IEICE TRANS. INF. & SYST., VOL.E93–D, NO.3 MARCH 2010

PAPER Special Section on Trust, Security and Privacy for Pervasive Applications

Hill-Climbing Attacks and Robust Online Signature Verification Algorithm against Hill-Climbing Attacks

Daigo MURAMATSU[†a)], ***Member***

SUMMARY Attacks using hill-climbing methods have been reported as a vulnerability of biometric authentication systems. In this paper, we propose a robust online signature verification algorithm against such attacks. Specifically, the attack considered in this paper is a hill-climbing forged data attack. Artificial forgeries are generated offline by using the hill-climbing method, and the forgeries are input to a target system to be attacked. In this paper, we analyze the menace of hill-climbing forged data attacks using six types of hill-climbing forged data and propose a robust algorithm by incorporating the hill-climbing method into an online signature verification algorithm. Experiments to evaluate the proposed system were performed using a public online signature database. The proposed algorithm showed improved performance against this kind of attack.

key words: *hill-climbing attack, online signature verification, biometrics, vulnerability, offline attack*

1. Introduction

Biometric person authentication technologies are becoming more important in the drive to ensure security. These technologies are being actively studied, and some of them are being used in real situations. However, several vulnerabilities have been reported [1], [2], including a hill-climbing attack [3]. Hill-climbing attacks against biometric systems of several modalities, such as face [4]–[7], fingerprints [8], [9], and online signatures [10], [11] have been reported.

There are two types of hill-climbing attack:

1. Online hill-climbing attack: Attackers access the targeted biometric system directly and attack it repeatedly [4]–[11].
2. Offline hill-climbing attack (*hill-climbing forged data attack*): Attackers repeatedly access a physically different biometric system from the targeted biometric system and generate hill-climbing forged data. Then, the forged data is input to the targeted biometric system.

Several countermeasures against online attacks have been reported, for example, limiting the number of sequential attempts [3], score transformation [12], and so on. Also, a parameter-updating method [13], [14] is one possible solution. However, these measures are not useful against an offline attack because attackers do not need to access the targeted biometric system repeatedly; rather, they input forged data to the targeted biometric system only once. Thus, different countermeasures are necessary against an offline attack. However, no useful countermeasures for any type of modality have been proposed.

In this paper, we propose a countermeasure against offline hill-climbing attacks for online signature verification. A robust online signature verification algorithm against such attacks was implemented by incorporating a hill-climbing algorithm into a verification algorithm.

Experiments were performed using a public online signature database, SVC2004 [15]. Experimental results show that the proposed algorithm had improved performance against hill-climbing forged data attacks.

We first describe a basic online signature verification algorithm in Sect. 2.1. In Sect. 2.2, we explain why the hill-climbing attack is a menace for online signature verification, and in Sect. 2.3, we propose a robust online signature verification algorithm against hill-climbing attacks. We performed several experiments using a public database. In Sects. 3.1 and 3.2, we explain details of the algorithm for online signature verification and hill-climbing used in the experiment, and we report and discuss the experimental results in the rest of Sect. 3. Finally, we conclude this paper in Sect. 4.

2. Methodology

2.1 Basic Online Signature Verification Algorithm

Before describing hill-climbing attacks and our proposed system, let us review a basic online signature verification algorithm. Online signature verification is an automatic biometric person authentication method that uses data obtained while a signature is being written. This method has been extensively studied [16]–[20] and several compatitions were held [15], [21]–[23].

Online verification algorithms can be classified into two approaches: parameter-based and function-based [16] (or feature-based and function-based [20]). In this paper, we focus on a function-based algorithm that utilizes dynamic time warping [24] to calculate dissimilarity scores.

A basic function-based algorithm is depicted in Fig. 1. The algorithm consists of an enrollment phase and a verification phase.

All online signatures are represented as elements of a set S. Let $GS_E \subset S^M$ be a set of (sequences of) genuine signatures submitted during the enrollment phase, and GS_V and $FS \subset S$ be sets consisting of the genuine signatures and forged signatures presented during the verification phase.

Manuscript received July 1, 2009.
Manuscript revised October 8, 2009.

[†]The author is with the Department of Electrical and Mechanical Engineering, Seikei University, Musashino-shi, 180–8633 Japan.
a) E-mail: muramatsu@st.seikei.ac.jp

DOI: 10.1587/transinf.E93.D.448

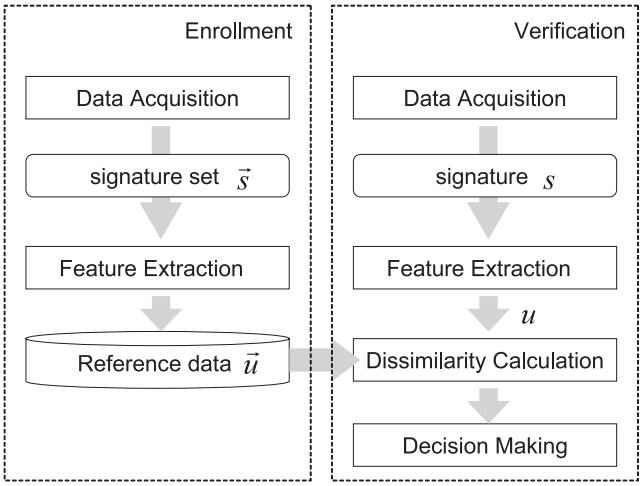

Fig. 1 Basic online signature verification algorithm.

Let U be a feature space, and $\mathcal{E} : S \to U$ be feature extracttion from the set S to a set U. Define $\vec{\mathcal{E}} : S^M \to U^M$ by

$$\begin{aligned}\vec{\mathcal{E}}(\vec{s}) &= (\mathcal{E}(s_i))_{1\le i\le M} \\ &= (u_i)_{1\le i\le M} \in U^M\end{aligned} \tag{1}$$

for any $\vec{s} = (s_i)_{1\le i\le M} \in GS_E$. For any $s \in GS_V$ (or FS) and any $\vec{s} \in \vec{S}$, let $Dscore(s, \vec{s}; \mathcal{E})$ denote a dissimilarity score s presented during the verification phase for the enrolled online signature reference data $\vec{s}$, associated with feature extraction.

2.1.1 Enrollment Phase

In this phase, a user provides signatures for enrollment. From these signatures, $\vec{s}$ are obtained, and time-series feature vectors $\vec{u}$ are extracted and enrolled as reference data.

2.1.2 Verification Phase

In this phase, a user submits a signature s. Then a time-series feature vector u is extracted, and a decision is made as to whether s is

$$\begin{cases} \text{accepted} & \text{if } Dscore(s, \vec{s}; \mathcal{E}) < \Theta_{NS} \\ \text{rejected} & \text{otherwise} \end{cases}, \tag{2}$$

where θ_{NS} is a threshold for decision making.

2.2 Hill-Climbing Attack

A schematic diagram of the hill-climbing (HC) forged data attack is shown in Fig. 2. This attack is an offline attack; thus, there are two systems: a targeted system and an HC-attacked system.

Targeted system This system is the one that attackers want to attack.

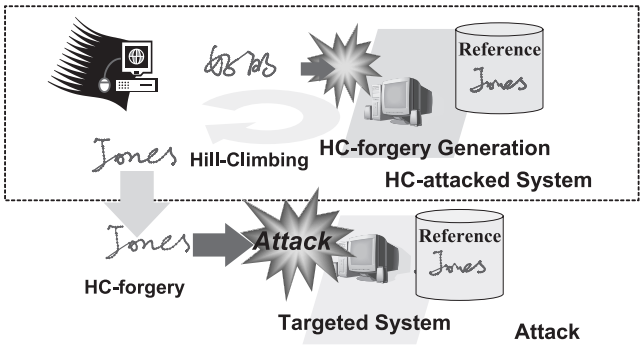

Fig. 2 Hill-climbing forged data attack.

HC-attacked system This system is a physically different one from the targeted system. Hill-climbing attacks are performed against this system to generate hill-climbing forged data (*HC forgery*).

The hill-climbing forged attack is performed in two steps: an HC-forgery generation step and an attack step.

2.2.1 HC-Forgery Generation Step

Let $sf \in FS$ and $sg \in GS_V$ be a forged signature and a genuine signature, respectively. Let $\vec{s}$ be a genuine signature set used for reference generation in the HC-attacked system and $\vec{s'}$ be a genuine signature set used in the target system. Let $\mathcal{E}$ and $\mathcal{E}'$ be feature extraction schemes used in the HC-attacked system and the target system, respectively. The forged signature sf is input to the HC-attacked system, and an HC forgery is generated using hill-climbing methods. Let $\mathcal{H} : S \to S$ be a modification from the set S to S, and let $sf_{\mathcal{E}}^{(i)}(\vec{s})$ be the i-th modified signature from sf, described by

$$\begin{aligned}sf_{\mathcal{E}}^{(i)}(\vec{s}) &= \mathcal{H}(sf_{\mathcal{E}}^{(i-1)}, \vec{s}; \mathcal{E}) \\ &= \mathcal{H}^{(i)}(sf, \vec{s}; \mathcal{E}).\end{aligned} \tag{3}$$

The modification is designed so that

$$Dscore(sf_{\mathcal{E}}^{(i+1)}(\vec{s}), \vec{s}; \mathcal{E}) < Dscore(sf_{\mathcal{E}}^{(i)}(\vec{s}), \vec{s}; \mathcal{E}). \tag{4}$$

By repeating the modification, an attacker can decrease the dissimilarity score to a converged score, obtaining converged hill-climbing forged data $sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s})$ described by

$$sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s}) = \mathcal{H}^{(I_{cvg})}(sf, \vec{s}; \mathcal{E}).$$

Here,

$$I_{cvg} = \min_i \{i | Dscore(sf_{\mathcal{E}}^{(i)}(\vec{s}), \vec{s}; \mathcal{E}) = Dscore(sf_{\mathcal{E}}^{(i-L)}(\vec{s}), \vec{s}; \mathcal{E})\},$$

and L is a parameter for the convergence test.

2.2.2 Attack Step

The generated HC forgery $sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s})$ is input to the targeted

system only once. No hill-climbing method is performed in this step. Thus, countermeasures for a direct hill-climbing attack (e.g. [3], [12]) are not useful against a hill-climbing forged data attack. The decision is made based on (2); thus, $sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s})$ is falsely accepted if $Dscore(sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s}), \vec{s}'; \mathcal{E}')$ is less than the threshold.

2.3 Robust Algorithm for Hill-Climbing Attack

The reason why a hill-climbing forged data attack becomes a menace is that the dissimilarity scores $Dscore(sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s}), \vec{s}'; \mathcal{E}')$ of many $sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s})$ generated from sf are less than the dissimilarity scores $Dscore(sg, \vec{s}'; \mathcal{E}')$ of genuine signatures sg. Therefore, the system cannot make decisions correctly by using (2). If there is a way to make the dissimilarity scores of genuine signatures less than $Dscore(sf_{\mathcal{E}}^{(I_{cvg})}(\vec{s}), \vec{s}'; \mathcal{E}')$, a robust algorithm against a hill-climbing forged data attack can be generated.

We applied a hill-climbing method to modify genuine signatures $sg_i \in GS_V, i = 1, 2, \ldots, 5$ and observed changes of the dissimilarity scores[†]. Figure 3 shows changes of dissimilarity scores as a function of iteration number. Solid lines indicate the changes of dissimilarity scores of sg. For comparison, changes of dissimilarity scores of random forgeries $sf_i \in FS, i = 1, 2, \ldots, 5$ are also illustrated in broken lines. Initial dissimilarity scores (iteration number = 0) of sg_i and sf_i represent those of raw genuine signatures $Dscore(sg_i, \vec{s}'; \mathcal{E}')$ and raw forgeries $Dscore(sf_i, \vec{s}'; \mathcal{E}')$. All dissimilarity scores of both types of signatures are improved and converged. However, hill-climbing methods cannot find a global optimum, but only a local optimum. The converged scores of sf_i are exactly $Dscore(sf_{i\mathcal{E}'}^{(I_{cvg})}(\vec{s}'), \vec{s}'; \mathcal{E}')$. Although the initial dissimilarity scores $Dscore(sg_i, \vec{s}'; \mathcal{E}')$ are larger than $Dscore(sf_{i\mathcal{E}'}^{(I_{cvg})}(\vec{s}'), \vec{s}'; \mathcal{E}')$, the converged dissimilarity scores of sg_i (dissimilarity scores of hill-climbing genuine $sg_{i\mathcal{E}'}^{(I_{cvg})}(\vec{s}')$ $Dscore(sg_{i\mathcal{E}'}^{(I_{cvg})}(\vec{s}'), \vec{s}'; \mathcal{E}')$ are less than $Dscore(sf_{\mathcal{E}'}^{(I_{cvg})}(\vec{s}'), \vec{s}'; \mathcal{E}')$. Converged scores can be useful for rejecting hill-climbing forged data.

Therefore, we propose an algorithm that uses a hill-climbing method, as depicted in Fig. 4. The difference between the proposed algorithm and the basic one depicted in Fig. 1 is that a data modification step is incorporated into the verification phase and this data modification is performed repeatedly. In the data modification step, a time-series feature vector is modified so as to decrease a dissimilarity score, and the data modification and dissimilarity calculation are repeated until the dissimilarity score converges. By repeating these steps, hill-climbing is performed, and a converged dissimilarity score is calculated. Then, using the converged dissimilarity score, a decision is made as to whether s is

$$\begin{cases} \text{accepted if } Dscore(\mathcal{H}^{(I_{cvg})}(s, \vec{s}'; \mathcal{E}'), \vec{s}'; \mathcal{E}') < \Theta_{HS} \\ \text{rejected} \quad \text{otherwise} \end{cases}, \quad (5)$$

where θ_{HS} is a threshold for decision making.

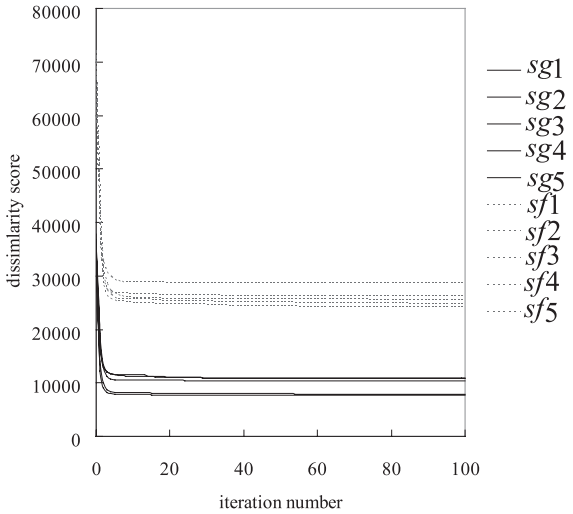

Fig. 3 Changes of signature dissimilarity scores.

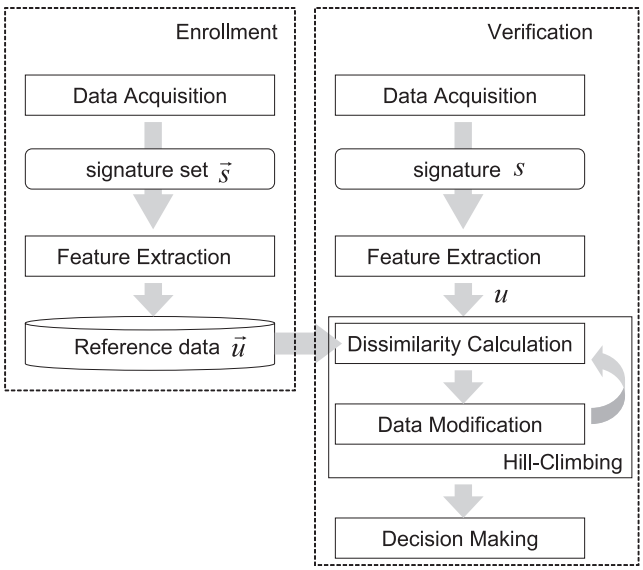

Fig. 4 Proposed online signature verification algorithm.

3. Simulation

3.1 Online Signature Verification Algorithm

3.1.1 Raw Data

Trajectories of pen position, pen pressure, and pen inclinations are acquired from a tablet. However, only pen position trajectories are considered in this paper. The raw data $s \in S$ acquired from the tablet is

$$s = (x_t, y_t), t = 1, 2, \ldots, T,$$

where T is the number of the sample points.

[†]Details of the online signature verification algorithm and the hill-climbing algorithm used in this simulation are described in Sects. 3.1 and 3.2.

3.1.2 Feature Extraction

Several features can be extracted from a signature s. In this paper, we considered three feature extractions: $\mathcal{E}_{XY}$, $\mathcal{E}_{VxVy}$, and $\mathcal{E}_{|V|\theta}$. In each feature extraction, a time-series feature vector $u = (u1_t, u2_t)$ is extracted:

($\mathcal{E}_{XY}$):

$$\begin{aligned} u1_t &= x_t \\ u2_t &= y_t, t = 1, 2, \ldots, T \end{aligned}$$

($\mathcal{E}_{VxVy}$):

$$\begin{aligned} u1_t &= vx_t \\ &= x_{t+1} - x_t \\ u2_t &= vy_t \\ &= y_{t+1} - y_t, t = 1, 2, \ldots, T-1 \end{aligned}$$

($\mathcal{E}_{|V|\theta}$):

$$\begin{aligned} u1_t &= \sqrt{vx_t^2 + vy_t^2} \\ u2_t &= \tan^{-1}\left(\frac{vy_t}{vx_t}\right), t = 1, 2, \ldots, T-1. \end{aligned}$$

3.1.3 Dissimilarity Calculation and Decision Making

The extracted feature data u is compared with reference data $\vec{u} = (u_i)_{1 \le i \le M}$, and dissimilarity scores $Dscore(s, \vec{s'}; \mathcal{E}')$ are calculated. Comparing the two signature data items is not easy because the length and shape of the signatures differ every time they are written, even if they are generated by the same user. Thus, dynamic time warping (DTW) [24] is used for the distance calculation. Figure 5 shows details of the dynamic time warping algorithm used in this paper. By using this, a minimum distance between two time-series features $D(\cdot, \cdot)$ is calculated.

The dissimilarity score $Dscore(s, \vec{s'}; \mathcal{E}')$ is calculated by

$$Dscore(s, \vec{s'}; \mathcal{E}') = \sum_{i=1}^{M} \frac{D(\mathcal{E}'(s), \mathcal{E}'(s'_i))}{T_i}, \tag{7}$$

where T_i is the duration of the data $u'_i = \mathcal{E}'(s'_i)$. A decision is made based on (2).

3.2 Hill-Climbing Algorithm

Let $Z = \{(z_t)\}_{t=1}^{T} = \{(zx_t, zy_t)\}_{t=1}^{T}$ be a two-dimensional time-series vector to be modified gradually in the hill-climbing method. The following steps are repeated until the dissimilarity score converges.

Step 1 $t = 1$

Step 2-1 Target point ($\star$ in Fig. 6) to be modified is $z_t = (zx_t, zy_t)$.

K candidate points $czx_t^{*(k)}, k = 0, 1, \ldots, K-1$ ($\bullet$ in Fig. 6) are set as follows:

$$\begin{aligned} czx_t^{*(k)} &= (zx_t^{*(k)}, zy_t) \\ zx_t^{*(k)} &= zx_t + \left(-\left[\frac{K}{2}\right] + k\right), \end{aligned}$$

where $[q]$ is the largest integer that does not exceed q. Using these candidates, K items of data $cZ_x^{*(k)}$ are produced:

$$cZ_x^{*(k)} = \{z_1, .., z_{t-1}, czx_t^{*(k)}, z_{t+1}, .., z_T\}.$$

A distance $D(cZ_x^{*(k)}, u_i)$ is calculated. Then, the best candidate czx_t^* is determined based on:

$$\begin{aligned} czx_t^* &= czx_t^{*(\hat{k})} \\ \hat{k} &= \underset{k}{\operatorname{argmin}}\, D(cZ_x^{*(k)}, u_i). \end{aligned}$$

Step 2-2 K candidates $cz_t^{*(k)}, k = 0, 1, \ldots, K-1$ ($\odot$ in Fig. 6) are set as follows:

$$\begin{aligned} cz_t^{*(k)} &= (zx_t^{*(\hat{k})}, zy_t^{*(k)}) \\ zy_t^{*(k)} &= zy_t + \left(-\left[\frac{K}{2}\right] + k\right). \end{aligned}$$

Then, K items of data $cZ^{*(k)}$ are produced:

$$cZ^{*(k)} = \{z_1, \ldots, z_{t-1}, cz_t^{*(k)}, z_{t+1}, \ldots, z_T\}.$$

Let the two time-series features to be compared be
$a = \{(a_i)\}_{i=1}^{T_I} = \{(a1_i, a2_i)\}_{i=1}^{T_I}$,
$b = \{(b_j)\}_{j=1}^{T_J} = \{(b1_j, b2_j\}_{j=1}^{T_J}$
1.Initialization
$dist(0, 0) = 0$
2.Recursion

$$dist(i, j) = = min \begin{cases} dist(i-1, j-1) + d(a_i, b_j) \\ dist(i-1, j) + d(a_i, b_j) \\ dist(i, j-1) + d(a_i, b_j) \end{cases}$$

where $d(a_i, b_j) = (|a1_i - b1_j| + 1) \times (|a2_i - b2_j| + 1)$ (6)
3.Termination
$D(a, b) = dist(T_I, T_J)$

Fig. 5 DTW algorithm.

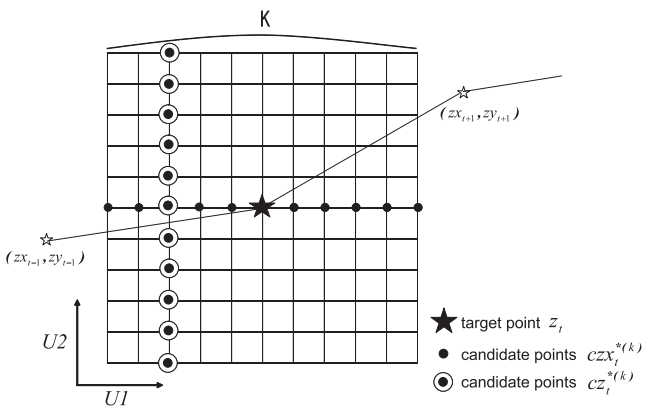

Fig. 6 A target point and candidate points.

Then, distances $D(cZ^{*(k)}, u_i)$ are calculated, and the best candidate cz_t^* is determined based on:

$$cz_t^* = cz_t^{*(\hat{\hat{k}})}$$
$$\hat{\hat{k}} = \underset{k}{\operatorname{argmin}}\, D(cZ^{*(k)}, u_i).$$

Step 3 Set $z_t \leftarrow cz_t^*$.
If $t \neq T$: $t \leftarrow t+1$ and return to Step 2-1.
If $t = T$:
If distance has converged, go to Step 4.
If distance has not converged, return to Step 1.
Step 4

$$\text{HC forgery} = \{(z_t)\}_{t=1}^{T}.$$

3.3 Experimental Settings

3.3.1 Database and Experimental Settings

The public database SVC2004 [15][†] task 2 was used for performance evaluation. Signatures associated with 40 persons in SVC are open to the public, and 20 genuine signatures for each person are available.

Genuine data Let GS_E be $\{sg_1, sg_2, \ldots, sg_5\}$, and GS_V be $\{sg_6, sg_7, \ldots, sg_{20}\}$ of each person. The first genuine signature sg_1 was used for generating reference data of a target system, four consecutive genuine signatures $(sg_i)_{2\leq i\leq 5}$ were used for adjusting the threshold, and $sg_i \in GS_V$ for each person were used for evaluation. In this setting, $M = 1$.

HC forgery The seriousness of the threat of a hill-climbing forged data attack depends on three items: the initial forgery, reference data, and a dissimilarity criterion.

Initial forgery Signatures generated by using the hill-climbing method depend on the initial signature. Thus, random forgeries were used as preliminary forgeries for HC-forgery generation. Genuine signatures sg_1 of others were used as random forgeries (39 signatures were used as an initial forgery for each person).

Reference data Let $\vec{s_a}$ be a signature set used for generating reference data in the target system. Two settings are possible:

- The reference data enrolled in the HC-attacked system are generated from the same signature set $\vec{s_a}$.
- The reference data enrolled in the HC-attacked system are generated from a different signature set $\vec{s_b}$ ($\vec{s_a} \neq \vec{s_b}$). In this experiment, $\vec{s_a} = (sg_1)$ and $\vec{s_b} = (sg_6)$

Dissimilarity criterion The hill-climbing method modifies signatures so as to improve the dissimilarity criterion. Thus, whether or not the dissimilarity criterion used for HC-forgery generation is the same as that of the target system has a major impact on the performance. Moreover, the similarity of the dissimilarity criteria also has an impact on performance. To simplify the experimental settings, we used the same equation, but used three different feature extractions to achieve different dissimilarity criteria: $\mathcal{E}_{XY}$, $\mathcal{E}_{VxVy}$, and $\mathcal{E}_{|V|\theta}$. Note that $VxVy$ and $|V|\theta$ are more similar than XY and $VxVy$ or XY and $|V|\theta$. Thus, we can consider the degree of "dissimilarity criteria's similarity" by using these three feature sets.

The following six types of HC forgeries were considered in this experiment.

Type 1 $\mathcal{H}^{(I_{cvg})}(sf_i, \vec{s_a}; \mathcal{E}_{XY})$: Reference data extracted from $\vec{s_a}$ was used for HC-forgery generation, and $\mathcal{E}_{XY}$ was used for feature extraction. Hill-climbing was performed in $X - Y$ space.

Type 2 $\mathcal{H}^{(I_{cvg})}(sf_i, \vec{s_b}; \mathcal{E}_{XY})$: Reference data extracted from $\vec{s_b}$ was used for HC-forgery generation, and $\mathcal{E}_{XY}$ was used for feature extraction. Hill-climbing was performed in $X - Y$ space.

Type 3 $\mathcal{H}^{(I_{cvg})}(sf_i, \vec{s_a}; \mathcal{E}_{VxVy})$: Reference data extracted from $\vec{s_a}$ was used for HC-forgery generation, and $\mathcal{E}_{VxVy}$ was used for feature extraction. Hill-climbing was performed in $Vx - Vy$ space.

Type 4 $\mathcal{H}^{(I_{cvg})}(sf_i, \vec{s_b}; \mathcal{E}_{VxVy})$: Reference data extracted from $\vec{s_b}$ was used for HC-forgery generation, and $\mathcal{E}_{VxVy}$ was used for feature extraction. Hill-climbing was performed in $Vx - Vy$ space.

Type 5 $\mathcal{H}^{(I_{cvg})}(sf_i, \vec{s_a}; \mathcal{E}_{|V|\theta})$: Reference data extracted from $\vec{s_a}$ was used for HC-forgery generation, and $\mathcal{E}_{|V|\theta}$ was used for feature extraction. Hill-climbing was performed in $|V| - \theta$ space.

Type 6 $\mathcal{H}^{(I_{cvg})}(sf_i, \vec{s_b}; \mathcal{E}_{|V|\theta})$: Reference data extracted from $\vec{s_b}$ was used for HC-forgery generation, and $\mathcal{E}_{|V|\theta}$ was used for feature extraction. Hill-climbing was performed in $|V| - \theta$ space.

Samples of initial forgeries and HC forgeries are shown in Figs. 7–12, together with target reference data items.

The parameter L for the convergence test was set to $L = 5$, and the number of candidate points was set to $K = 9$ in the experiments.

3.4 Menace Analysis for Systems with the Same Dissimilarity Criterion

First, we evaluated the performance of target systems that use the same dissimilarity criterion as the HC-forgery generation. This menace is equivalent to that for an online hill-climbing attack.

We used error trade-off curves and false accept rates (FARs) at the thresholds Θ to achieve false reject rates (FRRs) of 5% and 10% as criteria for measuring the menace. If HC forgeries are cleverly generated, dissimilarity scores become smaller. Then, FARs of HC forgeries become worse than those of random forgeries, leading to performance degradation.

[†]The database can be download directly from http://www.cse.ust.hk/svc2004/download.html

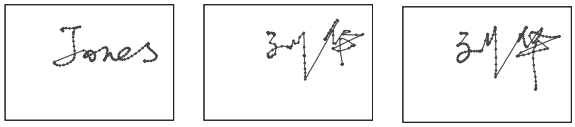

Fig. 7 HC forgery sample (Type 1) in $X - Y$ space.

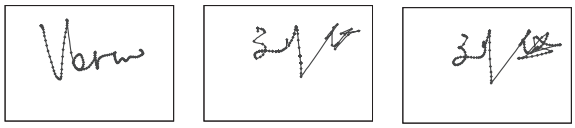

Fig. 8 HC forgery sample (Type 2) in $X - Y$ space.

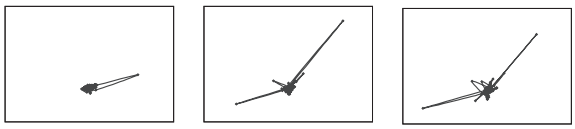

Fig. 9 HC forgery sample (Type 3) in $Vx - Vy$ space.

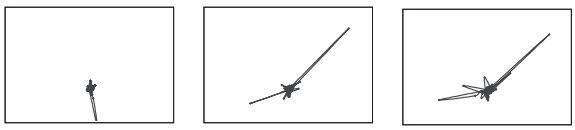

Fig. 10 HC forgery sample (Type 4) in $Vx - Vy$ space.

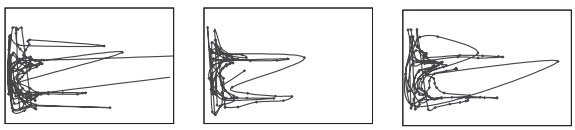

Fig. 11 HC forgery sample (Type 5) in $|V| - \theta$ space.

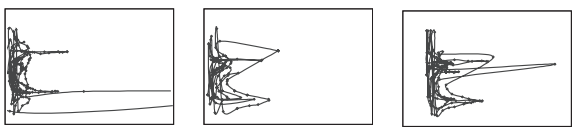

Fig. 12 HC forgery sample (Type 6) in $|V| - \theta$ space.

The FAR of the HC forgery calculated in this experiment is defined by

$$FAR_{\Theta}^{B,H,S} = \frac{1}{\#FS} \times \sum_{sf \in FS} Pr[Dscore(\mathcal{H}^{(I_{cvg})}(sf, \vec{s}; \mathcal{E}'), \vec{s'}; \mathcal{E}') < \Theta], \quad (8)$$

and the FAR of the random forgery is defined by

$$FAR_{\Theta}^{B,R} = \frac{1}{\#FS} \sum_{sf \in FS} Pr[Dscore(sf, \vec{s'}; \mathcal{E}') < \Theta] \quad (9)$$

where $\#FS$ is the number of random forgeries; in this experiment $\#FS = 39 \times 40$. The false reject rate (FRR) is calculated by

$$FRR_{\Theta} = \frac{1}{\#GS_V} \sum_{s \in GS_V} Pr[Dscore(s, \vec{s}; \mathcal{E}) \geq \Theta]. \quad (10)$$

Figures 13–18 show error trade-off curves, and FARs are summarized in Table 1.

Figures 13–18 show that the performance of all three systems was degraded drastically. In the worst case, the FAR@FRR = 10% was degraded from 0.1% to 98.4% (in Fig. 17). The performance of Types 1 and 2 (or Types 3 and 4, or Types 5 and 6) have little difference; therefore, the difference of the reference data items had a minor impact on this evaluation.

3.5 Menace Analysis for System with a Different Dissimilarity Criterion

Second, we evaluated the performance of a target system that used a different dissimilarity criterion from the HC-forgery generation. In this experiment, the target system with a dissimilarity criterion based on $\mathcal{E}_{VxVy}$ was used as a target system. Let $\mathcal{E} \in \{\mathcal{E}_{XY}, \mathcal{E}_{VxVy}, \mathcal{E}_{|V|\theta}\}$ and $\vec{s}, \vec{s'} \in \{\vec{s_a}, \vec{s_b}\}$ be a feature extraction scheme and signature data items used for HC-forgery generation, respectively. The FAR in this experiment is defined by

$$FAR_{\Theta}^{B,H,R} = \frac{1}{\#FS} \times \sum_{sf \in FS} Pr[Dscore(\mathcal{H}^{(I'_{cvg})}(sf, \vec{s}; \mathcal{E}), \vec{s'}; \mathcal{E}') < \Theta], \quad (11)$$

and the FRR is the same as (10).

Figure 19 shows the error trade-off curves of each type of HC forgery, together with a curve of random forgeries. The performance against Type 3 was by far the worst, because HC forgeries of this type were generated using the same system as the target system ($\vec{s} = \vec{s'} = \vec{s_a}$ and $\mathcal{E} = \mathcal{E}' = \mathcal{E}_{VxVy}$).

The performance against Types 4 and 5 was better than that against Type 3. However, the performance was degraded relative to that against random forgeries. Type 4 was generated using the system with the same feature extraction ($\mathcal{E} = \mathcal{E}'$, $\vec{s} \neq \vec{s'}$), and Type 5 was generated using the system whose reference data were extracted from the same data ($\vec{s} = \vec{s'} = \vec{s_a}, \mathcal{E} \neq \mathcal{E}'$).

The performance against other types of HC forgeries was much better than that for Types 3, 4, and 5, though the performance was slightly worse than that against random forgeries.

From these observations, a hill-climbing method can improve the pre-determined score, but it cannot restore the original data perfectly. However, the performance against Types 3, 4, and 5 was much worse than that against random forgeries. Therefore, a suitable countermeasure is necessary.

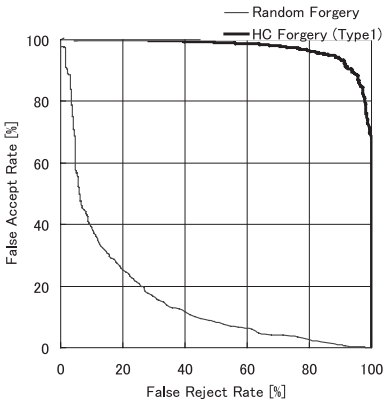

Fig. 13 Error trade-off curve (Type 1).

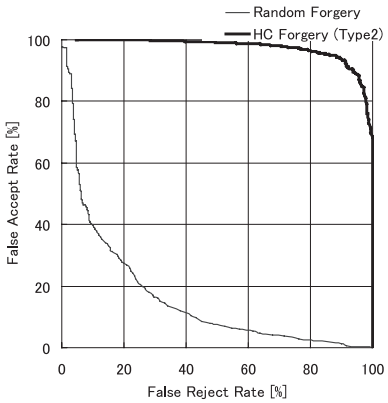

Fig. 14 Error trade-off curve (Type 2).

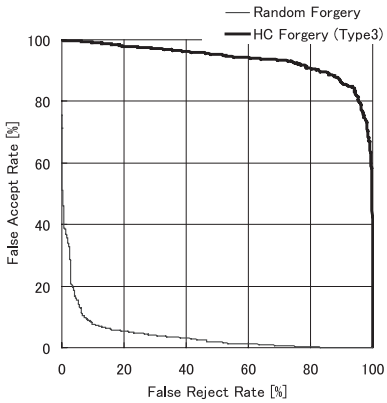

Fig. 15 Error trade-off curve (Type 3).

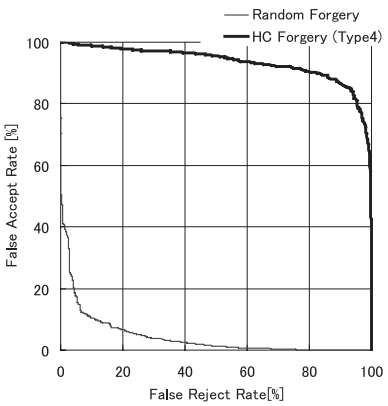

Fig. 16 Error trade-off curve (Type 4).

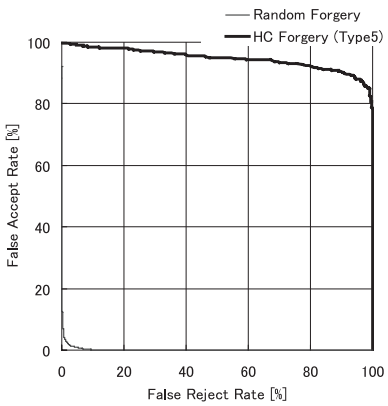

Fig. 17 Error trade-off curve (Type 5).

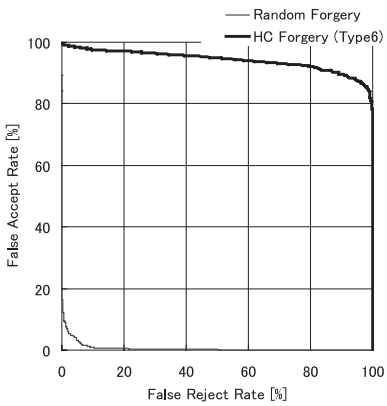

Fig. 18 Error trade-off curve (Type 6).

Table 1 FARs of a target system ($\mathcal{E} = \mathcal{E}'$) [%].

	$\Theta_{@FRR=5\%}$		$\Theta_{@FRR=10\%}$	
setting	HC forgery	Random	HC forgery	Random
Type 1	100	56.9	99.9	39.0
Type 2	100	57.8	99.9	39.6
Type 3	99.6	15.4	99.2	7.6
Type 4	99.3	17.5	98.8	10.3
Type 5	99.0	0.9	98.4	0.1
Type 6	98.5	3.1	97.6	1.0

3.6 Robustness of Proposed System

We evaluated the proposed online signature verification algorithm using the HC forgeries and genuine signatures. In this experiment, the system with a dissimilarity criterion based on $\mathcal{E}_{V_X V_Y}$ was used as a target system.

The FAR and FRR of the proposed system are defined by

$$FAR_{\Theta}^{P,H,R} = \frac{1}{\#FS} \times \sum_{sf \in FS} Pr[Dscore(\mathcal{H}^{(I_{cvg})}(sf_{\mathcal{E}}^{(I'_{cvg})}(\vec{s}), \vec{s}'; \mathcal{E}'), \vec{s}'; \mathcal{E}') < \Theta], \tag{12}$$

and the false reject rate (FRR) is defined by

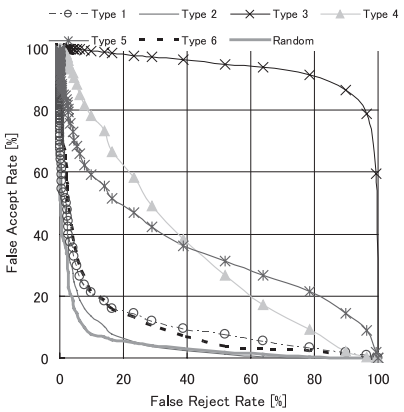

Fig. 19 Menace of HC-forged data attack.

$$FRR_{\Theta}^{P} = \frac{1}{\#GS_V} \times \sum_{s \in GS_V} Pr[Dscore(\mathcal{H}^{(I'_{cvg})}(sg, \vec{s}'; \mathcal{E}'), \vec{s}'; \mathcal{E}') \geq \Theta].$$

The EERs are summarized in Table 2. Error trade-off curves are shown in Figs. 20–25. In the table and the figures, the proposed algorithm is indicated by "With HC". For comparison, experimental results of an algorithm without hill-climbing were also evaluated as a baseline system. This

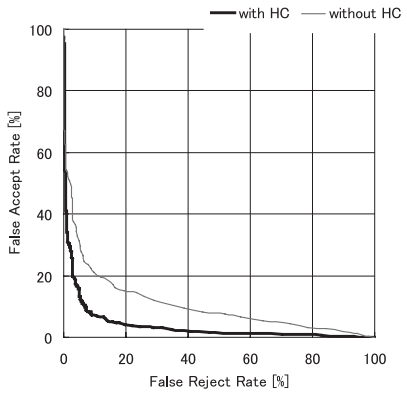

Fig. 20 Error trade-off curve (Type 1).

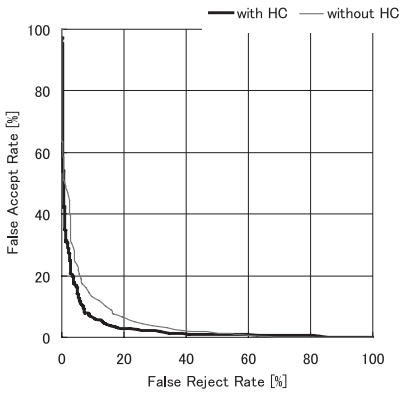

Fig. 21 Error trade-off curve (Type 2).

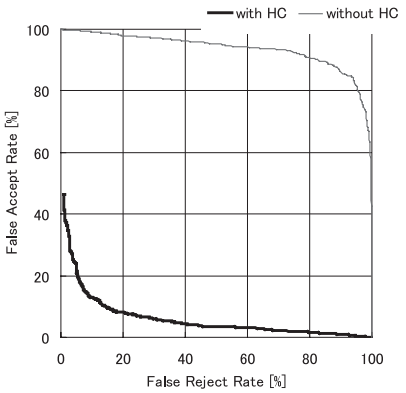

Fig. 22 Error trade-off curve (Type 3).

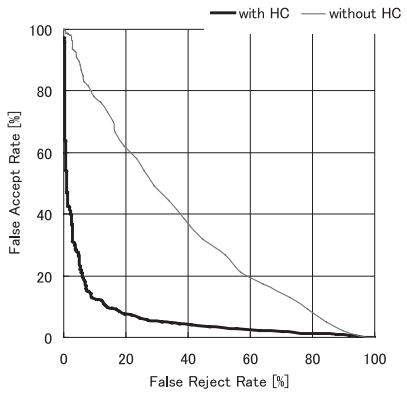

Fig. 23 Error trade-off curve (Type 4).

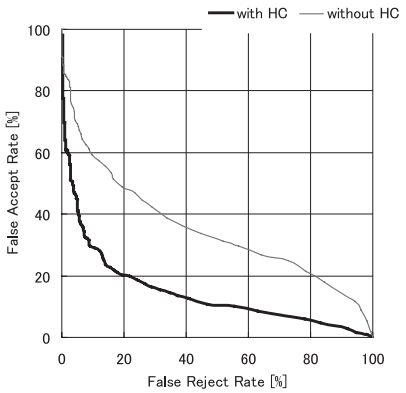

Fig. 24 Error trade-off curve (Type 5).

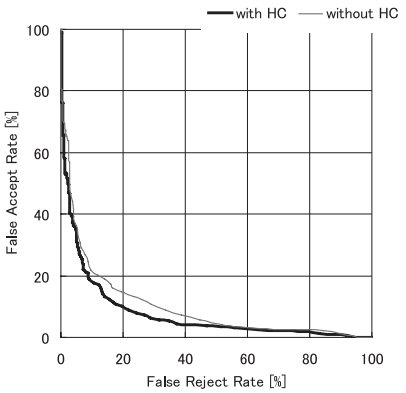

Fig. 25 Error trade-off curve (Type 6).

Table 2 EERs of proposed system [%].

type of HC forgery	proposed (with HC)	baseline (without HC)
Type 1	8.2	16.3
Type 2	7.9	12.0
Type 3	12.1	88.0
Type 4	11.9	38.6
Type 5	20.3	37.1
Type 6	13.8	16.3

baseline system is indicated by "Without HC" in the table and figures. The FAR and FRR of the system without HC were calculated using (11) and (10).

The experimental results of "Without HC" show the performance of the online signature verification algorithm without any countermeasures for hill-climbing forged data attacks. The EERs of the algorithm were 88.0% for Type 3, 38.6% for Type 4, and 37.1% for Type 5. These results show that the hill-climbing forged data attack is a menace for basic online signature verification algorithms.

On the other hand, the EERs of the proposed algorithm were 12.1% for Type 3, 11.9% for Type 4, and 20.3% for Type 5, and the EERs of other types of data were also improved. Therefore, a robust online signature verification algorithm for hill-climbing forged data attacks can be generated by incorporating a hill-climbing method into the basic online signature verification algorithm.

The reason why the proposed algorithm can improve the performance against a hill-climbing forged data attack is that converged dissimilarity scores are used for verification. Though hill-climbing methods modify the input data and can improve the score, it is extremely difficult to find a global optimum in high dimensional problems. In many cases, hill-climbing methods find a local optimum, and the value of the local optimum depends on the initial setting (initial input). This dependence on the initial input is important in the proposed algorithm. As shown in Fig. 3, converged dissimilarity scores of genuine signatures are smaller than those of forgeries. It can be considered that genuine signatures are better initial inputs than forgeries and can produce better local optima. Furthermore, converged dissimilarity scores of genuine signatures become smaller than those of many HC forgeries because HC forgeries are generated using the hill-climbing method, and because the initial inputs for HC-forgery generation are forgeries. Thus, converged dissimilarity scores of HC forgeries become the same as or similar to those of the initially input forgeries because of the dependence on initial inputs. We utilized this characteristic of hill-climbing methods, namely, that local optima (converged scores) depend on the initial inputs, in this proposed algorithm.

4. Conclusions

In this paper, we focus on an offline hill-climbing attack against online signature verification algorithms. To generate a countermeasure against offline hill-climbing attacks for online signature verification, first we generated several types of HC forgery using the public database SVC2004 and analyzed their degree of menace. Then, based on the analysis, we proposed a robust online signature verification algorithm that uses a dissimilarity score converged by hill-climbing for verification. Several experiments were performed for evaluating the proposed algorithm, and the experimental results show that the proposed algorithm is promising.

Though, the proposed algorithm is a promising solution for offline hill-climbing attacks, repeated calculation of the dissimilarity score is necessary, which increases the processing time. Thus, a fast hill-climbing algorithm for verification should be developed.

Moreover, the performance of the proposed system against non hill-climbing forgeries may degrade because all input signatures are modified so as to make the dissimilarity scores small. To reduce this possible degradation, we need to select the signatures to be modified. A possible approach for this solution is to modify only signatures that are accepted as being genuine signatures.

In this paper, we only applied the proposed scheme to a basic online signature verification algorithm. This scheme can be applied to more sophisticated online signature verification algorithms such as [25], [26]. Thus, our future projects will include evaluation of the scheme in such algorithms.

References

[1] A.K. Jain, A. Ross, and U. Uludag, "Biometrics: A tool for information security," IEEE Trans. Information Forensics and Security, vol.1, no.2, pp.125–143, June 2006.

[2] A. Andler, "Biometric system security," in Handbook of Biometrics, ed. A.K. Jain, P. Flynn, and A.A. Ross, ch. 19, Springer Science+Business Media, LLC, 2008.

[3] C. Soutar, "Biometric system security," http://www.bioscrypt.com/assets/security_soutar.pdf

[4] A. Adler, "Sample images can be independently restored from face recognition template," Proc. 2003 Canadian Conference on Electrical and Computer Engineering, vol.2, pp.1163–1166, 2003.

[5] A. Adler, "Images can be regenerated from quantized biometric match score data," Proc. 2004 Canadian Conference on Electrical and Computer Engineering, pp.469–472, 2004.

[6] A. Adler, "Vulnerabilities in biometric encryption systems," AVBPA, ed. T. Kanade, A. Jain, and N.K. Ratha, Lect. Notes Comput. Sci., vol.3546, pp.1100–1109, Springer, 2005.

[7] J. Galbally, J. Fierrez, J. Ortega-Garcia, C. McCool, and S. Marcel, "Hill-climbing attack to an eigenface-based face verification system," Proc. 2009 First International Conference on Biometrics, Identity and Security, 2009.

[8] U. Uludag and A. Jain, "Attacks on biometric systems: A case study in fingerprints," Proc. SPIE-EI 2004, Security, Steganography and Watermarking of Multimedia Contents, pp.622–633, Jan. 2004.

[9] M. Martinez-Diaz, J. Fierrez-Aguilar, F. Alonso-Fernandez, J. Ortega-Garcia, and J.A. Siguenza, "Hill-climbing and brute-force attacks on biometric systems: A case study in match-on-card fingerprint verification," Proc. IEEE International Carnahan Conferences Security Technology, pp.151–159, 2006.

[10] Y. Yamazaki, A. Nakashima, K. Tasaki, and N. Komatsu, "A study on vulnerability in on-line writer verification system," Proc. Eighth International Conference on Document Analysis and Recognition, pp.640–644, 2005.

[11] J. Galbally, J. Fierrez, and J. Ortega-Garcia, "Bayesian hill-climbing attack and its application to signature verification," Proc. International Conference on Biometrics, LNCS, vol.4642, pp.386–395, 2007.

[12] H. Yasunaga, A. Nakajima, Y. Yamazaki, and N. Komatsu, "A study on vulnerabilities and countermeasures in writer recognition systems," 2007 Symposium on Cryptography and Information Security, 2007.

[13] D. Muramatsu, Y. Kato, and T. Matsumoto, "Online signature verification using Monte Carlo methods," J. Human Interface Society, vol.9, no.2, pp.191–200, 2007.

[14] G. Amayeh, G. Bebis, and M. Nicolescu, "Improving hand-based verification through online finger template update based on fused confidences," Proc. IEEE Third International Conference on Biometrics: Theory, Application and Systems, 2009.

[15] D.Y. Yeung, H. Chang, Y. Xiong, S. George, R. Kashi, T. Matsumoto, and G. Rigoll, "SVC2004: First international signature verification competition," Proc. International Conference on Biometric Authentication, LNCS, pp.16–22, 2004.

[16] R. Plamondon and G. Lorette, "Automatic signature verification and writer identification - The state of the art," Pattern Recognit., vol.22, no.2, pp.107–131, 1989.

[17] F. Leclerc and R. Plamondon, "Automatic signature verification: The state of the art - 1989-1993," International Journal of Pattern Recognition and Artificial Intelligence, vol.8, no.3, pp.643–660, 1994.

[18] R. Plamondon and S.N. Srihari, "On-line and off-line handwriting recognition: A comprehensive survey," IEEE Trans. Pattern Anal. Mach. Intell., vol.22, no.1, pp.63–84, 2000.

[19] D. Impedovo and G. Pirlo, "Automatic signature verification: The state of the art," IEEE Trans. Syst., Man Cybern. C, Appl. Rev., vol.38, no.5, pp.609–635, Sept. 2008.

[20] J. Fierrez and J. Ortega-Garcia, "On-line signature verification," in Handbook of Biometrics, ed. A.K. Jain, P. Flynn, and A.A. Ross, ch. 10, Springer Science+Business Media, LLC, 2008.

[21] "Biosecure multimodal evaluation campaign 2007 (BMEC 2007)," http://biometrics.it-sudparis.eu/BMEC2007/, 2007.

[22] B. Dorizzi, R. Cappelli, M. Ferrara, D. Maio, D. Maltoni, N. Houmani, S. Garcia-Salicetti, and A. Mayoue, "Fingerprint and on-line signature verification competitions at ICB2009," ICB, ed. M. Tistarelli and M.S. Nixon, Lect. Notes Comput. Sci., vol.5558, pp.725–732, Springer, 2009.

[23] V. Blankers, C. van den Heuvel, K. Franke, and L. Vuurpiji, "The ICDAR 2009 signature verification competition," Proc. 2009 10th International Conference on Document Analysis and Recognition, pp.1403–1407, 2009.

[24] L. Rabiner and B.H. Juang, Fundamentals of speech recognition, Prentice Hall, 1993.

[25] J. Pascual-Gaspar, V.C. noso Payo, and C. Vivarancho-Pascual, "Practical on-line signature verification," ICB, ed. M. Tistarelli and M.S. Nixon, Lect. Notes Comput. Sci., vol.5558, pp.1180–1189, Springer, 2009.

[26] D. Muramatsu and T. Matsumoto, "Online signature verification algorithm with a user-specific global-parameter fusion model," Proc. 2009 IEEE International Conference on Systems, Man, and Cybernetics, pp.486–491, 2009.

Daigo Muramatsu received B.S., M.E., and Ph.D. degrees in Electrical, Electronics, and Computer Engineering from Waseda University, Tokyo, in 1997, 1999, and 2006, respectively. He is currently an assistant professor of Seikei University.