IEICE TRANS. INF. & SYST., VOL.E103–D, NO.9 SEPTEMBER 2020

PAPER

Secure OMP Computation Maintaining Sparse Representations and Its Application to EtC Systems

Takayuki NAKACHI[†a)], ***Member and*** **Hitoshi KIYA**[††], ***Fellow***

SUMMARY In this paper, we propose a secure computation of sparse coding and its application to Encryption-then-Compression (EtC) systems. The proposed scheme introduces secure sparse coding that allows computation of an Orthogonal Matching Pursuit (OMP) algorithm in an encrypted domain. We prove theoretically that the proposed method estimates exactly the same sparse representations that the OMP algorithm for non-encrypted computation does. This means that there is no degradation of the sparse representation performance. Furthermore, the proposed method can control the sparsity without decoding the encrypted signals. Next, we propose an EtC system based on the secure sparse coding. The proposed secure EtC system can protect the private information of the original image contents while performing image compression. It provides the same rate-distortion performance as that of sparse coding without encryption, as demonstrated on both synthetic data and natural images.

key words: *sparse coding, orthogonal matching pursuit (OMP), random unitary transform, secure computation, encryption-then-compression (EtC)*

1. Introduction

Early work on sparse coding was based on the efficient coding hypothesis, which states that the goal of visual coding is to faithfully represent a visual input with minimal neural effort. The idea originated with Barlow [1]. It represents observed signals effectively as a linear combination of a small number of atoms chosen from basis functions trained by a dictionary learning algorithm. The sparse coding model has found numerous processing applications [2] for signals such as images/video [3]–[7], audio [8], biological signals [9], and seismic data [10].

Another trend is the adoption of edge/cloud computing in many fields, including applications that use sparse coding. Edge/cloud computing, however, poses several serious issues for end users, such as unauthorized use, data leaks, and privacy failures due to the unreliability of providers and accidents [11]. In recent years, considerable effort has been made in the fields of fully homomorphic encryption (FHE) [12], [13] and multi-party computation (MPC) [14]. Unfortunately, those methods require high communication, high computation complexity, or a large ciphertext size, so further advances are needed for some applications such as big data analysis and advanced image/video processing. Hence, those methods cannot be applied yet for sparse coding algorithms.

In this paper, we focus on a secure computation of sparse coding. The proposed scheme is based on a random unitary transform, which has much lower communication, lower computational complexity, and a smaller ciphertext size than either FHE or MPC has. Secure computation methods based on random unitary transforms have been reported for biometric template protection [15]–[17]. A random unitary transform also has some desirable properties such as being applicable in l_2-norm minimization problems. We thus propose a secure expression for an Orthogonal Matching Pursuit (OMP) algorithm to minimize the l_0-norm of a sparse representation [18], [19]. OMP [20] is a greedy algorithm that chooses atoms sequentially and calculates a sparse representation. Our proposed method can control the sparsity without decoding the encrypted signals. In the process of secure OMP computation, the sparsity can be controlled gracefully by adding atoms sequentially, whereas a conventional l_2-norm-based template protection method cannot control the sparsity.

Next, we propose an encryption-then-compression (EtC) system using image patches based on secure sparse coding. EtC systems have been proposed to securely transmit and compress images through an untrusted channel provider [21]–[26]. Currently, there is no EtC system using sparse coding. Image compression using sparse coding has been reported to provide better coding performance than that of the JPEG and JPEG2000 standards and state-of-the-art dictionary-learning-based methods [4]–[7]. The proposed secure sparse coding yields exactly the same sparse representations from the original and encrypted signal domains, resulting in excellent rate-distortion performance. Furthermore, the proposed EtC system achieves a graceful rate-distortion tradeoff that is inherent to natural images, while conventional EtC schemes do not have this property. Finally, we demonstrate the performance both on synthetic data and on natural images for application to an EtC system.

The organization of this paper is as follows. Section 2 reviews conventional unitary transform-based template protection methods and Encryption-then-Compression (EtC) systems. Section 3 overviews sparse coding. In Sect. 4, we propose the secure OMP computation, and Sect. 5 introduces its application to EtC systems. Section 6 shows the results of numerical demonstrations. Finally, Sect. 7 gives a conclusion and discusses future work.

Manuscript received November 21, 2019.
Manuscript revised March 10, 2020.
Manuscript publicized June 22, 2020.

[†]The author is with NTT Network Innovation Laboratories, NTT Corporation, Yokosuka-shi, 239–0847 Japan.
[††]The author is with Tokyo Metropolitan University, Hino-shi, 191–0065 Japan.
a) E-mail: takayuki.nakachi@ieee.org

DOI: 10.1587/transinf.2019EDP7309

2. Related Works

In this section, we provide a review of the conventional unitary transform-based template protection methods and Encryption-then-Compression (EtC) systems.

2.1 Unitary Transform-Based Template Protection

Secure computation methods based on random unitary transforms have been reported for biometric template protection [15]–[17]. Those methods have been shown to have a property that the l_2-norm minimization between templates protected by a unitary transform is the same as that between the original ones. Gram-Schmidt orthogonalization is a typical method for generating random unitary transforms. Random phase scrambling and random permutation have also been considered as schemes for generating random orthogonal matrices. Random unitary transform-based template protection has been applied in face recognition experiments to verify its effectiveness. Currently, it has not yet been applied to sparse coding algorithms for minimizing the l_0-norm of sparse representations.

2.2 Encryption-Then-Compression (EtC) Systems

Encryption-then-Compression (EtC) systems have been proposed to securely transmit and compress images through an untrusted channel provider [21]–[26], while the traditional way is to use Compression-then-Encryption (CtE) systems. EtC systems allow us to avoid non-encrypted images with social networking service (SNS) providers, because encrypted images can be directly compressed even when they are multiply recompressed by SNS providers. One type of state-of-the-art EtC system is a block-scrambling-based encryption scheme using the JPEG standard [24]–[26]. Such schemes in EtC systems mainly use geometric transformations (block scrambling, block rotation/inversion) and color transformations (negative-positive transformation, color component shuffling). Currently, there is no EtC system based on sparse coding.

3. Preparation

In this section, we overview sparse coding as a basis for secure computation.

3.1 Sparse Representation

As shown in Fig. 1, by using an overcomplete dictionary matrix $\boldsymbol{D} = \{\boldsymbol{d}_1, \ldots, \boldsymbol{d}_K\} \in \mathbb{R}^{n \times K}$, whose columns contain K prototype atoms $\boldsymbol{d}_i$, a signal vector $\boldsymbol{y} = \{y_1, \ldots, y_n\}^T \in \mathbb{R}^n$ can be represented as a sparse linear combination of the atoms:

$$\boldsymbol{y} = \boldsymbol{D}\boldsymbol{x}. \tag{1}$$

Here, the vector $\boldsymbol{x} = \{x_1, \ldots, x_K\}^T \in \mathbb{R}^K$ contains the sparse representation of the signal vector $\boldsymbol{y}$. If $n < K$ and $\boldsymbol{D}$ is a full-rank matrix, then the representation problem has an infinite number of solutions. The solution with the fewest nonzero coefficients is certainly an attractive representation. This sparsest representation is the solution of

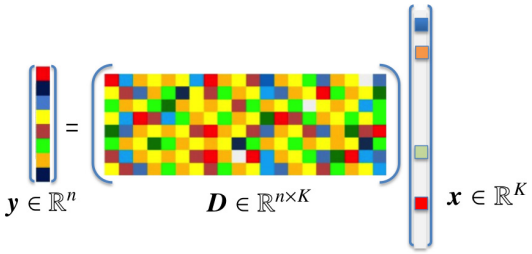


Fig. 1 Sparse coding: a linear combination of a small number of bases.

$$(P_0) \quad \min_{\boldsymbol{x}_0} \|\boldsymbol{x}\|_0 \quad \text{subject to} \quad \boldsymbol{y} = \boldsymbol{D}\boldsymbol{x}, \tag{2}$$

where $\|\cdot\|_0$ is the l_0-norm, counting the vector's nonzero entries. Unfortunately, extraction of the sparsest representation is an NP-hard problem [27].

3.2 Estimation of Sparse Representation

Dictionary atoms are typically selected by a "pursuit algorithm" that finds an approximate solution:

$$(P_{0,\epsilon}) \quad \min_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 \quad \text{subject to} \quad \|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{x}\|_2 \le \epsilon. \tag{3}$$

Well-known pursuit algorithms include Matching Pursuit (MP) [28] and Orthogonal Matching Pursuit (OMP) [20]. These methods are simple, as they involve computation of inner products between the signal and the dictionary atoms. OMP is a greedy, step-wise regression algorithm. At each stage, it selects the dictionary atom having the maximal projection onto the residual signal. After each selection, it applies least-squares search to find a sparse representation with respect to the atoms selected so far. Given the signals $\boldsymbol{y} \in \mathbb{R}^n$ and the dictionary $\boldsymbol{D}$ with K l_2-normalized columns $\{d_k\}_{k=1}^K$, the following is a formal description of the OMP algorithm:

[Orthogonal Matching Pursuit (OMP)]

Initialization: Set $k = 0$, and set

· Initial solution $\boldsymbol{x}^0 = \boldsymbol{0}$
· Initial residual $\boldsymbol{r}^0 = \boldsymbol{y} - \boldsymbol{D}\boldsymbol{x}^0 = \boldsymbol{y}$
· Initial solution support $\boldsymbol{S}^0 = \emptyset$

Main Iteration:
Increment k by 1 and perform the following steps:

· **Sweep:** Compute the errors

$$\epsilon(i) = \min_{x_i} \|x_i \boldsymbol{d}_i - \boldsymbol{r}^{k-1}\|_2^2 = \|\boldsymbol{r}^{k-1}\|_2^2 - \frac{(\boldsymbol{d}_i \cdot \boldsymbol{r}^{k-1})^2}{\|\boldsymbol{d}_i\|_2^2}. \tag{4}$$

Here, we define an atom $\boldsymbol{d}_i$ as

$$\boldsymbol{d}_i = \boldsymbol{D}\boldsymbol{\delta}_i, \tag{5}$$

where $\boldsymbol{\delta}_i = [(0, \cdots, 0, \delta(i), 0, \cdots, 0)]^T$ has all elements equal to 0 except one (i.e., the i-th element is 1). The approximation errors $\epsilon(i)$ in Eq. (4) are then expressed as

$$\epsilon(i) = \|\boldsymbol{r}^{k-1}\|_2^2 - \frac{(\boldsymbol{D}\boldsymbol{\delta}_i \cdot \boldsymbol{r}^{k-1})^2}{\|\boldsymbol{D}\boldsymbol{\delta}_i\|_2^2}. \tag{6}$$

· **Update Support:** Find the minimizer

$$i_0 = \arg\min_{i \notin \boldsymbol{S}^{k-1}} \{\epsilon(i)\}, \boldsymbol{S}^k = \boldsymbol{S}^{k-1} \cup \{i_0\}. \tag{7}$$

· **Update Provisional Solution:** Compute

$$\begin{aligned} \bar{\boldsymbol{x}}^k &= \arg\min_{\boldsymbol{x}_{\boldsymbol{S}^k}} \|\boldsymbol{y} - \boldsymbol{D}_{\boldsymbol{S}^k}\boldsymbol{x}_{\boldsymbol{S}^k}\|_2^2 \\ &= \{(\boldsymbol{D}_{\boldsymbol{S}^k})^T \boldsymbol{D}_{\boldsymbol{S}^k}\}^{-1}\{(\boldsymbol{D}_{\boldsymbol{S}^k})^T \boldsymbol{y}\}, \end{aligned} \tag{8}$$

where $\boldsymbol{D}_{\boldsymbol{S}^k}$ is a submatrix of $\boldsymbol{D}$ consisting of the columns $\boldsymbol{d}_i$ with $i \in \boldsymbol{S}^k$, and $\boldsymbol{x}_{\boldsymbol{S}^k}$ is the set of columns of $\boldsymbol{x}$ corresponding to the support $\boldsymbol{S}^k$.

· **Update Residual:** Compute

$$\boldsymbol{r}^k = \boldsymbol{y} - \boldsymbol{D}_{\boldsymbol{S}^k}\bar{\boldsymbol{x}}^k. \tag{9}$$

· **Stopping Rule:** If $\|\boldsymbol{r}^k\|_2 < \epsilon$, stop. Otherwise, perform another iteration.

Output: The proposed solution $\bar{\boldsymbol{x}}$ is obtained after k iterations.

3.3 Dictionary Learning

An overcomplete dictionary $\boldsymbol{D}$ that leads to sparse representations can either be chosen as a prespecified set of functions or designed by adapting its content to fit a given set of signal examples. Choosing a prespecified transform matrix is the approach used for overcomplete wavelets [29], curvelets [30], short-time Fourier transforms, and so on. On the other hand, dictionary learning algorithms such as MOD [31] and K-SVD [32] seek a dictionary that yields the best representations for a given set of training signals under strict sparsity constraints.

4. Secure OMP Computation

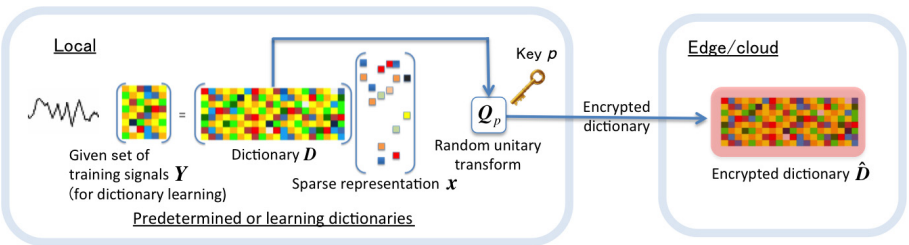
In this section, we propose a secure OMP computation that allows computation in the encrypted domain. We prove theoretically that the proposed computation has exactly the same sparse representation estimation performance as the non-encrypted variant of the OMP algorithm.

4.1 Secure Computation Architecture

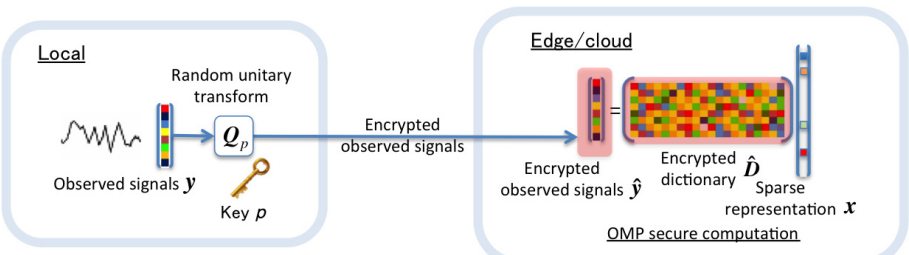
Figure 2 illustrates the architecture of the secure OMP computation. First, Fig. 2(a) shows the preparation by a local site. The dictionary $\boldsymbol{D}$ is predetermined or designed by using dictionary learning algorithms with a given set $\boldsymbol{Y}$ that consists of a number of training signals. Then, a transform function $T(\cdot)$ with a private key p is applied to $\boldsymbol{D}$ to generate an encrypted dictionary $\hat{\boldsymbol{D}}$. The encrypted dictionary $\hat{\boldsymbol{D}}$ is then sent to the intended edge/cloud site and stored in a database. Next, Fig. 2(b) shows the running process of the secure OMP computation for sparse representation selection. The local site applies the same transform function $T(\cdot)$ to observed signals $\boldsymbol{y}$ to generate encrypted observed signals $\hat{\boldsymbol{y}}$. The encrypted signals $\hat{\boldsymbol{y}}$ are then sent to the edge/cloud site, which uses $\hat{\boldsymbol{y}}$ and the stored dictionary $\hat{\boldsymbol{D}}$ sent in advance to perform secure sparse coding.

4.2 Random Unitary Transform

The vector $\boldsymbol{f}_i$ $(i = 1, \cdots, L) \in \mathbb{R}^N$ is encrypted by a ran-



(a) Preparation: generation of encrypted dictionary



(b) Running: secure OMP computation

Fig. 2 Secure sparse coding computation architecture.

dom unitary matrix $\boldsymbol{Q}_p \in \mathbb{C}^{N\times N}$ with a private key p in the following way:

$$\hat{\boldsymbol{f}}_i = T(\boldsymbol{f}_i, p) = \boldsymbol{Q}_p \boldsymbol{f}_i, \tag{10}$$

where $\hat{\boldsymbol{f}}_i$ is the encrypted vector, and L is the number of vectors. Note that the unitary matrix $\boldsymbol{Q}_p \in \mathbb{C}^{N\times N}$ satisfies

$$\boldsymbol{Q}_p^* \boldsymbol{Q}_p = \boldsymbol{I}, \tag{11}$$

where $[\cdot]^*$ and $\boldsymbol{I}$ denote the Hermitian transpose operation and the identity matrix, respectively. Gram-Schmidt orthogonalization is a typical method for generating $\boldsymbol{Q}_p$. In addition to unitarity, $\boldsymbol{Q}_p$ must offer randomness in generating the encrypted signal. Security analyses of such protection schemes have been considered in terms of brute-force attack, diversity, and irreversibility [15]–[17]. Furthermore, the encrypted vector has the following properties.

· Property 1: Conservation of Euclidean distance.

$$\left\|\boldsymbol{f}_i - \boldsymbol{f}_j\right\|_2^2 = \left\|\hat{\boldsymbol{f}}_i - \hat{\boldsymbol{f}}_j\right\|_2^2 \tag{12}$$

· Property 2: Norm isometry.

$$\left\|\boldsymbol{f}_i\right\|_2^2 = \left\|\hat{\boldsymbol{f}}_i\right\|_2^2 \tag{13}$$

· Property 3: Conservation of inner product.

$$\boldsymbol{f}_i^* \boldsymbol{f}_j = \hat{\boldsymbol{f}}_i^* \hat{\boldsymbol{f}}_j \tag{14}$$

These properties hold for orthogonal matrices, which are unitary matrices whose elements have real values.

4.3 Secure OMP Computation Algorithm

The proposed secure sparse coding computation generates the encrypted signals $\hat{\boldsymbol{y}}$ and the dictionary $\hat{\boldsymbol{D}}$ by the following transforms:

$$\hat{\boldsymbol{y}} = T(\boldsymbol{y}, p) = \boldsymbol{Q}_p \boldsymbol{y}, \tag{15}$$

$$\hat{\boldsymbol{D}} = T(\boldsymbol{D}, p) = \boldsymbol{Q}_p \boldsymbol{D}. \tag{16}$$

Instead of using Eq. (3), we consider the following optimization problem containing $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{D}}$:

$$(P_{0,\epsilon}) \quad \min_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 \quad \text{subject to} \quad \left\|\hat{\boldsymbol{y}} - \hat{\boldsymbol{D}}\boldsymbol{x}\right\|_2 \le \epsilon. \tag{17}$$

We focus here on the secure computation of greedy algorithms for minimizing the l_0-norm of a sparse representation, whereas the previous random unitary transform-based template protection solves an l_2-norm minimization problem. We then prove that the sparse representation yielded by the secure OMP computation matches the result of unencrypted computation. The proof is not straightforward, because the OMP algorithm provides an approximate solution. Therefore, whether the random unitary transform-based secure computation provides exactly the same result as the non-encrypted version depends on the algorithm used. From Eqs. (15) and (16) and the random unitary transform properties, the proof is given as follows.

[Secure OMP Computation]

Initialization: Set $k = 0$, and set

· Initial solution $\boldsymbol{x}^0 = \boldsymbol{0}$

· Initial residual $\hat{\boldsymbol{r}}^0 = \hat{\boldsymbol{y}} - \hat{\boldsymbol{D}}\boldsymbol{x}^0 = \hat{\boldsymbol{y}} = \boldsymbol{Q}_p \boldsymbol{y}$

· Iinitial solution support $\boldsymbol{S}^0 = \emptyset$.

Main Iteration:

Increment k by 1 and perform the following steps:

· **Sweep:** Compute the errors.

In Eq. (4), the dictionary $\boldsymbol{D}$ and residual $\boldsymbol{r^{k-1}}$ are replaced with $\hat{\boldsymbol{D}}$ and $\hat{\boldsymbol{r}}^{\boldsymbol{k-1}}$, respectively. From Eqs. (15) and (16), the initial estimation error can be written as

$$\hat{\epsilon}(i) = \min_{\hat{x}_i} \left\|\hat{x}_i \hat{\boldsymbol{D}}\boldsymbol{\delta}_i - \hat{\boldsymbol{r}}^{k-1}\right\|_2^2 = \left\|\hat{\boldsymbol{r}}^{k-1}\right\|_2^2 - \frac{(\hat{\boldsymbol{D}}\boldsymbol{\delta}_i \cdot \hat{\boldsymbol{r}}^{k-1})^2}{\left\|\hat{\boldsymbol{D}}\boldsymbol{\delta}_i\right\|_2^2}. \tag{18}$$

Next, we apply the properties of the unitary transform: $\left\|\hat{\boldsymbol{r}}^{k-1}\right\|_2^2 = \left\|\boldsymbol{r}^{k-1}\right\|_2^2$ (norm isometry), $\hat{\boldsymbol{D}}\boldsymbol{\delta}_i \cdot \hat{\boldsymbol{r}}^{k-1} = \boldsymbol{D}\boldsymbol{\delta}_i \cdot \boldsymbol{r}^{k-1}$ (conservation of inner product), and $\left\|\hat{\boldsymbol{D}}\boldsymbol{\delta}_i\right\|_2^2 = \|\boldsymbol{D}\boldsymbol{\delta}_i\|_2^2$ (norm isometry). From these properties, Eq. (18) can be rewritten as

$$\hat{\epsilon}(i) = \left\|\boldsymbol{r}^{k-1}\right\|_2^2 - \frac{(\boldsymbol{D}\boldsymbol{\delta}_i \cdot \boldsymbol{r}^{k-1})^2}{\|\boldsymbol{D}\boldsymbol{\delta}_i\|_2^2}. \tag{19}$$

Equation (19) is equivalent to Eq. (6), i.e., the relation $\hat{\epsilon}(i)$=$\epsilon(i)$ is satisfied.

· **Update Support:** Find the minimizer.

From $\hat{\epsilon}(i)$=$\epsilon(i)$, the following relation is also satisfied.

$$\begin{aligned} i_0 &= \arg\min_{i\notin \mathbf{S}^{k-1}} \{\hat{\epsilon}(i)\} \\ &= \arg\min_{i\notin \mathbf{S}^{k-1}} \{\epsilon(i)\}, \boldsymbol{S}^k = \boldsymbol{S}^{k-1} \cup \{i_0\}. \end{aligned} \tag{20}$$

· **Update Provisional Solution:**

The square error between the encrypted observed signal and the estimation yielded by using the current support $\boldsymbol{x}_{\boldsymbol{S}^k}$ is represented as $E_2 = \left\|\hat{\boldsymbol{y}} - \hat{\boldsymbol{D}}_{\boldsymbol{S}^k}\boldsymbol{x}_{\boldsymbol{S}^k}\right\|_2^2$. From $\frac{\partial E_2}{\partial \boldsymbol{x}_{\boldsymbol{S}^k}} = 0$, $\hat{\boldsymbol{x}}^k$, which provides the minimum square error, is represented by

$$\hat{\boldsymbol{x}}^k = \{(\hat{\boldsymbol{D}}_{\boldsymbol{S}^k})^T \hat{\boldsymbol{D}}_{\boldsymbol{S}^k}\}^{-1}\{(\hat{\boldsymbol{D}}_{\boldsymbol{S}^k})^T \hat{\boldsymbol{y}}\}. \tag{21}$$

In addition, from the property of conservation of inner product in Eq. (14), $(\hat{\boldsymbol{D}}_{\boldsymbol{S}^k})^T \hat{\boldsymbol{D}}_{\boldsymbol{S}^k}$ and $(\hat{\boldsymbol{D}}_{\boldsymbol{S}^k})^T \hat{\boldsymbol{y}}$ can also be given by $(\boldsymbol{D}_{\boldsymbol{S}^k})^T \boldsymbol{D}_{\boldsymbol{S}^k}$ and $(\boldsymbol{D}_{\boldsymbol{S}^k})^T \boldsymbol{y}$, respectively. Therefore, the provisional solution of Eq. (21) can be rewritten as

$$\hat{\boldsymbol{x}}^k = \{(\boldsymbol{D}_{\boldsymbol{S}^k})^T \boldsymbol{D}_{\boldsymbol{S}^k}\}^{-1}\{(\boldsymbol{D}_{\boldsymbol{S}^k})^T \boldsymbol{y}\}. \tag{22}$$

Equation (22) is equivalent to Eq. (8), i.e., the relation $\hat{\boldsymbol{x}}^k = \bar{\boldsymbol{x}}^k$ is satisfied.

· **Update Residual:**

The residual on the encrypted signals is expressed by $\hat{\boldsymbol{r}}^k = \hat{\boldsymbol{y}} - \hat{\boldsymbol{D}}_{\boldsymbol{S}^k}\hat{\boldsymbol{x}}^k$. From Eqs. (15)–(16) and the equality of the provisional residual, $\hat{\boldsymbol{x}}^k = \bar{\boldsymbol{x}}^k$, the residual can be rewritten as

$$\hat{\boldsymbol{r}}^k = \boldsymbol{Q}_p(\boldsymbol{y} - \boldsymbol{D}_{S^k}\bar{\boldsymbol{x}}^k) = \boldsymbol{Q}_p\boldsymbol{r}^k. \tag{23}$$

· **Stopping Rule:**
If $\|\hat{\boldsymbol{r}}^k\|_2 < \epsilon$, stop. From Eq. (23) and the norm isometry property, this can be expressed as

$$\left\|\hat{\boldsymbol{r}}^k\right\|_2 = \left\|\boldsymbol{r}^k\right\|_2 < \epsilon. \tag{24}$$

The stopping rule is thus equivalent to that of the unencrypted version. Unless it is satisfied, perform another iteration. An alternative stopping rule is that, if

$$k = T_k, \tag{25}$$

then stop, where T_k is the number of specified atoms. Iteration is repeated until the number of selected atoms reaches T_k.

Output: The proposed solution $\hat{\boldsymbol{x}} = \bar{\boldsymbol{x}}$ is obtained after k iterations.

The above analysis shows that the secure OMP computation estimates exactly the same sparse representation as that obtained by the non-encrypted version of the OMP computation. Furthermore, this algorithm can control the sparsity without decoding the encrypted signals. The sparsity is defined as (K - the number of sparse representations $\boldsymbol{x}$)/K, where K is the number of prototype atoms. In the process of the secure OMP computation, the sparsity can be controlled gracefully by adding the atom of the support i_0 sequentially. The sparsity is also indirectly controlled by the stopping rule ("if $\|\hat{\boldsymbol{r}}^k\|_2 < \epsilon$, stop"), and it can be directly controlled by the alternative stopping rule ("if $k = T_k$, stop").

5. Application to EtC System

The sparse coding model has found numerous applications, especially in the area of image processing. In this section, we show an application example of the secure OMP computation to an EtC system.

5.1 EtC System Using OMP Computation

Regarding the effectiveness of sparse coding for image compression, for example, experimental results [7] show that sparse coding outperforms JPEG and JPEG2000 by up to +6 dB and +2 dB, respectively. That work used OMP for updating an atom. Figure 3 illustrates an EtC system using the proposed secure OMP computation for image archives and sharing in an SNS. The figure shows (a) the overall system and (b) its encryption process. The basic structure is the same as that of block-scrambling-based EtC systems using the JPEG standard [24]–[26]. The system divides an image into patches (small images) and then performs encryption and image compression. The main differences from block-scrambling-based EtC systems using the JPEG standard are the following:

1. The proposed system applies a random unitary transform to encrypt each image patch, while a conventional EtC system uses negative-positive transformation and color component shuffling.
2. The proposed system uses overcomplete dictionary-based compression, while a conventional EtC system uses compression based on a discrete cosine transform (DCT).

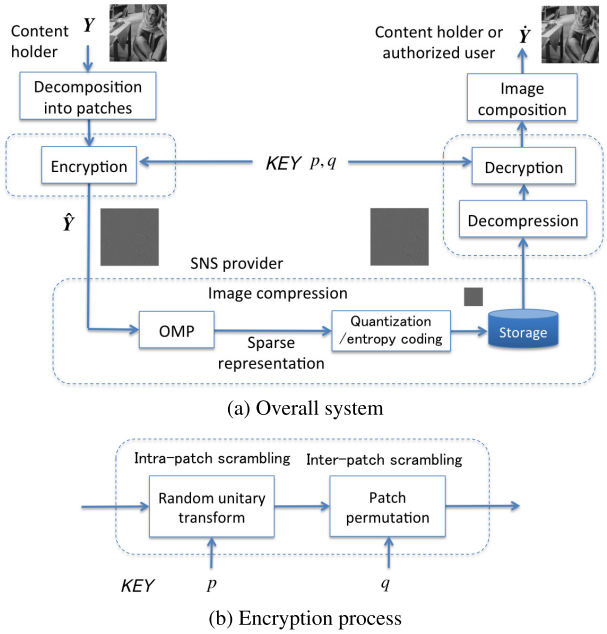


Fig. 3 Encryption-then-compression (EtC) system using secure OMP for image archiving and sharing in an SNS.

By using the random unitary transform, our proposed system can provide the same rate-distortion performance as that of sparse coding without encryption. That is, effective rate-distortion performance can be obtained by applying the secure OMP computation without decoding the encrypted signals.

5.2 Encryption Process

The procedure for generating an encrypted image $\hat{\boldsymbol{Y}}$ is as follows. $\boldsymbol{y}_i \in \mathbb{R}^n$

1. **Preprocessing**: Decomposition into image patches
   We consider image patches of size $\sqrt{n} \times \sqrt{n}$ pixels that are ordered lexicographically as column vectors $\boldsymbol{y}_i \in \mathbb{R}^n (i = 1, \cdots, N)$, where N is the total number of patches. The patches are extracted from an image $\boldsymbol{Y}$ as shown in Fig. 4. We assume that every image patch $\boldsymbol{y}_i$ can be represented sparsely over the overcomplete dictionary $\boldsymbol{D} \in \mathbb{R}^{n \times K}$:

   $$\boldsymbol{y}_i = \boldsymbol{D}\boldsymbol{x}_i, \tag{26}$$

   where $\boldsymbol{x}_i \in \mathbb{R}^K$ $(i = 1, 2, \cdots, N)$ are sparse representations, and N is the total number of image patches. In advance, the dictionary $\boldsymbol{D}$ is designed for the images by applying training algorithms such as MOD [31] and K-SVD [32] at the local site.

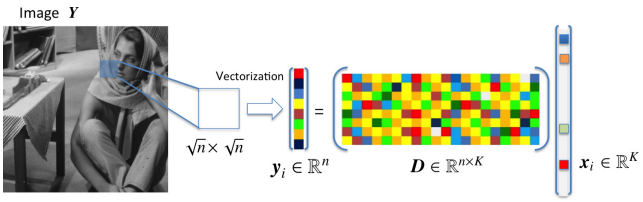


Fig. 4 Sparse coding for image patches.

2. **Intra-Patch Scrambling**: Random unitary transform The secure OMP computation proposed in the previous section is applied to each image patch $\boldsymbol{y}_i$. It generates an encrypted image patch $\hat{\boldsymbol{y}}_i$ and a dictionary $\hat{\boldsymbol{D}}$ by the following transforms:

$$\hat{\boldsymbol{y}}_i = T(\boldsymbol{y}_i, p) = \boldsymbol{Q}_p \boldsymbol{y}_i, \tag{27}$$

$$\hat{\boldsymbol{D}} = T(\boldsymbol{D}, p) = \boldsymbol{Q}_p \boldsymbol{D}, \tag{28}$$

where p and $\boldsymbol{Q}_p$ are a private key and a random unitary transform, respectively, for image patch $\boldsymbol{y}_i$. Sparse representations $\hat{\boldsymbol{x}}_i$ are then estimated for each image patch $\boldsymbol{y}_i$.
3. **Inter-Patch Scrambling**: Patch permutation The encrypted image patches $\hat{\boldsymbol{y}}_i$ $(i = 1, 2, \cdots, N)$ are randomly permuted using a random integer generated by a private key q. Finally, the permuted patches are combined to form an encrypted image $\hat{\boldsymbol{Y}}$, which is fed to the OMP computation.

The proposed method provides enhanced security by combining the intra-patch scrambling using the random unitary matrix with the inter-patch scrambling using the permutation. Security analyses of using the random unitary matrix have been considered in terms of brute-force attack, diversity, and irreversibility [15]–[17]. Regarding the permutation, we can evaluate its security in terms of its key space, assuming that an attacker performs a brute-force attack. The key space of the random permutation is $N!$. For example, in the setting of the numerical demonstrations described in Sect. 6, $N = 4096$ (as calculated for a 512×512 pixel image with 8×8 pixel image patches). The key space $N!$ is thus larger than that of the 256-bit key, i.e., $N! > 2^{256}$.

5.3 Image Compression with Rate-Distortion Control

By feeding the encrypted dictionary $\hat{\boldsymbol{D}}$ and the encrypted image $\hat{\boldsymbol{Y}}$ to the secure OMP computation, we obtain the sparse representation $\hat{\boldsymbol{x}}_i$ for each image patch. The decompressed/decrypted image patch is then obtained as $\hat{\boldsymbol{y}}_i = \boldsymbol{Q}_p^* \hat{\boldsymbol{D}} \hat{\boldsymbol{x}}_i$. The rate-distortion tradeoff between the compression ratio and decompressed/decrypted image quality of each image patch can be controlled by altering the threshold ϵ_i or T_k without decoding the encrypted image. The threshold ϵ_i determines the stopping condition of the secure OMP algorithm, i.e., $\left\|\hat{\boldsymbol{r}}_i^k\right\|_2 < \epsilon_i$. Rate-distortion control can be done gracefully by adding atoms sequentially. Applications that use graceful rate-distortion control can benefit from the capability to adapt the bitstream according to network conditions. This allows more graceful degradation as compared with nonscalable coding, in which reductions in bitrate typically cause more severe drops in image quality, often rapidly reaching unacceptable quality for viewing.

Finally, the prominent features of the proposed OMP-based EtC system are summarized as follows.

1. The rate-distortion performance exceeds that of EtC systems based on a predefined transform (DCT, wavelet, etc.), because the dictionary is trained to fit the images.
2. The rate-distortion tradeoff can be controlled gracefully by adding atoms sequentially without decoding the encrypted image.

6. Numerical Demonstrations

To evaluate the effectiveness of the proposed secure OMP computation, we demonstrated its performance on both synthetic data and natural images, assuming EtC system application in the latter case.

6.1 Synthetic Data

We created a random matrix $\boldsymbol{D}$ of size 30×50. Each column was normalized to a unit l_2-norm. We generated sparse vectors $\boldsymbol{x}$ with independent and identically distributed (iid) random support candidates in the range [1,10], and nonzero entries drawn as random uniform variables in the range [-2,-1] $\cup$ [1,2]. Once $\boldsymbol{x}$ was generated, we computed $\boldsymbol{y} = \boldsymbol{D}\boldsymbol{x}$. We performed 1000 trials per cardinality and report the average results here in terms of two measures: the l_2-error and the support recovery. The l_2-error is computed as the ratio $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2 / \|\boldsymbol{x}\|^2$. The support recovery means the l_2 proximity between the two solutions. Denoting the two supports as $\hat{S}$ and S, we define the distance between them by

$$dist(\hat{S}, S) = \frac{max\{|\hat{S}|, |S|\} - |\hat{S} \cap S|\}}{max\{|\hat{S}|, |S|\}}. \tag{29}$$

We applied the three algorithms below to find $\boldsymbol{x}$:

- OMP
- Secure OMP
- Nonunitary OMP

Here, "OMP" is simply the non-encrypted version of OMP, while "secure OMP" is the proposed method. "Nonunitary OMP" is a method in which the encrypted signals $\hat{\boldsymbol{y}}$ and the encrypted dictionary $\hat{\boldsymbol{D}}$ are transformed by using a random nonunitary transform, instead of the random unitary transform $\boldsymbol{Q}_p$. All these algorithms look for the solution until triggering the stopping rule, $\|\hat{\boldsymbol{r}}^k\|_2 < \epsilon$, where we set $\epsilon = \sqrt{1e-4}$.

Figure 5 plots the average of the l_2-norm residual $\|\boldsymbol{r}^k\|_2$ as a function of the iteration number. By the time the stopping rule is satisfied, adequate convergence is achieved. The

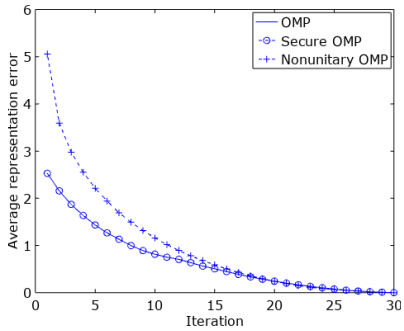


Fig. 5 Average of l_2-norm residual $\|\boldsymbol{r}^k\|_2$.

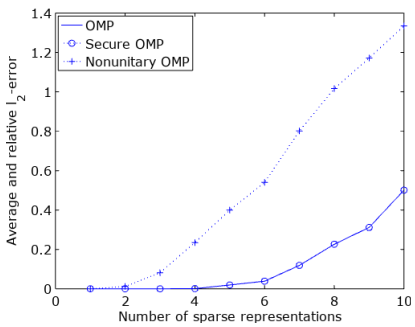


Fig. 6 Average of $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2/\|\boldsymbol{x}\|^2$.

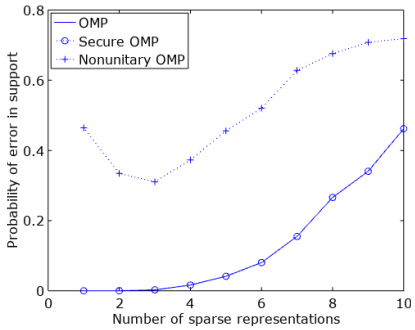
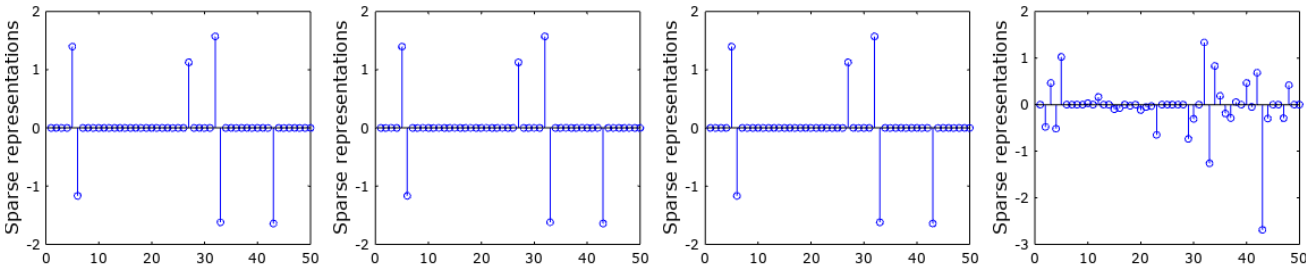


Fig. 7 Recovery of $(\hat{\boldsymbol{S}}, \boldsymbol{S})$.



(a) True value (b) OMP (c) Secure OMP (d) Nonunitary OMP

Fig. 8 Samples of sparse representations $\boldsymbol{x}$ (for 6 representations).

averages of the l_2-error $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2/\|\boldsymbol{x}\|^2$ and the support recovery dist$(\hat{\boldsymbol{S}}, \boldsymbol{S})$ are shown in Figs. 6 and 7, respectively. These figures show that secure OMP gives exactly the same performance as OMP does. On the other hand, nonunitary OMP performs poorly. Lastly, Fig. 8 shows samples of sparse representations $\boldsymbol{x}$ when the number of representations is 6. These results confirm that the proposed method estimates exactly the same sparse representations as OMP does. This shows the importance of the transform's unitarity property.

Next, we evaluated secure OMP with regard to security. We assumed that the number of users was 100 and a private key p_i $(i = 1, 2, \cdots, 100)$ was created for each user at the encoding step. We compared the following two cases:

- Secure OMP by authorized users $(p_i = p_j)$,
- Secure OMP by unauthorized users $(p_i \neq p_j)$,

where p_j $(j = 1, 2, \cdots, 100)$ is a private key at the decoding step. Table 1 lists the average and minimum of the l_2-error $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|^2/\|\boldsymbol{y}\|^2$ for authorized/unauthorized users. The results show that signals encrypted by secure OMP cannot be decrypted by unauthorized users.

Table 1 l_2-norm error $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|^2/\|\boldsymbol{y}\|^2$ for authorized $(p_i = p_j)$ and unauthorized $(p_i \neq p_j)$ users.

(a) Average

Secure OMP $(p_i = p_j)$	Secure OMP $(p_i \neq p_j)$
0.0000	1.9251

(b) Minimum

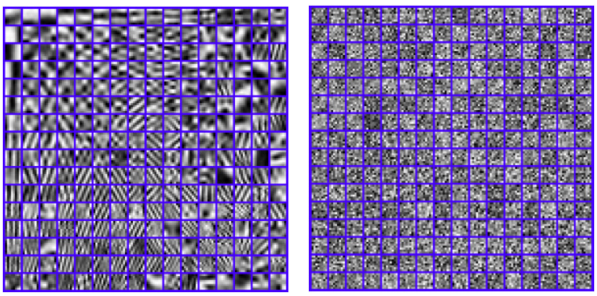
Secure OMP $(p_i = p_j)$	Secure OMP $(p_i \neq p_j)$
0.0000	1.9195

(a) Barbara (b) Mandrill

Fig. 9 Original images.

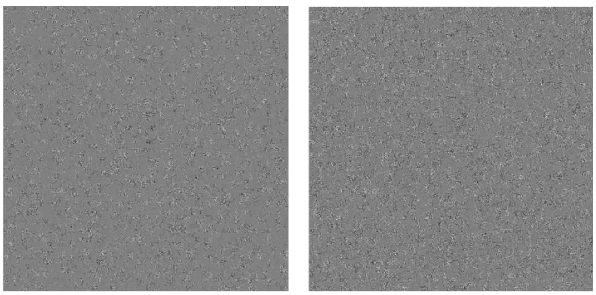
6.2 Natural Images

Next, we conducted experiments on natural images to show the practicality of the secure OMP computation for EtC systems. Specifically, we processed the Barbara and Mandrill images shown in Fig. 9. Both images are 512×512 pixels, with 8 bits/pixel in grayscale. In the preparation process, we trained a dictionary $\boldsymbol{D}$ by K-SVD to sparsely represent patches of 8×8 pixels. Then, the images $\boldsymbol{Y}$ and the trained dictionary $\boldsymbol{D}$ were transformed by a 64×64 random unitary transform $\boldsymbol{Q}_p$ to produce encrypted images $\hat{\boldsymbol{Y}}$ and an encrypted dictionary $\hat{\boldsymbol{D}}$. The random unitary transform $\boldsymbol{Q}_p$ was designed by Gram-Schmidt orthogonalization. Figure 10 shows the dictionary and the corresponding encrypted dictionary. Figure 11 shows the encrypted images $\hat{\boldsymbol{Y}}$, confirming that we cannot see any visible information from the encrypted dictionary $\hat{\boldsymbol{D}}$ or the encrypted images $\hat{\boldsymbol{Y}}$.



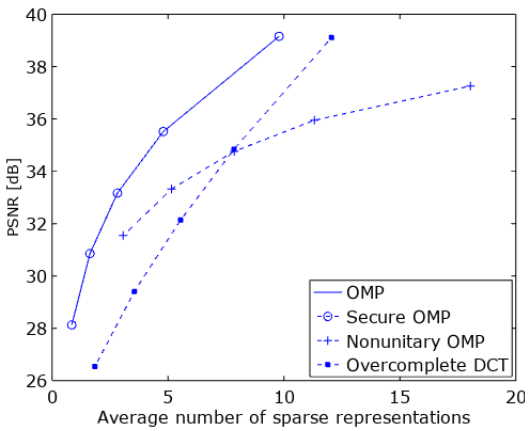
(a) Dictionary $\boldsymbol{D}$ (b) Encrypted dictionary $\hat{\boldsymbol{D}}$

Fig. 10 Trained and corresponding encrypted dictionaries.

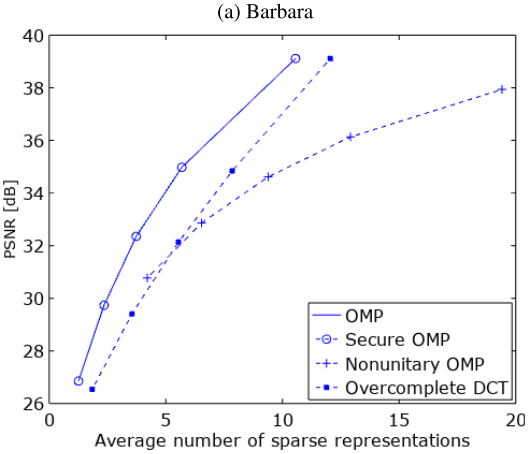


(a) Barbara (b) Mandrill

Fig. 11 Encrypted images $\hat{\boldsymbol{Y}}$.



(a) Barbara



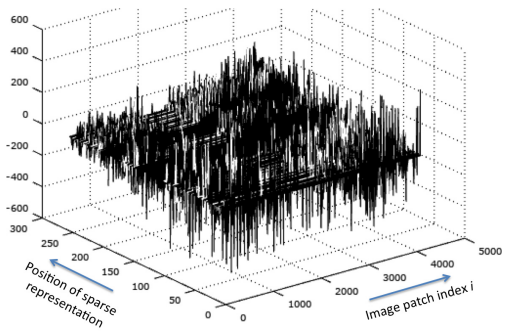
(b) Mandrill

Fig. 12 Rate-distortion performance of secure OMP: average number of sparse representations, $\bar{S}$, vs. decompressed/decoded image quality.

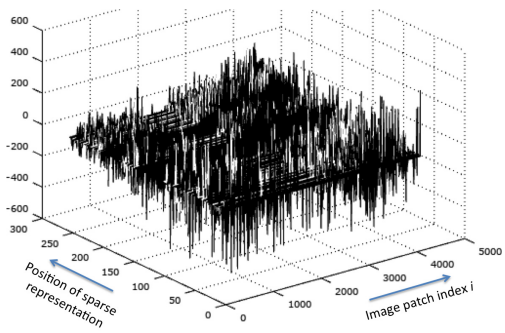
By feeding the encrypted dictionary $\hat{\boldsymbol{D}}$ and the encrypted images $\hat{\boldsymbol{Y}}$ to the secure OMP computation, we obtained a sparse representation $\hat{\boldsymbol{x}}_i$ for each image patch. For both the Barbara and Mandrill images, Fig. 12 shows the rate-distortion performance in terms of the average number of sparse representations, $\bar{S}$, vs. the decompressed/decrypted image quality measured by PSNR [dB], when compared to the non-encrypted version of OMP, nonunitary OMP, and an non-encrypted version of the overcomplete DCT. The average number of sparse representations, $\bar{S}$, is defined by $\bar{S}=\sum_{i=1}^{N} s_i/N$, where s_i is the number of nonzero sparse representations of $\hat{\boldsymbol{x}}_i$, i.e., $s_i = \|\hat{\boldsymbol{x}}_i\|_0$.

The rate-distortion control was done gracefully by adding atoms sequentially in the order of $\epsilon_i = \{15.0, 10.0, 7.0, 5.0, 3.0\}$. From Fig. 12, we can confirm that secure OMP provides the same results as OMP does. The equivalence between the original and encrypted image domains ensures that an original image and its encrypted version yield exactly the same sparse representations, resulting in the same rate-distortion tradeoff, depending on how many sparse representations are preserved. On the other hand, nonunitary OMP provides poor performance. Furthermore, we can see that secure OMP can represent images with a smaller number of sparse representations than the overcomplete DCT can.

Figure 13 shows the sparse representations for the Barbara image (threshold $\epsilon_i = 10.00$). For comparison, the figure also shows the sparse representations estimated by OMP



(a) OMP

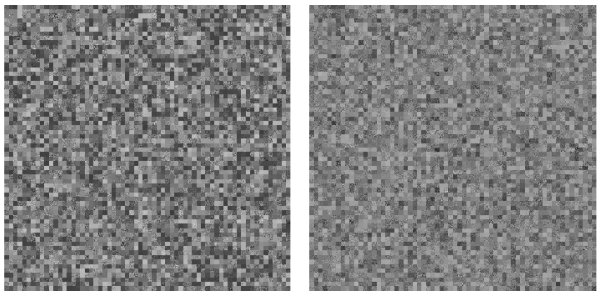


(b) Secure OMP

Fig. 13 Sparse representations $\hat{\boldsymbol{x}}_i$ ($i = 1, 2, \cdots, N = 4096$) for the Barbara image (ϵ_i=10.00).

(a) OMP (PSNR=30.85 dB) (b) Secure OMP (PSNR=30.85 dB)

Fig. 14 Decompressed/decrypted images obtained by an authorized user for the Barbara image (ϵ_i=10.00).



(a) Barbara (b) Mandrill

Fig. 15 Decompressed/decrypted images obtained by an unauthorized user for the Barbara and Mandrill images (ϵ_i=10.00).

Table 2 Decompressed/decrypted image quality obtained by authorized and unauthorized users for the Barbara image.

(a) Authorized user ($\boldsymbol{Q}_{p_i} = \boldsymbol{Q}_{p_j}$)

ϵ_i	15.0	10.0	7.0	5.0	3.0
$\bar{S}$	0.85	1.63	2.82	4.80	9.79
PSNR [dB]	28.13	30.85	33.12	35.56	39.12

(b) Unauthorized user ($\boldsymbol{Q}_{p_i} \neq \boldsymbol{Q}_{p_j}$)

ϵ_i	15.0	10.0	7.0	5.0	3.0
$\bar{S}$	0.85	1.63	2.82	4.80	9.79
PSNR [dB]	10.46	10.39	10.34	10.54	10.40

Table 3 Decompressed/decrypted image quality obtained by authorized and unauthorized users for the Mandrill image.

(a) Authorized user ($\boldsymbol{Q}_{p_i} = \boldsymbol{Q}_{p_j}$)

ϵ_i	15.0	10.0	7.0	5.0	3.0
$\bar{S}$	1.26	2.35	3.73	5.68	10.56
PSNR [dB]	26.85	29.73	32.36	34.98	39.12

(b) Unauthorized user ($\boldsymbol{Q}_{p_i} \neq \boldsymbol{Q}_{p_j}$)

ϵ_i	15.0	10.0	7.0	5.0	3.0
$\bar{S}$	1.26	2.35	3.73	5.68	10.56
PSNR [dB]	13.27	13.22	13.15	13.19	13.15

(a). We can see that the proposed secure OMP (b) estimates exactly the same sparse representations $\hat{\boldsymbol{x}}_i$ as OMP does. Moreover, authorized users can decompress and decrypt the image to obtain the decompressed/decrypted image $\dot{\boldsymbol{Y}}$ shown in Fig. 14.

Next, we evaluated the security of secure OMP from a viewpoint of objective image quality (PSNR) and the visibility of decompressed/decrypted images. We considered both (a) access by authorized users ($p_i = p_j$), and (b) access by unauthorized users ($p_i \neq p_j$). Tables 2 and 3 list the decompressed/decrypted image quality obtained by authorized and unauthorized users for the Barbara and Mandrill images, respectively. From these tables, we can see that the decompressed/decrypted image quality obtained by an unauthorized user is very low regardless of the threshold ϵ_i. Figure 15 shows decompressed/decrypted image examples obtained by an unauthorized user for the Barbara and Mandrill images (ϵ_i=10.0). These results show that the encrypted images cannot be decrypted by an unauthorized user.

7. Conclusion and Future Work

In this paper, we proposed a secure OMP computation using a random unitary transform. We proved theoretically that the proposed method estimates exactly the same sparse representations as the non-encrypted version of OMP does. Furthermore, this method can control sparsity without decoding encrypted signals. Thus, we applied the method in an EtC system. The proposed secure EtC system achieves better rate-distortion performance than an overcomplete DCT does. The rate-distortion tradeoff can be controlled by adding atoms sequentially, without decoding the encrypted image.

Regarding the use of secure OMP in image compression, the experiments described herein are merely the first step. Further study is required to deploy the proposed secure OMP computation in EtC systems, including implementation of quantization and entropy coding.

References

[1] H.B. Barlow, "Possible principles underlying the transformations of sensory messages," Sensory Communication, pp.217–234, 1961.
[2] M. Elad, "Sparse and redundant representation modeling—what next?," IEEE Signal Process. Lett. , vol.19, no.12, pp.922–928, Dec. 2012.
[3] M. Elad, Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing, Springer, 2010.
[4] K. Skretting and K. Engan, "Image compression using learned dictionaries by RLS-DLA and compared with K-SVD," IEEE ICASSP, pp.1517–1520, 2011.
[5] O. Bryt and M. Elad, "Compression of facial images using the K-SVD algorithm," J. Visual Communication and Image Representation, vol.19, no.4, pp.270–282, May 2008.
[6] Y. Sun, X. Tao, Y. Li, and J. Lu, "Dictionary learning for image coding based on multisample sparse representation," IEEE Trans. Circuits Syst. Video Technol., vol.24, no.11, pp.2004–2010, Nov. 2014.
[7] X. Zhang, W. Lin, Y. Zhang, S. Wang, S. Ma, L. Duan, and W. Gao, "Rate-distortion optimized sparse coding with ordered dictionary for image set compression," IEEE Trans. Circuits Syst. Video Technol., vol.28, no.12, pp.3387–3397, Dec. 2018.
[8] M.D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M.E. Davies, "Sparse representations in audio and music: From coding to source separation," Proc. IEEE, vol.98, no.6, pp.995–1005, June 2010.
[9] H. Morioka, A. Kanemura, J. Hirayama, M. Shikauchi, T. Ogawa, S.

Ikeda, M. Kawanabe, and S. Ishii, "Learning a common dictionary for subject-transfer decoding with resting calibration," NeuroImage, vol.111, pp.167–178, 2015.
[10] F.J. Herrmann and G. Hennenfent, "Non-parametric seismic data recovery with curvelet frames," Geophysical Journal International, vol.173, no.1, pp.233–248, April 2008.
[11] C.T. Huang, L. Huang, Z. Qin, H. Yuan, L. Zhou, V. Varadharajan, and C-C.J. Kuo, "Survey on securing data storage in the cloud," APSIPA Transactions on Signal and Information Processing, vol.3, e7, 2014.
[12] W. Lu, S. Kawasaki, and J. Sakuma, "Using fully homomorphic encryption for statistical analysis of categorical, ordinal and numerical data," IACR Cryptology ePrint Archive, 2016:1163, 2016.
[13] Z. Brakerski, "Fundamentals of fully homomorphic encryption - A survey," Electronic Colloquium on Computational Complexity, report no.125, 2018.
[14] T. Araki, A. Barak, J. Furukawa, T. Lichter, Y. Lindell, A. Nof, K. Ohara, A. Watzman, and O. Weinstein, "Optimized honest-majority MPC for malicious adversaries - breaking the 1 billion-gate per second barrier," 2017 IEEE Symposium on Security and Privacy (SP), pp.843–862, May 2017.
[15] W. Yongjin and K.N. Plataniotis, "Face based biometric authentication with changeable and privacy preservable templates," Biometrics Symposium, pp.1–6, 2007.
[16] I. Nakamura, Y. Tonomura, and H. Kiya, "Unitary transform-based template protection and its application to l2-norm minimization problems," IEICE Trans. Inf. & Syst., vol.E99-D, no.1, pp.60–68, Jan. 2016.
[17] Y. Saito, I. Nakamura, S. Shiota, and H. Kiya, "An efficient random unitary matrix for biometric template protection," 2016 Joint 8th Int. Conf. Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), pp.366–370, 2016.
[18] T. Nakachi and H. Kiya, "Practical secure OMP computation and its application to image modeling," ACM IHIP2018.
[19] T. Nakachi, H. Ishihara, and H. Kiya, "Privacy-preserving network BMI decoding of covert spatial attention," IEEE Int. Conf. Signal Processing and Communication Systems (ICSPCS), pp.1–8, Cairns, Australia, 2018.
[20] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," 27th Asilomar Conference on Signals, Systems, and Computers, pp.40–44, 1993.
[21] W. Liu, W. Zeng, L. Dong, and Q. Yao, "Efficient compression of encrypted grayscale images," IEEE Trans. Image Process., vol.19, no.4, pp.1097–1102, April 2010.
[22] X. Zhang, "Lossy compression and iterative reconstruction for encrypted image," IEEE Trans. Inf. Forensics Security, vol.6, no.1, pp.53–58, March 2011.
[23] R. Hu, X. Li, and B. Yang, "A new lossy compression scheme for encrypted gray-scale images," IEEE Int. Conf. Acoustics, Speech Signal Process. (ICASSP), pp.7387–7390, 2014.
[24] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for JPEG/Motion JPEG standard," IEICE Trans. Fundamentals, vol.E98-A, no.11, pp.2238–2245, Nov. 2015.
[25] T. Chuman, K. Iida, W. Sirichotedumrong, and H. Kiya, "Image manipulation specifications on social networking services for encryption-then-compression systems," IEICE Trans. Inf. & Syst., vol.E102.D, no.1, pp.11–18. Jan. 2019.
[26] T. Chuman, W. Sirichotedumrong, and H. Kiya, "Encryption-then-compression systems using grayscale-based image encryption for JPEG images," IEEE Trans. Inf. Forensics Security, vol.14, no.6, pp.1515–1525, June 2019.
[27] B.K. Natarajan, "Sparse approximate solutions to linear systems," SIAM J. Computing, vol.24, no.2, pp.227–234, 1995.
[28] S.G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," IEEE Trans. Signal Processing, vol.41, no.12, pp.3397–3415, Dec. 1993.
[29] T.S. Lee, "Image representation using 2D Gabor wavelets," IEEE Trans. Pattern Anal. Mach. Intell., vol.18, no.10, pp.959–971, Oct. 1996.
[30] E. Candès and D. Donoho, "Curvelets: A surprisingly effective nonadaptive representation for objects with edges," ed. L.L. Schumaker et al., Curves and Surfaces, Vanderbilt University Press, 1999.
[31] K. Engan, S.O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," IEEE Int. Conf. Acoustics, Speech Signal Process. (ICASSP), pp.2443–2446, 1999.
[32] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," IEEE Trans. Signal Processing, vol.54, no.11, pp.4311–4322, Nov. 2006.

Takayuki Nakachi received a Ph.D. degree in electrical engineering from Keio University, Tokyo, Japan, in 1997. Since joining Nippon Telegraph and Telephone (NTT) Corporation in 1997, he has conducted research on super-high-definition image/video coding and media transport technologies. From 2006 to 2007, he was a visiting scientist at Stanford University. He also actively participates in MPEG international standardization activities. His current research interests include communication science, information theory, and signal processing. He received the 26th TELECOM System Technology Award, the 6th Paper Award of the Journal of Signal Processing, and the Best Paper Award at IEEE ISPACS2015. Dr. Nakachi is a member of the Institute of Electrical and Electronics Engineers (IEEE) and the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.

Hitoshi Kiya received B.E. and M.E. degrees from Nagaoka University of Technology, in 1980 and 1982, respectively, and a Dr.Eng. degree from Tokyo Metropolitan University in 1987. In 1982, he joined the faculty of Tokyo Metropolitan University, where he became a Full Professor in 2000. From 1995 to 1996, he worked at the University of Sydney, Australia, as a Visiting Fellow. He is a Fellow of the IEEE, IEICE, and ITE. He currently serves as the president-elect of APSIPA, and he served as the inaugural vice president (technical activities) of APSIPA from 2009 to 2013. Later, he was the regional director-at-large for region 10 of the IEEE Signal Processing Society from 2016 to 2017. He was also the president of the IEICE Engineering Sciences Society from 2011 to 2012, and he served there as a vice president and editor in chief for the IEICE Society Magazine and Society Publications. He has been an editorial board member of eight journals, including the IEEE Transactions on Signal Processing, Image Processing, and Information Forensics and Security; the chair of two technical committees; and a member of nine technical committees, including the APSIPA Image, Video, and Multimedia Technical Committee (TC) and the IEEE Information Forensics and Security TC. He has organized many international conferences, in such roles as the TPC chair of IEEE ICASSP 2012 and a general co-chair of IEEE ISCAS 2019. He has received numerous awards, including nine best paper awards.