, M_c and V_c are the sample mean and the sample variance of X_iW_i, respectively. i.e.,

$$M_c = \frac{\sum_{i=1}^{n} X_iW_i}{n}; V_c^2 = \frac{\sum_{i=1}^{n} (X_iW_i - M_c)^2}{n-1}. \tag{8}$$

Under hypothesis H_0, for large samples, q is approximately a normal distribution with zero mean and unit variance. i.e., $q \sim N(0, 1)$. Under H_1, for large samples, $q \sim N(\mu, 1)$, where $\mu > 0$. With the help of test statistic q, the watermark is "statistically visible" in the watermarked image without knowing the original image.

In CAISS, the horizontal and vertical DWT subbands are chosen as the candidates for watermark embedding, in order to ensure both the imperceptibility and robustness of the watermark system. More precisely, either the LH3 subband or the HL3 subband of 3-level DWT coefficient is chosen as the feature set $\{L_i\}$, as illustrated in Fig. 2. Reasonably, such information of selection is assumed to be known to the end user and the cloud severs, i.e., all the parties in CAISS know which parts of the data are used for detection.

In a practical CS image system, each image is usually processed block by block to reduce the storage and computation overhead. In this blocking manner, if the size of the processing block is the same as that of the feature set $\{L_i\}$, the watermark detection can be synchronized with the CS sampling process. Namely, the test feature $\{X_i\}$ and watermark $\{W_i\}$ can be transformed in the same CS domain. It is worth mentioning that, for security, different sensing matrices are suggested for different images/image blocks. The corresponding security issue will be discussed in Sections 4.4 and 5.2.

Denote $\mathbf{X} = \{X_i\}$ and $\mathbf{W} = \{W_i\}$, the CS transformations of those sets can be represented as $\mathbf{Y} = [Y_1, Y_2, \ldots, Y_m]^T = \mathbf{\Phi}_{m\times n}\mathbf{X}$, $\mathbf{Z} = [Z_1, Z_2, \ldots, Z_m]^T = \mathbf{\Phi}_{m\times n}\mathbf{W}$, respectively. In this way, the statistic q_{cs} in the CS domain can be defined as:

$$q_{cs} = \frac{\sum_{i=1}^{m} Y_iZ_i}{V_c'\sqrt{n}} = \frac{M_c'\sqrt{n}}{V_c'}, \tag{9}$$

where M_c' and V_c' are the sample mean and the sample variance of Y_iZ_i, respectively. i.e.,

$$M_c' = \frac{\sum_{i=1}^{m} Y_iZ_i}{m}; (V_c')^2 = \frac{\sum_{i=1}^{m} (Y_iZ_i - M_c')^2}{m-1}. \tag{10}$$

Similarly, under H_0, $q_{cs} \sim N(0, 1)$. Under H_1, $q_{cs} \sim N(\mu_{cs}, 1)$, where $\mu_{cs} > 0$.

Note that the larger the value of μ_{cs} is, the smaller the probability of the detection error is. What we want to explore is the relationship between μ and μ_{cs}, i.e., how the CS random projection affect the performance of detection. Besides, we can conclude that the identity information is also privacy protected, since the watermark is encrypted by the CS sampling process $\mathbf{Z} = \mathbf{\Phi W}$.

4.4. The algorithm description

Based on the previous design, five basic algorithms in CAISS are summarized below. Note that all the secret random matrices and sensing matrices are generated by using two keyed pseudo-random functions (PRF) with random seeds and master secret mk_1, mk_2. These two PRFs $\mathcal{F}_1(\cdot)$ and $\mathcal{F}_2(\cdot)$ can be denoted as $\mathcal{F}_1 : \{0,1\}^{|mk_1|} \times \{0,1\}^{l_1} \to \{0,1\}^{l_2}$ and $\mathcal{F}_2 : \{0,1\}^{|mk_2|} \times \{0,1\}^{l_2} \to \{0,1\}^{l_3}$, respectively. Moreover, the master secret mk_1, mk_2 are assumed to be properly shared between the data owner and the end user. Due to this mechanism, a flesh key and sensing matrix can be used for each image.

- *KeyGen*$(1^\kappa, mk_1, mk_2) \to (K, \mathbf{\Phi})$. On inputting a security parameter κ and master secret mk_1, mk_2, this algorithm generates the sensing matrix $\mathbf{\Phi}$ and the secret key tuple $K = (\mathbf{R}, \mathbf{D}, \mathbf{P}, \mathbf{P_2})$, i.e., two permutation matrices $\mathbf{P}$, $\mathbf{P_2}$, and two

diagonal matrices $\mathbf{R}$, $\mathbf{D}$ satisfying the structure described in Section 4.2. Note that the security parameter κ is related to the problem size, i.e., the size of the image/image block. The block size of at least 32×32 is recommended for security consideration.

- *ProbTransform*$(K, \Gamma) \rightarrow \Gamma_k$. This is a problem transformation algorithm that encrypts the original problem Γ into a new optimization problem Γ_k upon getting input of secret key K, according to the transformation mechanism introduced in Section 4.1.
- *DataDetect*$(\mathbf{Y}, \mathbf{Z}) \rightarrow q_{cs}$. This is a watermark detection algorithm running at the cloud side, which returns the statistic q_{cs} upon getting input of $\mathbf{Y} = \mathbf{\Phi}_{m\times n}\mathbf{X}$ and $\mathbf{Z} = \mathbf{\Phi}_{m\times n}\mathbf{W}$ described in Section 4.3.
- *ProbSolve*$(\Gamma_k) \rightarrow \mathbf{e}$. This algorithm is a general LP solver running at the cloud side, which solves the transformed problem Γ_k and outputs $\mathbf{e}$.
- *ImageRec*$(K, \mathbf{\Psi}, \mathbf{e}) \rightarrow \hat{\mathbf{x}}$. This algorithm runs at the end user side. It is used for recovering the original image by decryption $\mathbf{r} = \mathbf{M} \cdot \mathbf{e}$ and inverse sparse representation $\hat{\mathbf{x}} = \mathbf{\Psi}\hat{\mathbf{s}}$, where $\hat{\mathbf{s}} = \mathbf{u} - \mathbf{v}$ and $\mathbf{r} = [\mathbf{u}^T, \mathbf{v}^T]^T \in \mathbb{R}^{2n}$.

4.5. Scheme procedure

For the sake of clarity, we describe the detailed procedure of CAISS via an instantiation below. Note that an index i as a flesh seed is introduced to each image/image block.

(1) For the ith image, the data owner calls *KeyGen* to compute a random seed sd_i by $sd_i \leftarrow \mathcal{F}_1(mk_1, i)$, and to compute a random sequence rs_i by $rs_i \leftarrow \mathcal{F}_2(mk_2, sd_i)$. By using the rs_i, the data owner obtains a random sensing matrix $\mathbf{\Phi}_i$ and a secret key tuple $K_i = (\mathbf{R}_i, \mathbf{D}_i, \mathbf{P}_i, \mathbf{P_2}_i)$.
(2) The data owner performs the CS sampling on the watermarked image to obtain the CS measurement $\mathbf{y}_i$, and then uploads $\mathbf{y}_i$ to the cloud for storage. Note that the index i as a flesh seed for each image is attached with $\mathbf{y}_i$.
(3) To access the ith image, the end user first computes the related K_i and $\mathbf{\Phi}_i$ by calling *KeyGen* with the shared master secret mk_1, mk_2. With K_i and $\mathbf{\Phi}_i$, the end user encrypts the original problem Γ into a new optimization problem Γ_k by calling *ProbTransform*. Besides, the CS transformed watermark $\mathbf{Z}_i = \mathbf{\Phi}_i\mathbf{W}_i$ can also be obtained. Hence, the end user can issue an image access request to the cloud by sending the Γ_k and $\mathbf{Z}_i$.
(4) For the cloud, the access request is processed by calling *DataDetect* to resolve the rightful ownership of the image. If the detection result reveals "match", the cloud would call *ProbSolve* to solve the problem Γ_k and output the answer $\mathbf{e}$ to the authorized end user. Otherwise, the cloud would refuse to serve.
(5) After receiving the answer $\mathbf{e}$, the end user calls *ImageRec* to recover the ith image.

5. Theoretical analysis

5.1. Correctness analysis

We analyze the correctness of our scheme by giving the following proposition.

Proposition 1. *The decryption of reconstruction solution in Γ_k is the optimal solution of the original problem Γ.*

Proof. We prove it by two steps.

Firstly, recalling the original problem Γ and the encrypted problem Γ_k are corresponding to (3) and (6), respectively. Let us consider the equation constrains and in equation constrains in (3) and (6). For equation constrains, it is easy to show that $\mathbf{y} = \mathbf{F}' \cdot \mathbf{e} = \mathbf{F} \cdot \mathbf{M} \cdot \mathbf{e} = \mathbf{F} \cdot \mathbf{r}$. For in equation constrains, due to the fact that $\mathbf{H}$ is a positive definite matrix, it yields $\mathbf{M}' \cdot \mathbf{e} = \mathbf{H} \cdot \mathbf{M} \cdot \mathbf{e} \geq \mathbf{0} \Leftrightarrow \mathbf{M} \cdot \mathbf{e} = \mathbf{r} \geq \mathbf{0}$. To sum up, the constrains of Γ_k remain the same as that of Γ after the transformation.

Secondly, we consider the objective function, suppose $\mathbf{e}_1^*$ is the optimal solution for (6). Then, $\mathbf{r}_1^* = \mathbf{M}\mathbf{e}_1^*$ is an optimal solution for (3).

If not, there is another $\mathbf{r}_2^*$ such that $\mathbf{1}^T \cdot \mathbf{r}_2^* < \mathbf{1}^T \cdot \mathbf{r}_1^*$, having $\mathbf{e}_2^* = \mathbf{M}^{-1} \cdot \mathbf{r}_2^*$. Since $\mathbf{M}$ is a matrix that satisfies the condition $\mathbf{1}^T \cdot \mathbf{M} = \mathbf{1}^T$, it follows $\mathbf{1}^T \cdot \mathbf{e}_2^* = \mathbf{1}^T \cdot \mathbf{M} \cdot \mathbf{e}_2^* = \mathbf{1}^T \cdot \mathbf{r}_2^* < \mathbf{1}^T \cdot \mathbf{r}_1^* = \mathbf{1}^T \cdot \mathbf{M} \cdot \mathbf{e}_1^* = \mathbf{1}^T \cdot \mathbf{e}_1^*$. This means that $\mathbf{e}_1^*$ is not an optimal solution for (6), which is a contradiction.

Combining results of the two steps, we can obtain the proposition. This completes the proof. □

Hence, we can find the sparsest solution of Γ by decrypting $\mathbf{e}$, which holds the same constrain as that of the original problem. Furthermore, we can conclude that the transformed problem Γ_k also meets the CS theory. That is to say, the correctness of our scheme can be concluded from the above proposition.

5.2. Security analysis

We analyze the security of our scheme in output privacy and input privacy.

On one hand, to ensure the privacy of the output $\mathbf{e}$, we propose to encrypt the feasible region of Γ by applying a transformation on the decision variable $\mathbf{r}$ via $\mathbf{e} = \mathbf{M}^{-1} \cdot \mathbf{r}$. From the algorithms described in Section 4.4, it holds that for every $\mathbf{r} \in \mathbb{R}^{2n}$ and every $\mathbf{e} \in \mathbb{R}^{2n}$, there exists a unique $\mathbf{M}$ with $2n$ entries such that $\mathbf{r} = \mathbf{M} \cdot \mathbf{e}$. From the perspective of Shannon's entropy theory [24,25], the output privacy is achieved since an $2n$-entries secret key is sufficient to hide the $2n$ entries in $\mathbf{r}$

as long as $\mathbf{M}$ is chosen uniformly with equal probability by algorithm *KeyGen*. Besides, considering the high correlation on adjacent pixels in natural images, we exploit a permutation matrix $\mathbf{P}$ to reduce the correlation for security enhancement. Obviously, the effect of multiplying $\mathbf{M} = \mathbf{R} \cdot \mathbf{P}$ is that not only the locations of the pixels are exchanged, but also the values of the pixels are modified, i.e., the encryption possesses the effects of both permutation and substitution and results in a confusion-diffusion architecture, which is widely known in image encryption [14]. It follows that our scheme has a desired property of a secure cipher which was identified by Shannon in cryptography [25].

On the other hand, for input privacy, we will show that the privacy of all the inputs $\mathbf{F}'$, $\mathbf{M}'$ and $\mathbf{y}$ can be well protected under some reasonable assumptions. Firstly, considering $\mathbf{F}'$, in the case of encrypting the sensing matrix by $\mathbf{F}' = \mathbf{F} \cdot \mathbf{R}$, some information may be leaked to the cloud. If each $\mathbf{F}'$ is given to the cloud only once, there are $2mn + 2n$ unknown entries and $2mn$ known equations. Suppose each entry has 8 bits, the cloud has $16n$ bits unknown information. From the viewpoint of the cloud, its knowledge about $\mathbf{F}$ is 2^{-16n}. This leaked knowledge is negligible when n becomes large. Furthermore, if a random permutation matrix is added to the encryption via $\mathbf{F}' = \mathbf{F} \cdot \mathbf{R} \cdot \mathbf{P}$, the set of equations become nonlinear. It is known to be NP-hard when solving a system of nonlinear equations [15]. In other words, the cloud cannot get non-negligible information about $\mathbf{F}$ if a flesh key is used for each problem. Similarly, $\mathbf{M}' = \mathbf{H} \cdot \mathbf{M}$ hides $\mathbf{M}$ well under the same condition as that of $\mathbf{F}'$.

Next we consider the security of the CS measurement $\mathbf{y}$. This issue had been discussed in some pioneering works [21,23,34], where the authors state that the CS measurement is computationally secure if the sensing matrix is unknown to the adversary. It is worth mentioning that if the sensing matrix is re-used for multiple plaintexts, the assumption that the adversary has no access to the sensing matrix can be easily cracked under chosen-plaintext attack. However, we can avoid this attack by using a fresh sensing matrix for every source data to be sampled, i.e., our system exactly works in a one-time-pad manner. Note that such property of CAISS is realized by using the keyed PRF with random seeds. In this way, the security of CAISS is reduced to the security of $\mathcal{F}_1(\cdot)$ and $\mathcal{F}_2(\cdot)$, i.e., the proposed outsourcing system is secure if the two used PRFs are secure. Alternatively, the security of CS measurement can be achieved by some new proposals [13,33]. For example, in [13], Fay introduces the counter mode into the CS paradigm to refresh the sensing matrix on every new signal. This work can be grafted to our system by replacing the $\mathcal{F}_1(\cdot)$ with the counter mode operation. Moreover, in [33], Zhang et al. proposed a bi-level protected CS model to achieve security under the sensing matrix re-use scenario. In their model, the "multi-time-sampling" usage of sensing matrix while maintaining security is an appealing option in practice. Inspiringly, the combination of this model with our system would be our future work.

5.3. Efficiency analysis

For the data owner, the most consuming operations in the encryption are matrix-vector multiplications due to the choice of diagonal matrix in transformation. The computation cost of *ProbTransform*is $O(n^2)$. Also note in *ImageRec*, the end user needs to decrypt and recover the original image under sparse representation basis $\mathbf{\Psi}$. The most consuming operations are matrix-vector multiplications and the computation cost of is $O(n^2)$.

On the other hand, to solve a LP problem, a naive algorithm needs $O(n^3)$ time. Obviously, the data owner and the end user will not spend more time to transform and recover the problem than to solve the LP problem on their own. Therefore, the proposed scheme would allow the data owner and the end user to gain considerable computational savings in theory. This claim will be further validated by our experiments in Section 6.2.

As for communication cost, a recent study in [10] has shown that the CS compression could lead to a reduction of about 50% in the file size, compared to storing the original data.

5.4. Watermark detection analysis

Recall that under hypothesis H_1, for large n, $q \sim N(\mu, 1)$, $q_{cs} \sim N(\mu_{cs}, 1)$. Let $E(\cdot)$ denote the expectation operator. Deriving from the work of [31], we conclude that

$$\mu = \frac{E(\mathbf{W} \cdot \alpha\mathbf{W})\sqrt{n}}{V_c} \simeq \frac{\sqrt{E(\alpha^2 \cdot \mathbf{W}^2)}\sqrt{n}}{\sqrt{E(\mathbf{L}^2)}} \simeq \frac{\alpha\|\mathbf{W}\|_2\sqrt{n}}{\|\mathbf{L}\|_2}, \tag{11}$$

where α and $\mathbf{L}$ are the embedding strength and feature set, respectively. To analyze the relationship between μ and μ_{cs}, let us look at μ_{cs} in our watermark system, where

$$\mu_{cs} = \frac{E(\mathbf{\Phi W} \cdot \alpha\mathbf{\Phi W})\sqrt{m}}{V_c'} \simeq \frac{\sqrt{E(\alpha^2 \cdot (\mathbf{\Phi W})^2)}\sqrt{m}}{\sqrt{E((\mathbf{\Phi L})^2)}} \simeq \frac{\alpha\|\mathbf{\Phi W}\|_2\sqrt{m}}{\|\mathbf{\Phi L}\|_2}. \tag{12}$$

The detection performance is depending on $\mathbf{\Phi}$. Particularly, the larger $\alpha\|\mathbf{\Phi W}\|_2$ implies the better performance. As we know that the $\mathbf{\Phi W}$ is the random projection onto the row space of $\mathbf{\Phi}$. Thus, $\|\mathbf{\Phi W}\|_2$ is the norm of the component of $\mathbf{W}$ that lies in the row space of $\mathbf{\Phi}$. This quantity is at most $\|\mathbf{W}\|_2$, which implies the same performance as the detection in the original domain. On the contrary, it could be 0 if $\mathbf{W}$ lies in the null space of $\mathbf{\Phi}$. However, in the case where $\mathbf{\Phi}$ is random, we can see that $\alpha\|\mathbf{\Phi W}\|_2^2$ concentrates around $\alpha(m/n)\|\mathbf{W}\|_2^2$ in the lemma as follows.

Lemma 2. *[1]. For any* $0 < \epsilon < 1$ *and* $\mathbf{x} \in \mathbb{R}^n$, *the random variable* $\|\mathbf{\Phi x}\|_2^2$ *is strongly concentrated around* $(m/n)\|\mathbf{x}\|_2^2$, *that is,*

$$\Pr\left(\left|\|\mathbf{\Phi x}\|_2^2 - (m/n)\|\mathbf{x}\|_2^2\right| \geq \epsilon(m/n)\|\mathbf{x}\|_2^2\right) \leq 2\exp\left(-\frac{m}{2}(\epsilon^2/2 - \epsilon^3/3)\right), \tag{13}$$

where the probability taken over all $m \times n$ *random matrices* $\mathbf{\Phi}$ *[1].*

Specially, random Gaussian matrix $\mathbf{\Phi_G}$ (entries are chosen from Gaussian distribution with i.i.d. zero-mean and fixed variance) and random orthogonal matrix $\mathbf{\Phi_O}$ both satisfy (13), since the orientation of the row space of Gaussian matrix $\mathbf{\Phi_G}$ has random uniform distribution, which makes $\|\mathbf{\Phi_G x}\|_2^2$ has the same distribution as $\|\mathbf{\Phi_O x}\|_2^2$ for a random orthogonal projection.

This lemma shows that for any $\mathbf{x} \in \mathbb{R}^n$ and random matrices $\mathbf{\Phi}$, the random variable $\|\mathbf{\Phi x}\|_2^2$ has expected value $(m/n)\|\mathbf{x}\|_2^2$. It holds that

$$E(\|\mathbf{\Phi x}\|_2^2) = (m/n)\|\mathbf{x}\|_2^2. \tag{14}$$

Substituting (14) into (12), it follows

$$\mu_{cs} \simeq \frac{\alpha\|\mathbf{\Phi W}\|_2\sqrt{m}}{\|\mathbf{\Phi L}\|_2} \simeq \frac{\alpha\sqrt{m/n}\|\mathbf{W}\|_2\sqrt{m}}{\sqrt{m/n}\|\mathbf{L}\|_2} = \frac{\sqrt{m}}{\sqrt{n}}\mu. \tag{15}$$

The result of (15) (i.e., $\mu_{cs} \simeq \sqrt{\frac{m}{n}}\mu$) means that the watermark detection performance in the CS domain is inferior to that in the original domain. Furthermore, we can conclude that the watermark detection distortion in the CS transformation is mainly determined by the CS rate m/n, i.e., a larger $\sqrt{m/n}$ in the CS transformation will cause larger watermark detection distortion. We will verify it by experiment in Section 6.3.

6. Experimental results and performance analysis

In this section, we implement experiment to assess the performance of the proposed image outsourcing service scheme. In the following experiment, both client and cloud server computations are conducted on the same workstation to ignore the communication latency between the client and the cloud, since the computation dominates the running time in our system. For computational complexity consideration, we decompose each image into multiple small size blocks and process the sampling and reconstruction of each block independently. Gaussian random matrix is considered as the measurement matrix and discrete wavelet transform (DWT) basis is used as sparse representation basis. LP solver algorithm by the MOSEK optimization toolbox [20] is employed to realize the image CS reconstruction.

6.1. Effectiveness evaluation

We first assess the effectiveness of CAISS. Specifically, we focus on the correctness and privacy assurance of the design.

6.1.1. Correctness

From perspective of the client, the correctness of design implies the end user could carry out a deterministic decryption to the transformed problem. Intuitively, we can verify it by checking the visual quality of recovered images comparing to the original ones. Fig. 3(a)(e)(i) give some examples of the original images. The recovered images after the end user decryption are shown in Fig. 3(d)(h)(l). Obviously, there is no perceptible difference between the reconstructed images and the original ones.

6.1.2. Privacy preserving

For privacy preserving, CAISS should make sure that the cloud can get no perceptible underlying content from the reconstruct images. Fig. 3(b)(f)(j) shows the recovered images before the end user decryption without knowing the sparse basis $\mathbf{\Psi}$. Furthermore, to verify the security even if the cloud knows $\mathbf{\Psi}$, we implement the inverse sparse representation by $\hat{\mathbf{x}} = \mathbf{\Psi}\hat{\mathbf{s}}'$. The corresponding results are shown in Fig. 3(c)(g)(k). By observing those encrypted images in Fig. 3(c)(g)(k), it is clear that the cloud obtains nothing about the underlying content except some perceptual random noises. That is to say, CAISS provides a good privacy preserving on image content.

6.2. Efficiency evaluation

Theoretical analysis of CAISS has shown that outsourcing indeed benefits the client. In this subsection, we proceed to implement the experiment to evaluate its practical efficiency. Our goal is to find the computation saving gain for the client by outsourcing. In general, the main efficiency indicator is a ratio of the local computational cost without outsourcing over the cost that is needed by the client's computation if outsourcing is chosen. This ratio can be denoted as *asymmetric speedup* $= \frac{t_{original}}{t_{client}}$ according to the definition in Table 1. The value of *asymmetric speedup* should theoretically be a considerable positive number greater than 1, which means there is a considerable computation saving. The experiment on efficiency is implemented via Matlab 7.10.0 (R2010a), on a workstation with an Intel(R) Core i5 CPU running at 3.00GHz and 2.94GB RAM.

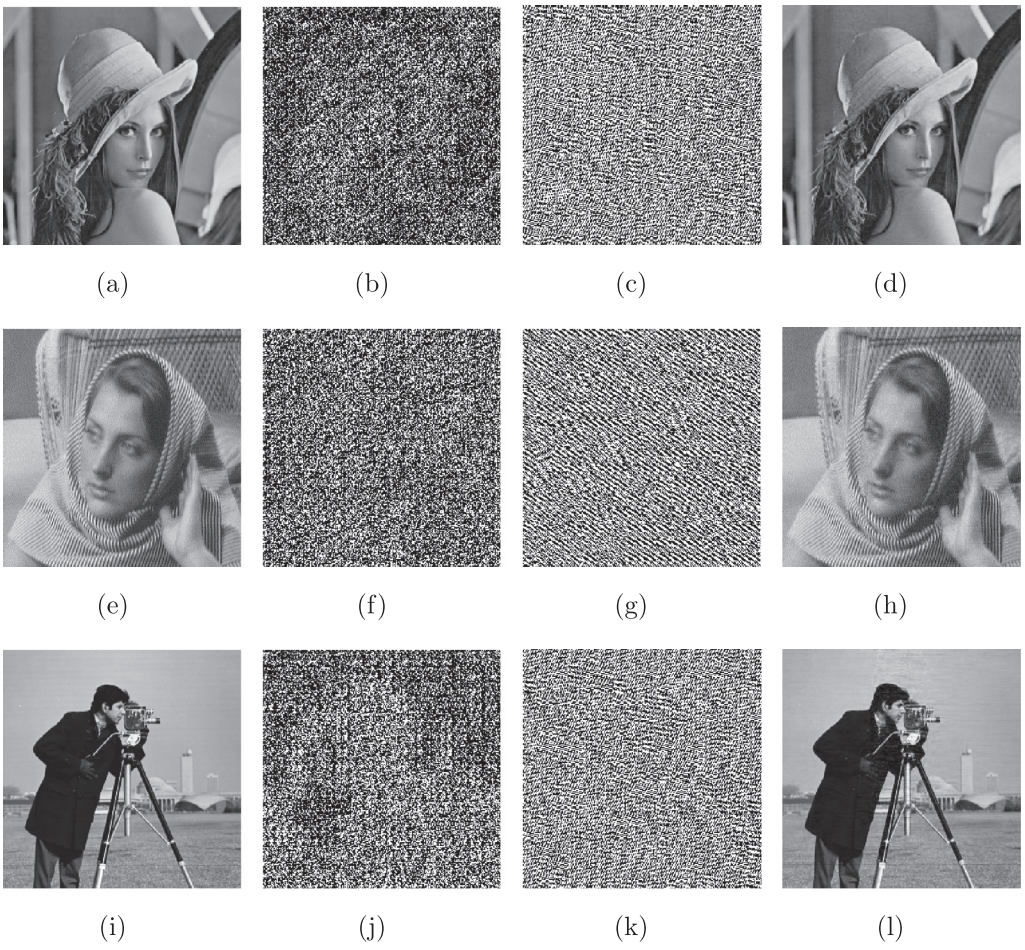

Fig. 3. Results on the effectiveness of CAISS. (a) (e) (i) Original images. (b) (f) (j) Reconstruction sparse data without decryption. (c) (g) (k) Reconstruction images without decryption. (d) (h) (l) Decrypted reconstruction images.

Table 1
Notation of time cost.

Notation	*Means*
$t_{original}$	The time for the client to compute the original LP problem locally
t_{cloud}	The time for the cloud to compute the outsourced LP problem
t_{client}	$t_{data\_owner} + t_{end\_user}$
t_{data_owner}	The time of secret key generation and encryption for the data owner
t_{end_user}	The time for the end user to decrypt and recover the image

Note that each image is decomposed into multiple small size blocks to be sampled and reconstructed independently. The experimental results are shown in Table 2, where each time entry is derived from the mean of 10 trials. From the last column of Table 2, we can see that client speedup monotonically increases along with the processing block size. That means substantial computational cost can be shifted from client to cloud.

Apart from that, we are also concerned about how much of additional burden the scheme imposes on the cloud side. To quantize this efficiency, we consider another indicator, denoted as *cloud efficiency*$=\frac{t_{original}}{t_{cloud}}$, where $t_{original}$ and t_{cloud} refer to the definitions in Table 1. Ideally, the outsourcing should not increase the time to solve the problem after transformation. Namely, the indicator *cloud efficiency* is expected to be close to 1. For the sake of comparison, we implement the experiment by two schemes (i.e., Scheme 1 and Scheme 2). Note that the two schemes transform the LP problem using the same mechanism proposed in this paper, except the different choice of the mask matrices $\mathbf{R}$ and $\mathbf{R}'$. In Scheme 1, we choose a diagonal matrix $\mathbf{R}$ with $2n$ random entries to encrypt the LP problem. In Scheme 2, a full rank random matrix $\mathbf{R}'$ with $(2n-1)\times 2n$ random entries is adopted. It is worth mentioning that full rank random matrix is also used as the mask matrix in the protocols proposed in [27,28]. The comparison results are shown in Table 3, where t_{cloud1} and t_{cloud2} denote the time to solve

Table 2
Performance of asymmetric speedup (time is in "seconds").

Benchmark		Original problem	Encrypted problem			Asymmetric speedup
#	*blocksize*	$t_{original}$	t_{data_owner}	t_{end_user}	t_{client}	$t_{original}/t_{client}$
1	16 × 16	0.36679	0.043202	0.000402	0.043604	8.4119
2	24 × 24	5.9174	0.32741	0.00213	0.32954	17.957
3	32 × 32	67.439	1.63525	0.01445	1.6497	40.879

Table 3
Performance of cloud efficiency (time is in "seconds").

Benchmark		Original problem	Encrypted problem		Cloud efficiency1	Cloud efficiency2
#	*blocksize*	$t_{original}$	t_{cloud1}	t_{cloud2}	$t_{original}/t_{cloud1}$	$t_{original}/t_{cloud2}$
1	16 × 16	0.35314	0.90549	4.2864	0.39	0.082385
2	24 × 24	6.6711	16.546	58.698	0.40318	0.11365
3	32 × 32	62.288	142.53	497.03	0.43703	0.12532

the LP problem encrypted by Scheme 1 and Scheme 2, respectively. We can see from Table 3, the efficiency of Scheme1, i.e., *cloud efficiency1*, keeps satisfactory values which are close to 0.5. In contrast, the values of *cloud efficiency2* are far from 1. This indicates that Scheme 2 with $\mathbf{R}'$ imposes too much additional computation burden on the cloud. The result confirms the assumption that diagonal matrices are more efficient than full rank random matrix for the computation problem transformation. However, from the perspective of security, the full rank random matrix $\mathbf{R}'$ with more random entries could provide stronger security. Therefore, the choice of $\mathbf{R}$ and $\mathbf{R}'$ can be viewed as a tradeoff between security and efficiency. In fact, for different application scenarios (e.g., the privacy preserving of medical image and military image), the definitions of sensitive information vary. At this point of view, our framework provides a flexibility to choose different privacy protection level according to the significance of the underlying content and efficiency requirement.

6.3. Detection feasibility verification

In this subsection, we verify the feasibility of the watermark detection in the CS domain. Experiments are implemented to investigate the performance of the designed statistical correlation hypothesis test based watermark system with CS transformation. In the following experiments, the feature set $\{L_i\}$ is selected from the DWT coefficient as described in Section 4.3. More precisely, the LH3 subband of 3-level DWT coefficient is chosen from a 512 × 512 sized image, i.e., $i = 1, \ldots, 4096$. After a possible watermark embedding process $X_i = L_i + \alpha W_i$, $\alpha = 20$, the test feature set $\{X_i\}$ is transformed to the CS domain (i.e., a sequence $\{Y_i\}$). Then, the watermark $\{W_i\}$ is also transformed to the CS domain using the same measurement matrix to obtain a sequence $\{Z_i\}$. At last, the statistic q and q_{cs} can be calculated by (7) and (9), respectively. Note that each statistic (i.e., q or q_{cs}) is calculated for 1000 trials to investigate its distribution.

At first, we verify the assertion in (15). Recall that the μ_{cs} can be estimated by calculating (15) with the original μ and the CS rate. Thus, we implement a experiment to compare the real watermark detection performance in the CS domain with the estimated μ_{cs}. The test exploits the standard 512 × 512 sized images including "Barbara", "Peppers", "Baboon". Note that each μ_{cs} is obtained from 1000 trials of q_{cs}. Fig. 4 shows the results. It presents that the watermark detection performance in the CS domain decreases with the CS rates (i.e., m/n), while the estimated μ_{cs} is close to the real tested μ_{cs} for all the test images. The remarkable matching observed therein confirms our analysis in Section 5.4. Therefore, the estimated μ_{cs} can be used as a reference to achieve a desired watermark detection performance in the CS domain under a certain CS rate. Fig. 4 also presents that the watermark detection performances of the test images are different with each other. We can see that μ_{cs} of "Baboon" image is higher than others throughout the different CS rates. This is because the bigger high frequency DWT coefficients are selected for watermark embedding due to the highly textured feature of 'Baboon' image.

Then, we assess the detection distortion introduced by the CS transformation. In detail, we investigate this distortion by comparing the distribution of statistic q_{cs} with that of statistic q. With the experiment setting mentioned above, we implement an investigation on the performance of watermark detection under a CS rate 0.5. Fig. 5 shows the results, where "Original" refers to statistic q, "m/n =0.5" refers to statistic q_{cs} with the CS rate 0.5, "H0" and "H1" refer to the two detection hypotheses. By observing the distortion, we can conclude that watermark detection in the CS domain is feasible in general. Nevertheless, according to the analysis result (15), the detection performance in the CS domain decreases with the CS rate. For this consideration, we continue to investigate the acceptable watermark detection performance in the CS domain under low CS rate. Fig. 6 shows the result of comparison between the statistic q_{cs} and q under the CS rate 0.1. The result shows that the watermark can still be detected even with high compressive rate. However, it may lead to detection errors if the CS rate becomes lower. Therefore, it is strongly recommended that CS rate is larger than 0.1, if the proposed system are used for resolving the ownership of image in the CS domain.

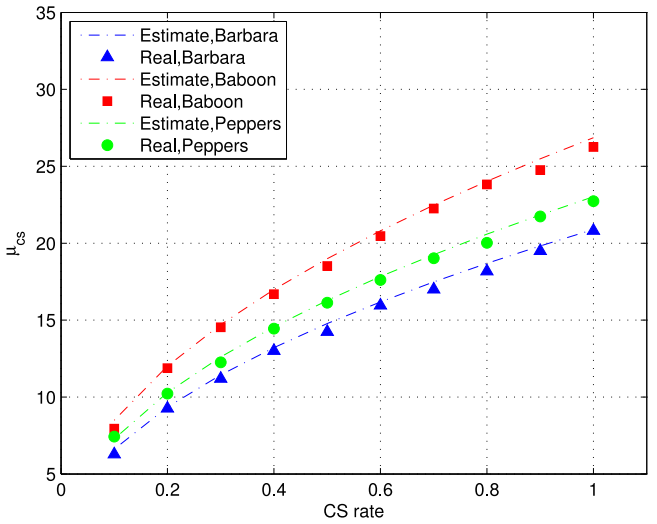

Fig. 4. Watermark detection performance of μ_{cs} with different CS rates.

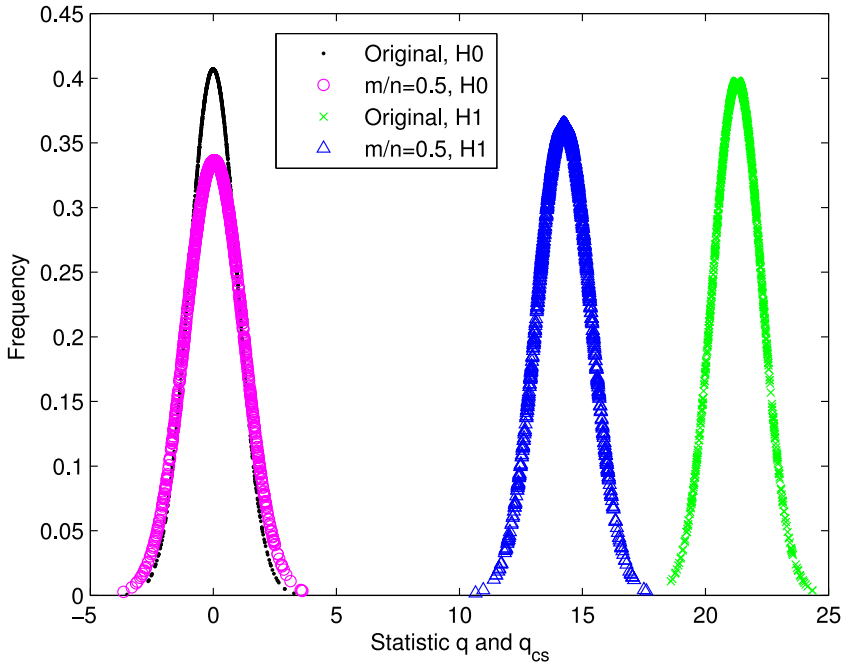

Fig. 5. Distribution of q_{cs} with CS rate 0.5.

To further assess the performance of the detector when the CS rate is sufficiently low, we conduct an experiment to depict the Receiver Operating Characteristic (ROC) curve, i.e., the relationship between True Positive Ratio (TPR) and False Positive Ratio (FPR) that are calculated by (16).

$$TPR = \frac{TP}{TP + FN}, FPR = \frac{FP}{TN + FP}, \tag{16}$$

where *TP*, *TN*, *FP* and *FN* represent the cases of True Positive, True Negative, False Positive and False Negative detection decisions, respectively. In more detail, for different CS rates, we conduct 1000 trials on the test samples with/without random watermark signals embedding. Then, the indices are calculated under an increasing threshold T_q. The results are shown in Fig. 7. As we can see, the ROC curve approaches the upper left corner in the case of $m/n = 0.1$. That means the detector can perform well when the CS rate is above 0.1. While it does not hold if $m/n < 0.1$, as the curves of such CS rates deviate from the upper left corner. This result further confirms the above analysis. The downgrade of detection performance can be

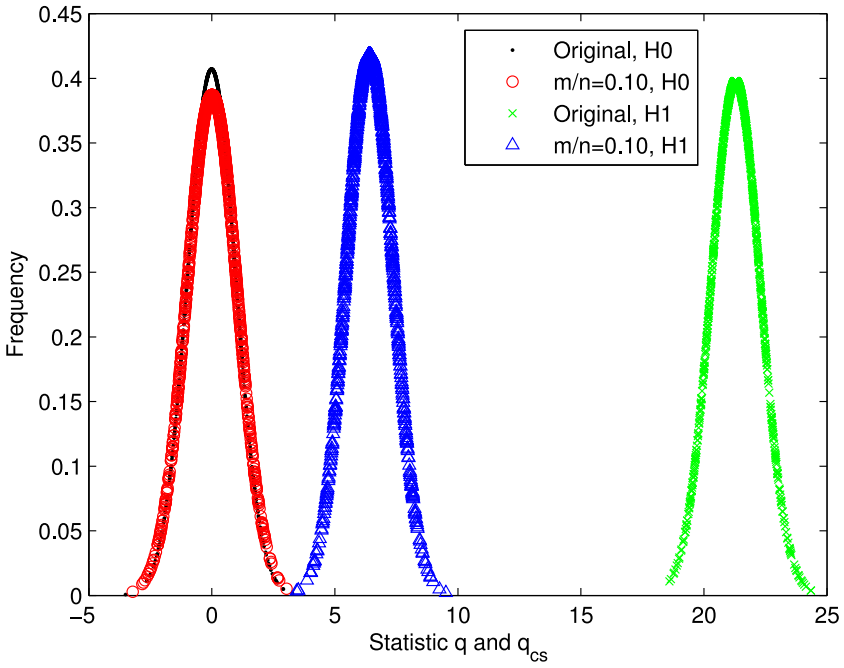

Fig. 6. Distribution of q_{cs} with CS rate 0.10.

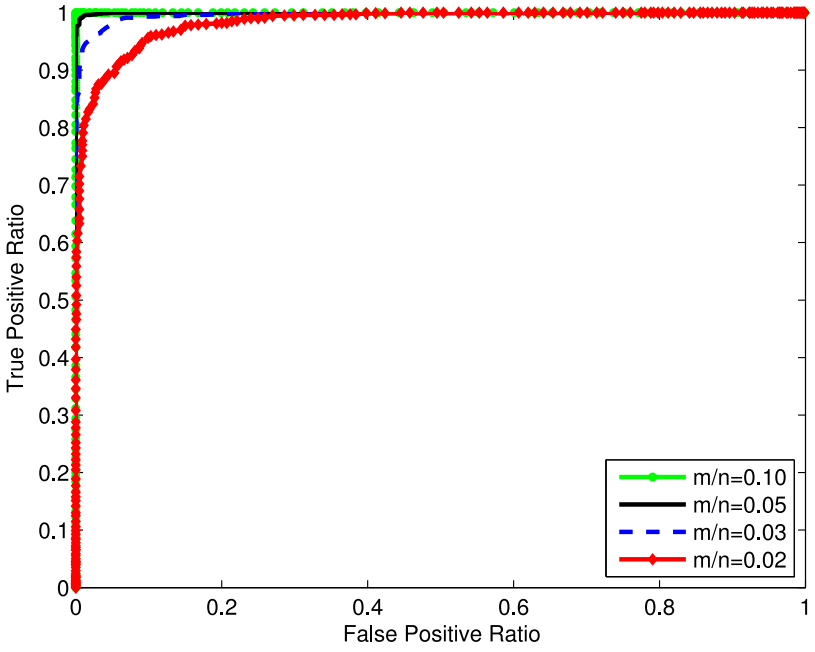

Fig. 7. ROC curve for different CS rates.

interpreted by the intersection of the two Probability Distribution Functions (PDF) of test statistic q_{cs}, i.e., in the cases of with and without watermark embedding, respectively.

To sum up, we can conclude that watermark detection in the CS domain is feasible under certain CS rate.

7. Conclusion

In this paper, a novel image service outsourcing scheme for CS reconstruction computation and identity authentication in cloud is proposed, which integrates the technique of CS domain processing into the secure computation outsourcing.

Theoretical and empirical evaluations demonstrate that the proposed scheme has a satisfactory security and efficiency performance. Experimental results also show that identity authentication in the CS domain is feasible. Moreover, this paper just presents a potential scenario of identity authentication. The framework can be easily extended to other authentication and signal processing applications of outsourcing in cloud, which is our important future work to be pursued.

Acknowledgement

The work was funded by the National Natural Science Foundation of China (Grant Nos. 61272043, 61472464, 61502399, 61572089), the Natural Science Foundation of Chongqing Science and Technology Commission (Grant Nos. cstc2012jjA40017, cstc2013jcyjA40017, cstc2013jjB40009, cstc2015jcyjA40039), the Chongqing Graduate Student Research Innovation Projects (Grant No. CYB14002), the Chongqing Higher Education Reform Projects (Grant No. 153012) and the Fundamental Research Funds for the Central Universities (Grant Nos. 106112013CDJZR180005, 106112014CDJZR185501).

References

[1] D. Achlioptas, Database-friendly random projections, in: Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, ACM, 2001, pp. 274–281.
[2] M. Ali, S.U. Khan, A.V. Vasilakos, Security in cloud computing: opportunities and challenges, Inf. Sci. 305 (2015) 357–383.
[3] J. Archer, A. Boehm, Security guidance for critical areas of focus in cloud computing, Cloud Security Alliance 2 (2009) 1–76.
[4] E.J. Candès, The restricted isometry property and its implications for compressed sensing, Comptes Rendus Mathematique 346 (9) (2008) 589–592.
[5] E.J. Candès, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inf. Theory 52 (2) (2006) 489–509.
[6] E.J. Candes, T. Tao, Decoding by linear programming, IEEE Trans. Inf. Theory 51 (12) (2005) 4203–4215.
[7] E.J. Candès, et al., Compressive sampling, in: Proceedings of the international congress of mathematicians, vol. 3, Madrid, Spain, 2006, pp. 1433–1452.
[8] F. Chen, T. Xiang, Y. Yang, Privacy-preserving and verifiable protocols for scientific computation outsourcing to the cloud, J. Parallel Distrib. Comput. 74 (3) (2014) 2141–2151.
[9] M. Davenport, P.T. Boufounos, M.B. Wakin, R.G. Baraniuk, et al., Signal processing with compressive measurements, IEEE J. Selected Top. Sig. Process. 4 (2) (2010) 445–460.
[10] A. Divekar, O. Ersoy, Compact storage of correlated data for content based retrieval, in: Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on, IEEE, 2009, pp. 109–112.
[11] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (4) (2006) 1289–1306.
[12] F. Emekci, A. Methwally, D. Agrawal, A. El Abbadi, Dividing secrets to secure data outsourcing, Inf. Sci. 263 (2014) 198–210.
[13] R. Fay, Introducing the counter mode of operation to compressed sensing based encryption, Inf. Process. Lett. 116 (4) (2016) 279–283.
[14] J. Fridrich, Symmetric ciphers based on two-dimensional chaotic maps, Int. J. Bifurcation Chaos 8 (06) (1998) 1259–1284.
[15] M.R. Garey, D.S. Johnson, Computers and Intractability, vol. 29, wh freeman, 2002.
[16] X. Lei, X. Liao, T. Huang, F. Heriniaina, Achieving security, robust cheating resistance, and high-efficiency for outsourcing large matrix multiplication computation to a malicious cloud, Inf. Sci. 280 (2014) 205–217.
[17] X. Lei, X. Liao, T. Huang, H. Li, C. Hu, Outsourcing large matrix inversion computation to a public cloud, IEEE Trans. Cloud Comput. 1 (1) (2013). 1–1
[18] K. Liu, H. Kargupta, J. Ryan, Random projection-based multiplicative data perturbation for privacy preserving distributed data mining, IEEE Trans. Knowl. Data Eng. 18 (1) (2006) 92–106.
[19] J. Lloret, M. Garcia, J. Tomas, J.J. Rodrigues, Architecture and protocol for intercloud communication, Inf. Sci. 258 (2014) 434–451.
[20] A. Mosek, The mosek optimization software 54 (2010). Online at http://www.mosek.com.
[21] A. Orsdemir, H.O. Altun, G. Sharma, M.F. Bocko, On the security and robustness of encryption via compressed sensing, in: Military Communications Conference, 2008. MILCOM 2008., IEEE, 2008, pp. 1–7.
[22] P. Paillier, Public-key cryptosystems based on composite degree residuosity classes, in: Advances in Cryptology EUROCRYPT 99, Springer, 1999, pp. 223–238.
[23] Y. Rachlin, D. Baron, The secrecy of compressed sensing measurements, in: 2008 46th Annual Allerton Conference on, Communication, Control, and Computing, IEEE, 2008, pp. 813–817.
[24] C.E. Shannon, Communication theory of secrecy systems*, Bell Syst. Tech. J. 28 (4) (1949) 656–715.
[25] C.E. Shannon, A mathematical theory of communication, ACM SIGMOBILE Mob. Comput. Commun. Rev. 5 (1) (2001) 3–55.
[26] C. Wang, K. Ren, J. Wang, Secure and practical outsourcing of linear programming in cloud computing, in: 2011 Proceedings IEEE, INFOCOM, IEEE, 2011, pp. 820–828.
[27] C. Wang, B. Zhang, K. Ren, J.M. Roveda, Privacy-assured outsourcing of image reconstruction service in cloud, IEEE Trans. Emerg. Top. Comput. 1 (1) (2013) 166–177.
[28] C. Wang, B. Zhang, K. Ren, J.M. Roveda, C.W. Chen, Z. Xu, A privacy-aware cloud-assisted healthcare monitoring system via compressive sensing, in: IEEE INFOCOM 2014-IEEE Conference on Computer Communications, IEEE, 2014, pp. 2130–2138.
[29] Q. Wang, W. Zeng, J. Tian, Compressive sensing based secure multiparty privacy preserving framework for collaborative data-mining and signal processing, in: 2014 IEEE International Conference on Multimedia and Expo (ICME), IEEE, 2014, pp. 1–6.
[30] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, A.V. Vasilakos, Security and privacy for storage and computation in cloud computing, Inf. Sci. 258 (2014) 371–386.
[31] W. Zeng, B. Liu, A statistical watermark detection technique without using original images for resolving rightful ownerships of digital images, IEEE Trans. Image Process. 8 (11) (1999) 1534–1548.
[32] F. Zhang, X. Ma, S. Liu, Efficient computation outsourcing for inverting a class of homomorphic functions, Inf. Sci. 286 (2014) 19–28.
[33] L.Y. Zhang, K.-W. Wong, Y. Zhang, Z. Zhou, Bi-level protected compressive sampling, IEEE Trans. Multimedia 18 (9) (2016) 1720–1732.
[34] Y. Zhang, L.Y. Zhang, J. Zhou, L. Liu, F. Chen, X. He, A review of compressive sensing in information security field, IEEE Access PP (99) (2016). 1–1.