# Privacy-Preserving SVM Computing in the Encrypted Domain

Takahiro Maekawa*, Ayana Kawamura*, Yuma Kinoshita* and Hitoshi Kiya*

*Tokyo Metropolitan University, Tokyo, 191-0065, Japan
maekawa-takahiro@tmu.ac.jp, kawamura-ayana@ed.tmu.ac.jp, kinoshita-yuma@tmu.ac.jp, kiya@tmu.ac.jp

***Abstract*—Privacy-preserving Support Vector Machine (SVM) computing scheme is proposed in this paper. Cloud computing has been spreading in many fields. However, the cloud computing has some serious issues for end users, such as unauthorized use and leak of data, and privacy compromise. We focus on templates protected by a block scrambling-based encryption scheme, and consider some properties of the protected templates for secure SVM computing, where templates mean features extracted from data. The proposed scheme enables us not only to protect templates, but also to have the same performance as that of unprotected templates under some useful kernel functions. Moreover, it can be directly carried out by using well-known SVM algorithms, without preparing any algorithms specialized for secure SVM computing. In an experiment, the pfroposed scheme is applied to a face-based authentication algorithm with SVM classifiers to confirm the effectiveness.**

***Index Terms*—Support Vector Machine, Privacy-preserving, EtC images**

## I. Introduction

Cloud computing and edge computing have been spreading in many fields, with the development of cloud services. However, the computing environment has some serious issues for end users, such as unauthorized use and leak of data, and privacy compromise, due to unreliability of providers and some accidents. While, a lot of studies on secure, efficient and flexible communications, storage and computation have been reported [1]–[4]. For securing data, full encryption with provable security (like RSA, AES, etc) is the most secure option. However, many multimedia applications have been seeking a trade-off in security to enable other requirements, e.g., low processing demands, retaining bitstream compliance, and flexible processing in the encrypted domain, so a lot of perceptual encryption schemes have been studied as one of the schemes for achieving a trade-off.

In the recent years, considerable efforts have been made in the fields of fully homomorphic encryption and multi-party computation [5]–[8]. However, these schemes can not be applied yet to SVM algorithms, although it is possible to carry out some simple statistical analysis of categorical and ordinal data. Moreover, the schemes have to prepare algorithms specialized for computing encrypted data.

Because of such a situation, we propose a privacy preserving SVM computing scheme in this paper . We focus on a block scrambling-based encryption scheme, which has been proposed for Encryption-then-Compression (EtC) systems with JPEG compression to consider the safety [9]–[13]. So far, the safety of the EtC systems has been evaluated based on the key space assuming the brute-force attacks, and the robustness against jigsaw puzzle attacks has been discussed [14]–[16]. In this paper, some new properties of templates encrypted by the encryption scheme are discussed under z-score normalization. It is also shown that the properties allow us to securely compute SVM algorithms without any degradation of the performances, even when some useful kernel functions are used. In addition, two key conditions are considered to enhance the encryption. In an experiment, the proposed scheme is applied to a face recognition algorithm with SVM classifiers to confirm the effectiveness.

## II. Preparation

### A. Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges, but it is mostly used in classification problems. In SVM, we input a feature vector $\boldsymbol{x}$ to the discriminant function as

$$y = \mathrm{sign}(\boldsymbol{\omega}^T \boldsymbol{x} + b)$$

with

$$\mathrm{sign}(u) = \begin{cases} 1 & (u > 1) \\ -1 & (u \leq 0) \end{cases}, \tag{1}$$

where $\boldsymbol{\omega}$ is a weight parameter, and $b$ is a bias.

SVM also has a technique called the kernel trick, which is a function that takes low dimensional input space and transform it to a higher dimensional space. These functions are called kernels. The kernel trick could be applied to Eq. (1) to map an input vector on further high dimension feature space, and then to linearly classify it on that space as

$$y = \mathrm{sign}(\boldsymbol{\omega}^T \phi(\boldsymbol{x}) + b). \tag{2}$$

The function $\phi(\boldsymbol{x}) : \mathbb{R}^d \to \mathcal{F}$ maps an input vector $\boldsymbol{x}$ on high dimension feature space $\mathcal{F}$, where $d$ is the number of the dimensions of features. In this case, feature space $\mathcal{F}$ includes parameter $\boldsymbol{\omega}$ ($\boldsymbol{\omega} \in \mathcal{F}$). The kernel function of two vectors $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ is defined as

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle, \tag{3}$$

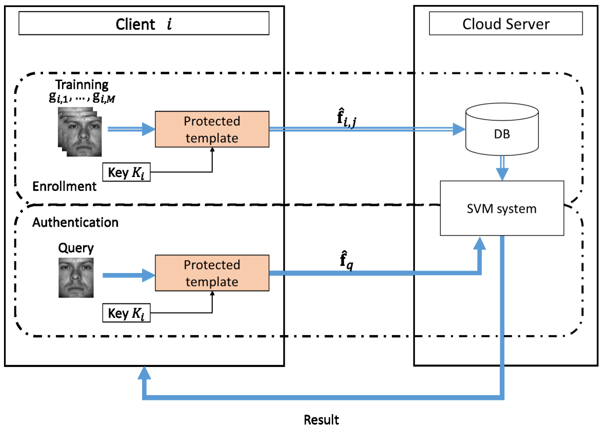



Fig. 1: Scenario

where $\langle \cdot, \cdot \rangle$ is an inner product. There are various kernel functions. For example, Radial Basis Function (RBF) kernel is given by

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\Upsilon \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2) \tag{4}$$

and polynomial kernel is provided by

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \boldsymbol{x}_i^T \boldsymbol{x}_j)^l, \tag{5}$$

where $\Upsilon$ is a high parameter to decide the complexity of boundary determination, $l$ is a parameter to decide the degree of the polynomial, and $T$ indicates transpose.

This paper aims to propose a new framework to carry out SVM with protected vectors.

### B. Scenario

Figure 1 illustrates the scenario used in this paper. In the enrollment, client $i$, $i \in \{1, 2, ..., N\}$, prepares training samples $\mathrm{g}_{i,j}, j \in \{1, 2, ..., M\}$ such as images, and a feature set $\mathbf{f}_{i,j}$, called a template, is extracted from the samples. Next the client creates a protected template set $\hat{\mathbf{f}}_{i,j}$ by a secret key $p_i$ and sends the set to a cloud server. The server stores it and implements learning with the protected templates for a classification problem.

In the authentication, Client $i$ creates a protected template as a query and sends it to the server. The server carries out a classification problem with a learning model prepared in advance, and then returns the result to Client $i$.

Note that the cloud server has no secret keys and the classification problem can be directly carried out by using well-known SVM algorithms. In the other words, the server does not have to prepare any algorithms specialized for the classification in the encrypted domain.

### C. Template Protection

Template protection schemes based on unitary transformations have been studied as one of methods for cancelable biometrics [17]–[21]. This paper has been inspired by those studies.

A template $\mathbf{f}_{i,j} \in \mathbb{R}^d$ is protected by a unitary matrix having randomness with a key $p_i$, $\mathbf{Q}_{p_i} \in \mathbb{C}^{N \times N}$ as,

$$\hat{\mathbf{f}}_{i,j} = T(\mathbf{f}_{i,j}, p_i) = \mathbf{Q}_{p_i} \mathbf{f}_{i,j}, \tag{6}$$

where $\hat{\mathbf{f}}_{i,j}$ is the protected template. Various generation schemes of $\mathbf{Q}_{p_i}$ have been studied to generate unitary or orthogonal random matrices such as Gram-Schmidt method, random permutation matrices and random phase matrices [18], [19]. Security analysis of the protection schemes have been also considered in terms of brute-force attacks, diversity and irreversibility. Protected templates generated according to Eq. (6) have the following properties under $p_i = p_s$ [19].

Property 1 : Conservation of the Euclidean distances:

$$\|\mathbf{f}_{i,j} - \mathbf{f}_{s,t}\|^2 = \|\hat{\mathbf{f}}_{i,j} - \hat{\mathbf{f}}_{s,t}\|^2.$$

Property 2 : Conservation of inner products:

$$\langle \mathbf{f}_{i,j}, \mathbf{f}_{s,t} \rangle = \langle \hat{\mathbf{f}}_{i,j}, \hat{\mathbf{f}}_{s,t} \rangle,$$

.

Property 3 : Conservation of correlation coefficients:

$$\frac{\langle \mathbf{f}_{i,j}, \mathbf{f}_{s,t} \rangle}{\sqrt{\langle \mathbf{f}_{i,j}, \mathbf{f}_{s,t} \rangle} \sqrt{\langle \mathbf{f}_{i,j}, \mathbf{f}_{s,t} \rangle}} = \frac{\langle \hat{\mathbf{f}}_{i,j}, \hat{\mathbf{f}}_{s,t} \rangle}{\sqrt{\langle \hat{\mathbf{f}}_{i,j}, \hat{\mathbf{f}}_{s,t} \rangle} \sqrt{\langle \hat{\mathbf{f}}_{i,j}, \hat{\mathbf{f}}_{s,t} \rangle}},$$

where $\mathbf{f}_{s,t}$ is a template of another client $s$, $s \in \{1, 2, ..., N\}$, who has $M$ training samples $\mathrm{g}_{s,t}$, $t \in \{1, 2, ..., M\}$.

## III. Proposed Framework

In this paper, we focus on block scramble-based image encryption schemes, which have been proposed for EtC systems [10]–[13], to generate protected templates. New properties of the protected templates are discussed under some kernel functions for SVM computing.

### A. Block scrambling-based image encryption

In the Block scrambling-based image encryption scheme [10]–[13]. an image with $X \times Y$ pixels is first divided into non-overlapped blocks with $B_x \times B_y$, then block-based processing steps, as illustrated in Fig. 2, are applied to the divided image. The procedure of the image encryption is given as follows:

Step1: Divide an image with $X \times Y$ pixels into blocks with $B_x \times B_y$ pixels, and permute randomly the divided blocks using a random integer generated by a secret key $K_1$, where $K_1$ is commonly used for all color components.

Step2 Rotate and invert randomly each block using a random integer generated by a key $K_2$, where $K_2$ is commonly used for all color components as well.

Step3: Apply the negative-positive transformation to each block using a random binary integer generated by a key $K_3$, where $K_3$ is commonly used for all color

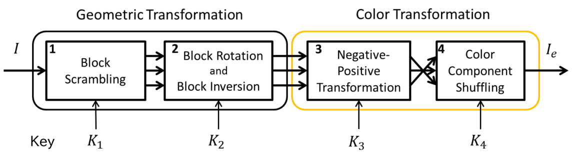



Fig. 2: Block scrambling-based image encryption

components. In this step, a transformed pixel value in $i$th block $B_i, p_0$ is computed by

$$p' = \begin{cases} p & (r(i) = 0) \\ 255 - p & (r(i) = 1) \end{cases} \tag{7}$$

where $r(i)$ is a random binary integer generated by $K_3$ under the probability $P(r(i)) = 0.5$ and $p \in B_i$ is the pixel value of an original image with 8 bpp.

Step4: Shuffle three color components in each block (the color component shuffling) using a random senary integer generated by a key $K_4$.

The key space of the block scrambling-based image encryption is generally large enough against the brute-force attacks [11]. Moreover, jigsaw puzzle solver attacks, which utilize the correlation among pixels in each block, have been considered [14]–[16]. It has been confirmed that the appropriate selection of the block size and the combination of each encryption step can improve the robustness of EtC systems against various attacks. In this paper, images encrypted by the above steps are referred to as EtC images.

### B. Properties of EtC images

Let us consider some properties of EtC images. It will be shown that generating EtC images is reduced to a unitary transformation.

#### 1) Without any normalization

In the encryption steps in Fig.2, block scramble, block rotation and inversion, and color component shuffling are an operation to permute pixels, so they are represented by using a permutation matrix. For example a permutation matrix $\mathbf{E}_{\boldsymbol{B_i}}$ is give as

$$\mathbf{E}_{\boldsymbol{B_i}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \tag{8}$$

where $\mathbf{E}_{\boldsymbol{B_i}}$ has only one element of 1 in each row or each column and others are 0 . $\mathbf{E}_{\boldsymbol{B_i}}$ satisfies the relation,

$$\mathbf{I} = {\mathbf{E}_{\boldsymbol{B_i}}}^T \mathbf{E}_{\boldsymbol{B_i}} \tag{9}$$

where $\mathbf{I}$ is an identity matrix. Therefore $\mathbf{E}_{\boldsymbol{B_i}}$ becomes an orthogonal matrix.

The protected template $\hat{\mathbf{f}}_{i,j}$ is computed by using $\mathbf{E}_{\boldsymbol{B_i}}$ as,

$$\hat{\mathbf{f}}_{i,j} = \mathbf{E}_{\boldsymbol{B_i}} \mathbf{f}_{i,j}. \tag{10}$$

However, in the case of using negative-positive transformation, we can not obtain the above relation. When $k$ th element of $\mathbf{f}_{i,j}$ is given as $p_{i,j}(k)$, the one of $\hat{\mathbf{f}}_{i,j}$ is expressed as $255 - p_{i.j}$. Then we can obtain the relation,

$$\begin{aligned} & ||(255 - p_{i,j}(k)) - (255 - p_{s,t}(k))||^2 \\ = \ & || - p_{i,j}(k) + p_{s,t}||^2 \\ = \ & ||p_{i,j}(k) - p_{s,t}(k)||^2. \end{aligned} \tag{11}$$

This relation shows that the Euclidean distance is preserved. However, each component of the inner product is calculated as

$$\begin{aligned} & (255 - p_{i,j}(k)) * (255 - p_{s,t}(k)) \\ = \ & 255^2 - 255(p_{i,j}(k) + p_{s,t}(k)) + p_{i,j}(k) \times p_{s,t}(k) \\ \neq \ & p_{i,j}(k) * p_{s,t}(k) \end{aligned} \tag{12}$$

Thus, the inner product is not preserved. Consequently, block scramble, block rotation and inversion, color component shuffling preserve inner products and Euclidean distance between original images and EtC images, but negative-positive transformation preserves only the Euclidean distance.

#### 2) With z-score normalization

Next we consider the case of using z-score normalization [22]. In z-score normalization, a value $x_i$, $i = 1, 2, \ldots, N$ is replaced with $z_i$ like

$$z_i = \frac{(x_i - \bar{X})}{S}, \tag{13}$$

where $\bar{X}$ is the mean value of the data and $S$ is the standard deviation,

$$S = \sqrt{\frac{\sum_{i=1}^{N} (x_i - \bar{X})^2}{N}}. \tag{14}$$

In negative-positive transformation, Eq.(13) is given as

$$\begin{aligned} \hat{z}_{i,j}(k) &= \frac{(255 - p_{i,j}(k)) - (255 - \bar{P}_k)}{S'} \\ &= -\frac{p_{i.j}(k) - \bar{P}_k}{S} \\ &= -z_{i,j}(k) \end{aligned} \tag{15}$$

where

$$\bar{P}_k = \frac{1}{N \times M} \sum_{i=1}^{N} \sum_{j=1}^{M} p_{i,j}(k) \tag{16}$$

$$\begin{aligned} S' &= \sqrt{\frac{\sum_{i=1}^{N} \sum_{j=1}^{M} ((255 - p_{i,j}(k)) - (255 - \bar{P}_k))^2}{N \times M}} \\ &= \sqrt{\frac{\sum_{i=1}^{N} \sum_{j=1}^{M} (-p_{i,j}(k) + \bar{P}_k)^2}{N \times M}} \\ &= S. \end{aligned} \tag{17}$$

Eq.(15) means that a normalized value $\hat{z}_{i,j}$ becomes the sign inversed value of $z_{i,j}$. A sign inversion matrix is a unitary matrix, so the inner product is preserved. Hence, in the case of adopting z-score normalization to EtC templates, negative-positive transformation, in addtion to block scramble, block rotation and inversion, and color component shuffling, allows us to keep the inner products.

### C. Classes of kernels

We consider applying the protected templates to a kernel function. In the case of using RBF kernel, the following relation is satisfied from property 1 and Eq.(4)

$$\begin{aligned} K(\hat{\mathbf{f}}_{i,j}, \hat{\mathbf{f}}_{s,t}) &= \exp(-\Upsilon \|\hat{\mathbf{f}}_{i,j} - \hat{\mathbf{f}}_{s,t}\|^2) \\ &= K(\mathbf{f}_{i,j}, \mathbf{f}_{s,t}) \end{aligned} \tag{18}$$

A stationary kernel $K_S(\boldsymbol{x}_i - \boldsymbol{x}_j)$ is one which is translation invariant:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = K_S(\boldsymbol{x}_i - \boldsymbol{x}_j), \tag{19}$$

that is, it depends only on the lag vector separating the two vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Moreover, when a stationary kernel depends only on the norm of the lag vectors between two vectors, the kernel $K_I(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|)$ is said to be isotropic (or homogeneous) [23], and is thus only a function of distance:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = K_I(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|). \tag{20}$$

For examples, RBF, WAVE and Rational quadratic kernels belong to this class, i.e, isotropic stationary kernel, called kernel class 1 in this paper. If kernels are isotropic, the propose scheme is useful under the kernels.

Besides, from property 3, we can also use a kernel $K_{In}(\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle)$ that depends only on the inner products between two vectors given as

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = K_{In}(\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle). \tag{21}$$

Polynomial kernel and linear kernel are in this class, referred to as class 2.

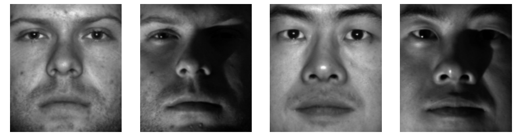
Some kernels such as Fisher and p-spectrum ones, to which the protected templates can not be applied, belong to other classes. We focus on using kernel class 1 and class 2.

### D. Dual problem

Next, we consider binary classification that is the task of classifying the elements of a given set. A dual problem to implement a SVM classifier with protected templates is expressed as

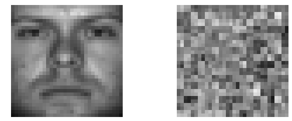
$$\begin{aligned} &\max_{\alpha} \left( -\frac{1}{2} \sum_{\substack{i,s \in N \\ j,t \in M}} \alpha_{i,j}\alpha_{s,t} y_{i,j} y_{s,t} \langle \phi(\hat{\mathbf{f}}_{i,j}), \phi(\hat{\mathbf{f}}_{s,t}) \rangle + \sum_{\substack{i \in N \\ j \in M}} \alpha_{i,j} \right) \\ &s.t. \sum_{\substack{i \in N \\ j \in M}} \alpha_{i,j} y_{i,j} = 0, 0 \le \alpha_{i,j} \le C, \end{aligned} \tag{22}$$

where $y_{i,j}$ and $y_{s,t} \in \{+1, -1\}$ are correct labels for each training data, $\alpha_{i,j}$ and $\alpha_{s,t}$ are dual variables and C is a regular coefficient. If we use kernel class 1 or class 2 described above, the inner product $\langle \phi(\hat{\mathbf{f}}_{i,j}), \phi(\hat{\mathbf{f}}_{s,t}) \rangle$ is equal to $K(\mathbf{f}_{i,j}, \mathbf{f}_{s,t})$. Therefore, even in the case of using protected templates, the dual problem with protected templates is reduced to the same problem as that of the original templates. This conclusion means that the use of the proposed templates gives no effect to the performance of the SVM classifier under kernel class 1 and class 2.



(a) person1 (b) person2

Fig. 3: Examples of Extended Yale Face Database B



(a) template (b) protected

Fig. 4: An example of protection

### E. Relation among keys

As shown in Fig 1, a protected template $\hat{\mathbf{f}}_{i,j}$ is generated from training data $\mathrm{g}_{i,j}$ by using a key $p_i$. Two relations among keys are summarized, here.

*1) Key condition 1:* $p_1 = p_2 = ... = p_N$

The first key choice is to use a common key in all clients, namely, $p_1 = p_2 = ... = p_N$. In this case, all protected templates satisfy the properties described in II-C and III-B, so the SVM classifier has the same performance as that of using the original templates.

*2) Key condition 2:* $p_1 \neq p_2 \neq ... \neq p_N$

The second key choice is to use a different key in each client, namely $p_1 \neq p_2 \neq ... \neq p_N$. In this case, the three properties are satisfied only among templates with a common key. This key condition allows us to enhance the robustness of the security against various attacks as discussed later.

## IV. Experimental Results

The propose scheme was applied to face recognition experiments which were carried out as a dual problem.

### A. Data Set

We used Extended Yale Face Database B [24] that consists of 2432 frontal facial images with $192 \times 168$-pixels of $N = 38$ persons like Fig. 3. 64 images for each person were divided into half randomly for training data samples and queries. We used block scrambling-based image encryption, which consists of block scrambling, block rotation and inversion, negative-positive transformation, and color component shuffling steps, to produce protected templates. Besides, RBF kernel and linear kernel were used, where they belong to kernel class 1 and class 2, respectively. The protection was applied to templates with 1024 dimensions generated by the down-sampling method [25]. The down-sampling method divides an image into non-overlapped blocks and then calculates the mean value in each block. Figure 4 shows examples of an original template and the protected one. In the experiment, z-score normalization was applied to all templates after the encryption.

### *B. Results and Discussion*

In face recognition with SVM classifiers, one classifier is created for each enrollee. The classifier outputs a predicted class label and a classification score for each query template $\hat{\mathbf{f}}_q$, where $\hat{\mathbf{f}}_q$ is a protected template generated from the template of a query, $\mathbf{f}_q$. The classification score is the distance from the query to the boundary ranging. The relation between the classification score $S_q$ and a threshold $\tau$ for the positive label of $\mathbf{f}_q$ is given as

$$if\ S_q \geq \tau\ then\ accept;\ else\ reject. \tag{23}$$

In the experiment, False Reject Rate (FRR), False Accept Rate (FAR), and Equal Error Rate (EER) at which FAR is equal to FRR were used to evaluate the performance.

*1)* $p_1 = p_2 = ... = p_N$

Figure 5 shows results in the case of using key condition 1. The results demonstrate that SVM classifiers with protected templates (protected in Fig 5) had the same performances as those fo SVM classifiers with the original templates (not protected in Fig 5). From the results, it is confirmed that the proposed framework gives no effect to the performance of SVM classifiers under key condition 1 and z-score normalization.

*2)* $p_1 \neq p_2 \neq ... \neq p_N$

Figure 6 shows results in the case of using key condition 2. In this condition, it is expected that a query will be authenticated only when it meets two requirements, i.e. the same key and the same person, although only the same person is required under key condition 1. Therefore, the performances in Fig. 6 were slightly different from those in Fig. 5, so the FAR performances for key condition 2 were better due to the strict requirements.

*3) Unauthorized outflow ($p_1 \neq p_2 \neq ... \neq p_N$)*

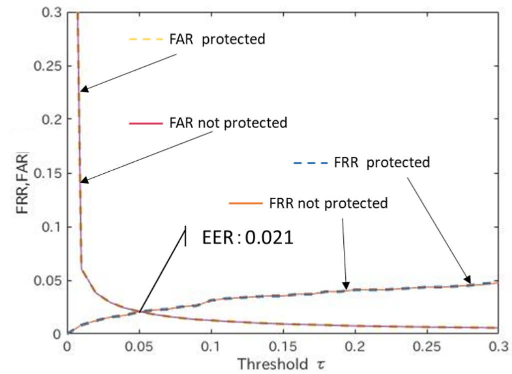
Figure 7 shows the FAR performance in the case that a key $p_i$ leaks out. In this situation, other clients could use the key $p_i$ without any authorization as spoofing attacks. As shown in Fig.7, the FAR (key leaked in Fig.7) still had low vales due to two requirements, although it was slightly degraded, compared to Fig.6.

Figure 8 is the FAR performance in the case that a template $\mathbf{f}_{i,j}$ leaks out. It is confirmed that the FAR (template leaked in Fig.8) still had low vales as well as in Fig.7.

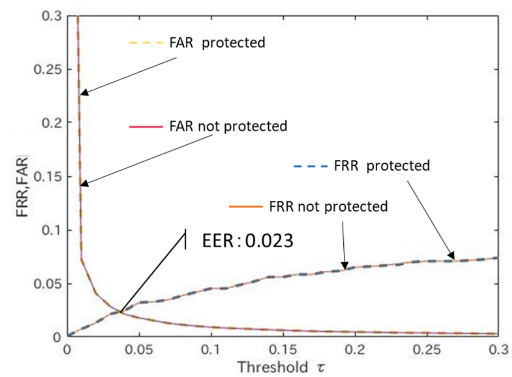
From these results, the use of key condition 2 enhances the robustness of the security against spoofing attacks.

## V. CONCLUSION

In this paper, we proposed a privacy-preserving SVM computing scheme with protected templates. It was shown that block scrambling-based image encryption, which has been proposed for EtC systems, is reduced to a unitary transform under the z-score normalization. It was also confirmed that templates protected by the encryption has some useful properties, and the properties allow us to securely compute SVM algorithms without any degradation of the performances, even when some useful kernel functions are used. Besides, two key conditions were considered to enhance the robustness of the security against various attacks. Some face-based authentication experiments using SVM classifiers were also demonstrated to experimentally confirm the effectiveness of the proposed frame work.




(a) Linear kernel ($C = 1$)




(b) RBF kernel ($C = 34$, $\Upsilon = 81$)

Fig. 5: FAR and FFR ($p_1 = p_2 = ... = p_N$)

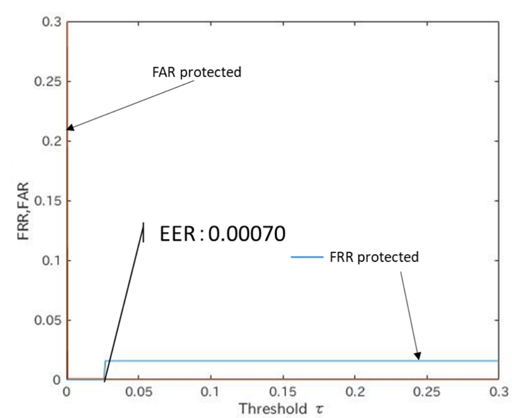



Fig. 6: FAR and FAR (RBF kernel, $p_1 \neq p_2 \neq ... \neq p_N$)

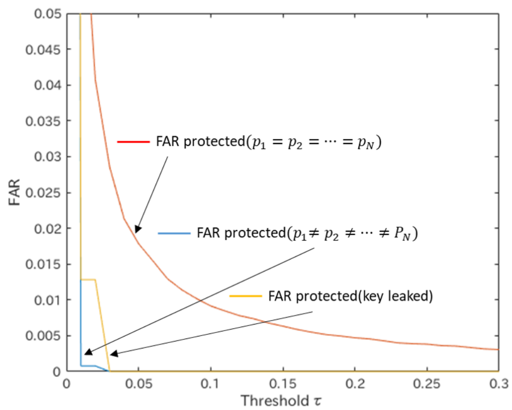



Fig. 7: FAR with leaked keys (RBF kernel)

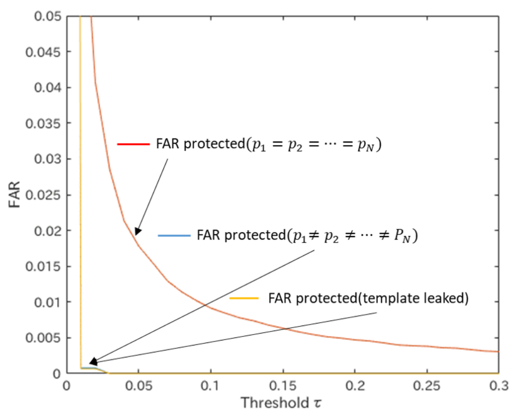



Fig. 8: FAR with leaked original templates (RBF kernel)

*Acknowledgements*

This work was partially supported by Grant-in-Aid for Scientific Research(B), No.17H03267, from the Japan Society for the Promotion Science.

## References

[1] 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, 2014.
[2] R. Lazzeretti and M. Barni, "Private computing with garbled circuits [applications corner]," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 123–127, 2013.
[3] M. Barni, G. Droandi, and R. Lazzeretti, "Privacy protection in biometric-based recognition systems: A marriage between cryptography and signal processing," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 66–76, 2015.
[4] R. L. Lagendijk, Z. Erkin, and M. Barni, "Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 82–105, 2013.
[5] 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," in *IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 843–862.
[6] T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara, "High-throughput semi-honest secure three-party computation with an honest majority," in *Proceedings of ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 805–817.
[7] W. Lu, S. Kawasaki, and J. Sakuma, "Using fully homomorphic encryption for statistical analysis of categorical, ordinal and numerical data," in *IACR Cryptology ePrint Archive*, vol. 2016, 2016, p. 1163.
[8] Y. Aono and T. Hayashi and L. Phong and L. Wang, "Privacy-preserving logistic regression with distributed data sources via homomorphic encryption," *IEICE Transactions on Information and Systems*, vol. E99.D, no. 8, pp. 2079–2089, 2016.
[9] O. Watanabe, A. Uchida, T. Fukuhara, and H. Kiya, "An encryption-then-compression system for jpeg 2000 standard," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 1226–1230.
[10] K. Kurihara, S. Shiota, and H. Kiya, "2015 an encryption-then-compression system for jpeg standard," in *Picture Coding Symposium (PCS)*, 2015, pp. 119–123.
[11] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for jpeg/motion jpeg standard," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E98.A, no. 11, pp. 2238–2245, 2015.
[12] K. Kurihara, O. Watanabe, and H. Kiya, "An encryption-then-compression system for jpeg xr standard," in *2016 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 2016, pp. 1–5.
[13] K. Kurihara, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for lossless image compression standards," *IEICE Transactions on Information and Systems*, vol. E100.D, no. 1, pp. 52–56, 2017.
[14] T. Chuman, K. Kurihara, and H. Kiya, "On the security of block scrambling-based etc systems against jigsaw puzzle solver attacks," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2157–2161.
[15] T. Chuman, K. Kurihara, and H. Kiya, "Security evaluation for block scrambling-based etc systems against extended jigsaw puzzle solver attacks," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, 2017, pp. 229–234.
[16] W. Sirichotedumrong, T. Chuman, S. Imaizumi, and H. Kiya, "Grayscale-based block scrambling image encryption for social networking services," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*, 2018.
[17] C. Rathgeb, and A. Uhl, "A survey on biometric cryptosystems and cancelable biometrics," in *EURASIP J. Information Security*, vol. 2011, no. 1, 2011, pp. 1–25.
[18] I. Nakamura, Y. Tonomura, and H. Kiya, "Unitary transform-based template protection and its properties," in *European Signal Processing Conference*, vol. SIPA-P3.4, 2015, pp. 2466–2470.
[19] I. Nakamura, Y. Tonomura, and H. Kiya, "Unitary transform-based template protection and its application to l2-norm minimization problems," in *IEICE Transactions on Information and Systems*, vol. E99-D, no. 1, 2016, pp. 60–68.
[20] K. Nandakumar, A. K. Jain, "Biometric template protection: Bridging the performance gap between theory and practice," in *Signal Processing Magazine, IEEE*, vol. 32, no. 5, 2015, pp. 88–100.
[21] S. Rane, "Standardization of biometric template protection," in *Signal Processing Magazine, IEEE*, vol. 21, no. 4, 2014.
[22] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern Recognition*, vol. 38, no. 12, pp. 2270 – 2285, 2005.
[23] M. G. Genton, "Classes of kernels for machine learning: A statistics perspective," *J. Mach. Learn. Res.*, vol. 2, pp. 299–312, 2002.
[24] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," in *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, 2001, pp. 643–660.
[25] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," in *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, 2009.