An Evolutionary Keystroke Authentication Based on Ellipsoidal Hypothesis Space

Jae-Wook Lee
School of Computer Science and Engineering
Seoul National University
Sillim-dong, Gwanak-gu,
Seoul, 151-744 Korea
jwlee@soar.snu.ac.kr

Sung-Soon Choi
School of Computer Science and Engineering
Seoul National University
Sillim-dong, Gwanak-gu,
Seoul, 151-744 Korea
sschoi@soar.snu.ac.kr

Byung-Ro Moon
School of Computer Science and Engineering
Seoul National University
Sillim-dong, Gwanak-gu,
Seoul, 151-744 Korea
moon@soar.snu.ac.kr

ABSTRACT

Keystroke authentication is a biometric method utilizing the typing characteristics of users. In this paper, we propose an evolutionary method for stable keystroke authentication. In the method, typing characteristics of users are represented by n-dimensional vectors and an ellipsoidal hypothesis space, which distinguishes a collection of the timing vectors of a user from those of the others, is evolved by a genetic algorithm. A filtering scheme and an adaptation mechanism are also presented to improve the stability and effectiveness of the proposed method. Empirical results show that the error rates of our method for authentication are reasonably small.

Categories and Subject Descriptors

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection (D.4.6, K.4.2)—*authentication*; G.1.6 [**Numerical Analysis**]: Optimization—*constrained optimization*

General Terms

Experimentation, Security

Keywords

Biometric authentication, identity verification, keystroke biometrics, keystroke dynamics

1. INTRODUCTION

The security of computer system is based on the successful verification whether a person is a legitimate user, who is permitted to get access to the system, or not. Most current access systems demand the pair of account-name and password to authenticate users. The performance of a password-based authentication relies on the secrecy of passwords. If the secrecy is broken, an imposter may be identified with a legitimate user. To prevent from an intrusion into the system, it is very important for users to maintain the secrecy of their passwords. Passwords consisting of common words, phrases, or terms are generally considered weak because a potential intruder can easily guess and find them via dictionary attacks. Nevertheless, some studies have shown that users tend to choose passwords that can be broken by an exhaustive search of a relatively small subset of all possible passwords. According to a case study for 13,797 Unix passwords, nearly 25% of the passwords were cracked by a small *dictionary* containing only 62,727 words [17]. This report is remarkable considering the fact that there are approximately 2×10^{14} different choices of eight-character password.

As the traditional password-based access system proved vulnerable, the requests for some other security devices to resolve this problem increased considerably, especially biometric devices that use physiological or behavioral human characteristics such as fingerprint, vein patterns, and iris analysis. However, these kinds of security devices need additional equipments to perceive human characteristics and the users have to make an effort to provide the system with their characteristics voluntarily. Besides, the implementation cost is too high.

Keystroke biometrics[1] is a behavioral biometrics based on the hypothesis that a user has a consistent pattern in keystroke rhythms and the patterns are different between users. The authentication system extracts the typing characteristics of a particular user from his typing log, analyzes them, and then identifies a user. Keystroke biometrics-based authentication is appealing for many reasons. First, it is relatively inexpensive to implement, since the system needs no additional device. The only hardware required are the computer as usual. Second, it does not bother a user, since the keystroke measurement does not require any additional information except for keystroke. Finally, maintaining and updating the system are very convenient, since the system administrator has only to control softwares.

Several problems, however, disturb the practical use of this method. One of the major problems is the instability of typing patterns due to temporal variation. Unlike

[1]Keystroke biometrics is also known as *keystroke dynamics*. Throughout the paper, we use the term "keystroke biometrics" instead of the more popular term "keystroke dynamics", since the term "keystroke dynamics" is misleading as Gunetti and Picardi [13] pointed out.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.

other biometric features, keystroke characteristics can vary because of the physical state of a user, the environmental conditions including the types of keyboards, and normal statistical variation. Due to these variations, keystroke-based authentication shows much higher error rates than others.

In this paper, we propose an evolutionary method for stable keystroke-based authentication. In the method, the typing characteristics of users are represented by n-dimensional vectors (called *timing vectors*) and an ellipsoidal hypothesis space, which distinguishes a collection of the timing vectors of a user from the others, is evolved by a genetic algorithm [15, 12]. A filtering scheme and an adaptation mechanism are also presented to improve the stability and effectiveness of the proposed method. Empirical results for a number of data sets show that the error rates of our method for authentication are reasonably small.

The rest of the paper is organized as follows. In the next section, we briefly review various approaches in keystroke biometrics briefly and analyze their weak points. In Section 3, the main ideas of the proposed approaches are described. We give a full detail of the implementations of the approaches in Section 4 and provide experimental results in Section 5. Finally, Section 6 concludes the paper with suggestions for future work.

2. PREVIOUS WORK

The effectiveness of typing characteristics as personal identifiers was already known in the 19$^{\text{th}}$ century. For example, telegraph operators could recognize each other based on their typing rhythms. An approach to using keystroke patterns for identification was first suggested in 1975 [24], but most applicative papers have published since 1985 [25, 18, 11, 26, 16]. Due to the instability of typing rhythms, the incipient studies utilized a long predefined text in making a template of the user [25, 18]. Though it is tedious that the system requires a user to type too much, a long predefined text is used for diverse purposes [10, 2]. Most research papers utilize relatively short strings such as first name, last name or password, but some research use a free text for extracting the typing characteristics of a particular user [21, 9, 13].

Latencies between keystrokes and durations of keystrokes are popular measurements of typing characteristics because they can be easily measured without any additional equipment. There exist other measurements for identifying a user. Young and Hammon [26] utilized *keystroke pressure* in addition to the time periods between keystrokes. Due to an expensive additional equipment for measuring typing pressure, there were few studies utilizing keystroke pressure. De Ru *et al.*[8] analyzed a typing feature with the physical distance between the keys in the keyboard called *typing difficulty*, as well as the *time interval* between successive characters. Recently, Cho and Hwang [5] presented *artificial rhythms* and *timing cues* for the purpose of reducing the instability of typing patterns. Users type their names and passwords according to artificially designed rhythms, and timing cues including auditory, visual and audiovisual cues help users type consistently.

The methods for classifying inputs into a legitimate user's or an imposter's have been developed in a variety of ways. *Statistical methods* using averages and standard deviations are most popular [25, 18, 16, 10, 9, 14]. Some researchers applied *fuzzy logic* to the measured values of keystroke timings [8, 1], and there are also *neural network*-based methods [3, 19, 4]. De Ru and Eloff [8] categorized time intervals between two successive keys into four classes, and classified users based on the categorized intervals. Yu and Cho [27] proposed a *genetic algorithm* combined with a Support Vector Machine (GA-SVM). SVM is an excellent novelty detector with fast speed, but it performs poor when the feature set is too large. Genetic algorithm was employed to implement a randomized search for a feature subset selection process.

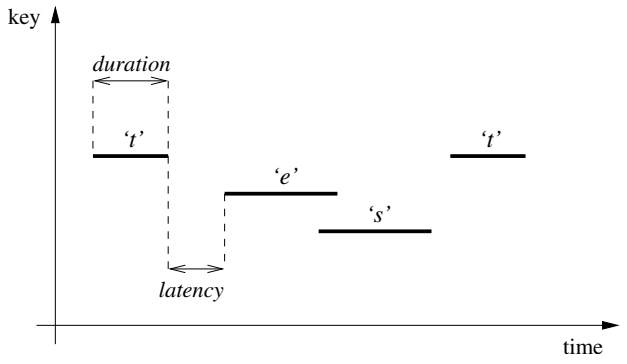


Figure 1: Timing vector corresponding to "test"

Although some studies showed remarkable performances, *i.e.* low error rates, most of them are impractical. Obaidat and Sadoun [22] reported no error in user verification, but they used the imposter's patterns as well as the legitimate user's in training. The imposter's patterns are not available in the training phase unless a password is exposed. Lin [19] reported 1.1% False Reject Rate (FRR)[2] and 0% False Accept Rate (FAR)[3], but the experiments were conducted with only 151 test samples including 60 imposter samples. Moreover, there are a few studies that a huge number of training data were used [22, 4, 2, 27].

3. THE PROPOSED SYSTEM

3.1 Hypothesis Space Modeling

Typing characteristics of a user is represented by a timing vector. The notion of timing vector was introduced by Garcia [11].[4] Figure 1 illustrates a timing vector representing an example instance for the string "test". The horizontal and vertical axes indicate the time flow and the key pressed, respectively. Thick lines represent the duration of key and blanks between thick lines represent the latency between consecutive keys. Durations and latencies appear in the timing vector alternatively. The dimension of the vector space is $n = 2\ell - 1$ where ℓ is the length of the string. For example, if the string "test" of Figure 1, the timing vector is $(120, 30, 160, -20, 150, 10, 80)$. Latencies can be negative if the key is pressed before the preceding key is released.

It is generally observed that the typing instances of a user locate closely together in the vector space. Figure 2

[2]The rate of typing instances in which a legitimate user is denied. Also known as False Alarm Rate (FAR).
[3]The rate of typing instances in which an imposter is allowed. Also known as Imposter Pass Rate (IPR).
[4]Timing vector in this paper is slightly different from Garcia's.

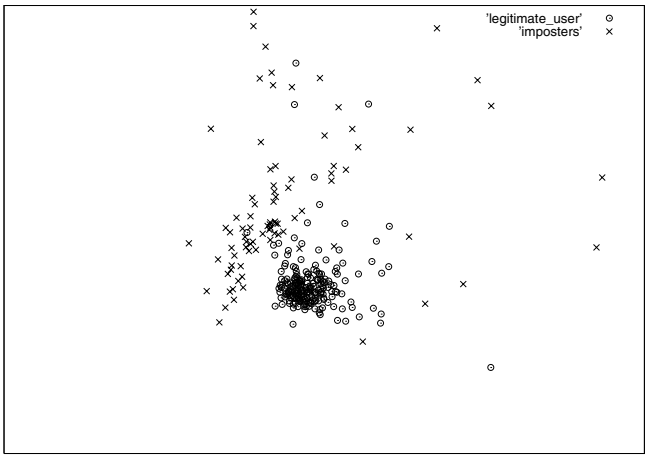


Figure 2: Visualization of timing vector examples

shows examples of typing instances for a number of users. In the figure, the timing vectors are transformed into the two-dimensional vectors by *Sammon's mapping* [23] for visualization of vector space. As seen in Figure 2, the typing instances of a user appear clustered, and some outliers are observed. Thus, it is a reasonable choice to consider the n-dimensional convex body of relatively small volume including legitimate typing instances except the outliers as the hypothesis space. Convex hull may be a candidate for the hypothesis space, however there are some difficulties in computational aspects. First of all, the computational complexity of computing the convex hull of m points in an n-dimensional space is known as $\Omega(m^{\lfloor n/2 \rfloor})$ [6], which is exponential in the dimension of the vector space. Moreover, the expression of convex hull is too intricate to manipulate the hypothesis space. Alternatively, we propose a hypothesis space of a simple formulation based on the p-norm on $\mathbb{R}^n$, so that a distance from an arbitrary vector to the hypothesis space can be easily computed.

Suppose that p is a real number with $p \geq 1$. The p-norm of a vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ is defined as the following:

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}}. \tag{1}$$

The value of $\|\mathbf{x}\|_p$ represents the distance from the origin to the point $(x_1, x_2, \ldots, x_n)$ with respect to the distance metric induced by the p-norm. The set of the vectors whose norm is at most one, unit circle, has a different shape according to the p value. In the two-dimensional vector space, the unit circle for the 1-norm (taxicab norm) is a rhomboid, a circle for the 2-norm (Euclidean norm), and a square for the infinity norm.

The suggested hypothesis space is constructed based on the *extended* p-norm defined as

$$\|\mathbf{x}\|_{\mathbf{r},p} = \left(\sum_{i=1}^{n} \left| \frac{x_i}{r_i} \right|^p \right)^{\frac{1}{p}}. \tag{2}$$

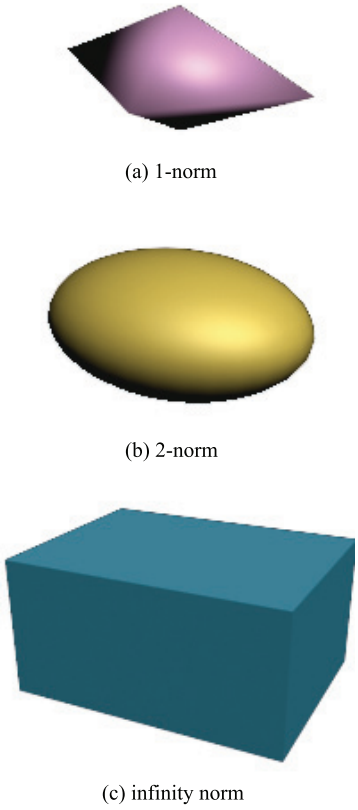
The extended p-norm is a generalization of the p-norm described in equation (1). It is obtained by adding the scaling parameters $\mathbf{r} = (r_1, r_2, \ldots, r_n)$ to the p-norm.[5] The scaling parameters enable the hypothesis space to have different radius for each axis according to the distribution of vector elements. A major assumption is that there exists the ideal timing vector which describes the characteristics of a particular user and the measured vectors of the user may differ from the ideal vector slightly. The difference between the ideal vector and the measured vector is defined as the p-norm distance between two vectors. The hypothesis space H based on the extended p-norm is defined as follows:

$$H = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x} - \mathbf{c}\|_{\mathbf{r},p} \leq 1\}. \tag{3}$$

The above is a convex body in n-dimensional vector space centered at $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ with radii $\mathbf{r} = (r_1, r_2, \ldots, r_n)$, *i.e.* the lengths of the n semi-axes. The center point $\mathbf{c}$ and the radii $\mathbf{r}$ represent the ideal vector and the scaling parameters respectively. Figure 3 illustrates the shape of hypothesis spaces on the three-dimensional vector space. It is an octahedron for $p = 1$, an ellipsoid for $p = 2$, and a cube for $p = \infty$. As you can see, the hypothesis space for $p = \infty$ is the largest and the hypothesis space for $p = 1$ is the smallest among the three.

We define the *pseudo-volume* V of a hypothesis space H as follows:

$$V(H) = \prod_{i=1}^{n} r_i. \tag{4}$$



(a) 1-norm

(b) 2-norm

(c) infinity norm

Figure 3: The shapes of the hypothesis spaces on the three-dimensional vector space

[5]The extended p-norm is also a norm.

The objective of the problem is to find a hypothesis space of the minimum pseudo-volume containing all the measured vectors of a legitimate user. More formally, letting L be the set of the timing vectors of a legitimate user and $\mathbf{c}_H$ and $\mathbf{r}_H$ be the center and the radii of H, respectively, the formulation of the problem is

$$\min V(H) \text{ subject to } \|\mathbf{x} - \mathbf{c}_H\|_{\mathbf{r}_H,p} \leq 1 \text{ for all } \mathbf{x} \in L. \quad (5)$$

The pseudo-volume defined above is not an actual volume[6], but it is enough to explain the bulk of a solid body. There are $2n$ parameters related to $\mathbf{r}$ and $\mathbf{c}$ in a hypothesis space, and we use a genetic algorithm to find optimal parameter values and consequently to find the hypothesis space of the minimum volume.

3.2 Eliminating Outliers

The performance of the authentication system strongly depends on how good the user templates are, and thus making a good template is a very important problem. Nevertheless, this is not easy due to the instability of inputs. Inputs used in making a hypothesis space usually contain some amount of noise because the physical state of a user or the environmental condition may disturb a user in typing in the enrollment process. Cho and Hwang [5] introduced artificial rhythms and timing cues to improve the quality of timing vectors used in making a hypothesis space. In their study, uniqueness, consistency, and discriminability are improved by using artificial rhythms and timing cues.

In this paper, we simply eliminate outliers according to the average and standard deviation of each elements in vectors. The following describes the eliminating scheme.

$$|x_i - m_i| \leq \alpha \cdot \sigma_i. \quad (6)$$

In inequality (6), m_i and σ_i are the average and the standard deviation of the i^{th} element x_i $(i = 1, \ldots, n)$ of $\mathbf{x} \in L$, where L is the set of the measured timing vectors in the enrollment process. The constant α adjusts the degree of elimination. The elements which do not satisfy inequality (6), outliers, are not considered in making a hypothesis space. By replacing the value of x_i with c_i, we obtain the effect of ignoring outliers.

3.3 Adaptation Mechanism

As the user gets more accustomed to typing a particular string (especially a password string in this case), the typing patterns of the user may change gradually. It is important to reflect the variation of user's characteristics for a behavioral biometric method. This kind of instability can be reduced by applying an adaptation mechanism. An adaptation mechanism allows the authentication system to update a hypothesis space of the user according to the variation of user's characteristics instead of making a hypothesis space once in the enrollment process. Some studies reported a potential that an adaptation mechanism could be added to the established system [20, 1, 14, 7]. Monrose *et al.* [20] first (to the best of our knowledge) applied an adaptation mechanism.

We find the hypothesis space containing all $\mathbf{x} \in L$ in Section 3.1 if L is a set of timing vectors used for the enrollment. We can obtain an adaptation effect by updating L. If an input vector $\mathbf{x}$ is classified as a legitimate user's, $\mathbf{x}$ replaces the oldest element in L.

3.4 Classification

A legitimate user's vector which does not participate in making a hypothesis space may appear out of the hypothesis space slightly, because the hypothesis space is constructed tightly according to the vectors in the enrollment process. Therefore, we append some margin to the hypothesis space as follows:

$$\|\mathbf{x} - \mathbf{c}\|_{\mathbf{r},p} \leq 1 + \gamma \cdot \sigma_P, \quad (7)$$

where σ_P denotes the standard deviation of the value of $\|\mathbf{x} - \mathbf{c}\|_{\mathbf{r},p}$ for $\mathbf{x} \in L$, and γ is a constant for adjusting the degree of security level. A timing vector is classified as the vector of the legitimate user if it satisfies inequality (7). Otherwise, it is classified as the vector of an imposter. The smaller the value of γ is, the higher the security level of the system is.

4. IMPLEMENTATION

4.1 Overall System

There are three processes, the enrollment process, the authentication process, and the adaptation process in the system. Figure 4 illustrates the overall structure of the authentication system. Solid lines indicate the process flow, while dashed lines indicate the data flow. If a new user wishes to be enrolled in the system, the system requests a user to type a password string repeatedly. When the requested number of timing vectors are measured, the system eliminates outliers in the measured vectors before making a template[7] of the user. Then it creates a proper template which minimizes the target function in equation (4). Created hypotheses are stored in the database for the authentication process. This corresponds to the enrollment process, and it is shown at the top of Figure 4.

In the authentication process, the system determines whether the user who attempted to log in is legitimate with his timing vector. The extended p-norm distance is computed for the measured vector, and the system classifies an input into a legitimate user or an imposter according to inequality (7). The measured vector is saved in the database for the adaptation process if a user is identified as a legitimate user. The users identified as legitimate are allowed to access the system, and the others are rejected.

For the adaptation, the system extracts the timing vectors of a legitimate user and imposters from the database. Then it creates the updated templates as described in Section 3.3. The adaptation process may be executed every time the log-in attempts are made unless it accompanies a heavy load. It is also possible to execute the adaptation process whenever the system administrators want.

4.2 Web-based System

Most authentication systems are based on the network, especially the Internet, in these days. A serious problem is occurred in measuring a timing vector on the network because of the network delay. Even if a server is connected to a

[6]The actual volume on the three-dimensional space can be easily computed by multiplying the pseudo-volume by a constant, $\frac{4}{3}$ for $p = 1$, $\frac{4\pi}{3}$ for $p = 2$, and 8 for $p = \infty$.

[7]We use the term "template" for a set of the parameter values constituting a hypothesis space.

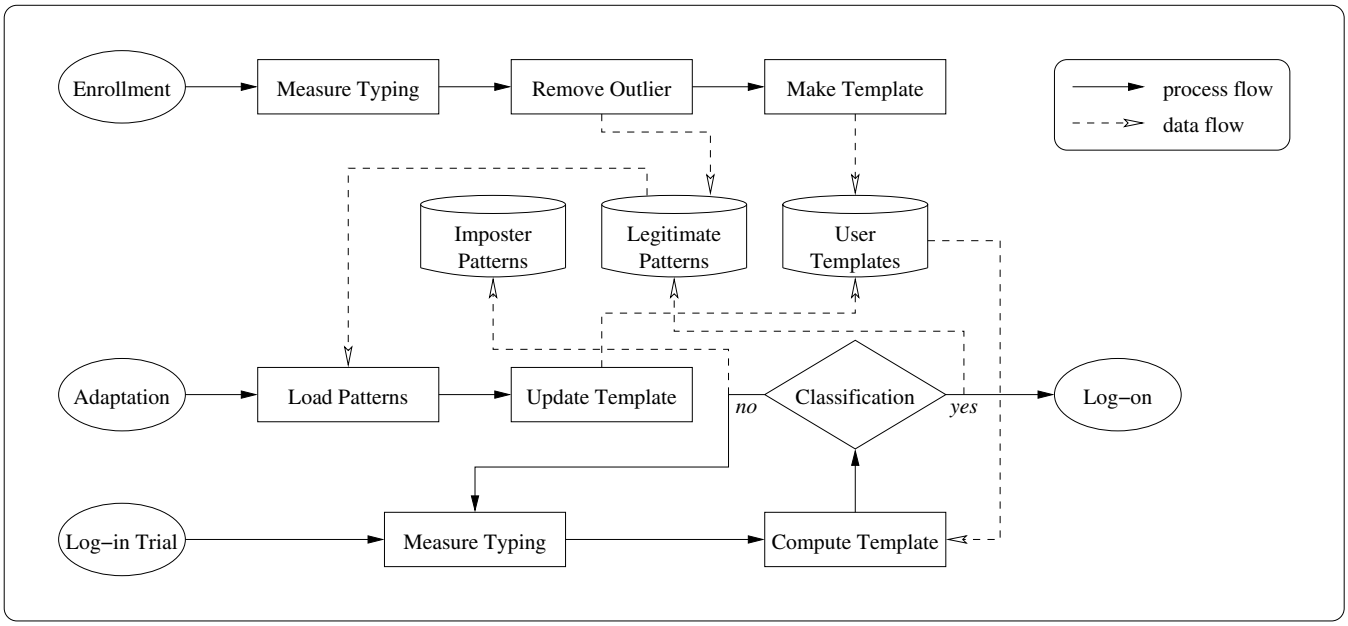


Figure 4: Flowchart of overall system

create initial population of a fixed size;
do {
  choose *parent1* and *parent2* from population;
  offspring ← crossover(*parent1*, *parent2*);
  local-optimization(*offspring*);
  replace(population, *offspring*);
} **until** (stopping condition);
return the best individual;

Figure 5: The outline of the hybrid genetic algorithm

client with a high-speed communication line, a small quantity of delay may produce a fatal error because durations and latencies are extremely short. Therefore, it is troublesome to measure a timing vector on the network generating a random delay. We adapt a Java applet into the system so as to prevent the network environment from affecting a measurement. When a server sends an applet byte code to a client, the web browser of a client executes the applet on Java Virtual Machine. The measurement is independent of the network environment because an applet runs on the local machine.

4.3 Genetic Framework

We used a hybrid steady-state genetic algorithm. Figure 5 shows the outline of the hybrid genetic algorithm. The details are described in the following.

4.3.1 Representation and Evaluation

A hypothesis space consists of $2n$ parameters concerned with $\mathbf{c}$ and $\mathbf{r}$ in the definition of the extended p-norm, where n is the dimension of a timing vector. We simply use the one-dimensional array to represent parameters. To evaluate a chromosome, the value of the pseudo-volume of the corresponding hypothesis space is used.

4.3.2 Selection and Crossover

Proportional selection is used for parent selection. The probability that the best chromosome is chosen is set to four times higher than the probability that the worst chromosome is chosen. Two-point crossover is used to produce offspring, and mutation operators are not used.

4.3.3 Replacement and Stopping Criterion

The offspring replaces the inferior out of the two parents if the offspring is not worse than both parents. Otherwise, it replaces the most inferior member of the population. The genetic algorithm stops after a given number of generations, and it also stops if all the chromosomes in the population are the same.

4.3.4 Local Optimization

An approximate gradient method is used for local optimization, since the pseudo-volume function defined in equation (4) is not differentiable. The approximation gradient at a point is obtained by searching the neighborhoods of the point: For each coordinate, a small constant is added to, or subtracted from the coordinate value of the point and the direction, in which the value of pseudo-volume is increased, is chosen as the element of the approximate gradient. To find a local minimum of the pseudo-volume function, we take steps proportional to the negative of the approximate gradient.

5. EXPERIMENTAL RESULTS

5.1 Data Sets

We collected the data sets from 16 participants. Each participant was requested to type the 10 passwords in Table 1 repeatedly. The 10 participants among them were each considered as the legitimate user for one password, each played the role of an imposter for the passwords. For each password, the other 15 participants were considered as imposters

Table 1: Password strings used in the experiments

No.	Password	Dimension	Number of Timing Vectors
1	qlalfqjsgh	19	338
2	rhdWkdudghk	21	339
3	rntk1tod	15	317
4	tjdnf1945	17	335
5	j6kbkeakd	17	359
6	transaction	21	342
7	DoItYourself	23	348
8	money4nothing	25	340
9	Sk8erBoi	15	327
10	FvVohx7x9P	19	338
Ave.		19.2	338.3
Min.		15	317
Max.		25	359

except the legitimate user. The passwords 1 to 5 are the Korean words on the English keyboard, and the passwords 6 to 9 are the English words. The password 10 is a random string including upper-cases, lower-cases, and numbers. The password strings were chosen carefully with regard to the familiarity and the typing difficulty on our own. The dimensions of the passwords and the numbers of samples are shown in Table 1. We used 20 samples of a legitimate user in making a hypothesis space, and the rest of the samples were used for test.

5.2 The Shapes of the Hypothesis Spaces

We proposed the extended p-norm to describe the hypothesis space. As shown in Figure 3, the value of p determines the shape of a hypothesis space. To find the proper shape of a hypothesis space, we tested for the hypothesis spaces of $p = 1$, $p = 2$, and $p = \infty$. The results of the test is shown in Table 2. Bold faces figures represent the best pair of results with respect to the total error rate[8] in the corresponding row. Each of the extended 1-norm and the extended 2-norm showed the best in 4 cases among the 10 passwords. Especially for the password 5, the extended 1-norm showed nearly zero error rates in the classification. The extended 2-norm showed relatively small error rates for the password 4. We could not find a considerable superiority between the extended 1-norm and the extended 2-norm, but it was observed that the extended infinity-norm performed poor in the test. Choosing the value of γ is another question. If the value of γ is too low (high security level), the FAR is nearly zero, but the FRR is too high. On the other side, if the value of γ is too high (low security level), the FRR is nearly zero, but the FAR is too high. With the current version, the values between 5 and 15 seems to be reasonable.

5.3 Effects of Eliminating Outliers

To observe the effects of eliminating outliers, we examined the hypothesis spaces using the extended 2-norm for $\alpha = \infty$, $\alpha = 2$, and $\alpha = 1$. If the value of α is infinite in inequality (6), no elimination occurs. The experimental results are shown in Table 3. For the 7 passwords excepting the passwords 2, 4, and 7, the error rates were reduced by eliminating outliers. It was observed that the hypothesis space for $\alpha = 2$ performed better than the hypothesis space for $\alpha = 1$ on average. We suspect that it is because most of the elements in the timing vectors were eliminated when $\alpha = 1$.

[8]The rate of typing instaces in which a legitimate user is denied or an imposter is allowed.

Table 3: The error rates for eliminating outliers

No.	Extended 2-norm					
	$\alpha = \infty$		$\alpha = 2$		$\alpha = 1$	
	FRR†	FAR‡	FRR	FAR	FRR	FAR
1	4.41	3.51	4.41	3.51	**2.45**	**3.51**
2	**9.76**	**0.88**	10.24	0.88	14.63	3.51
3	7.25	4.81	**5.70**	**5.77**	7.25	2.88
4	**1.60**	**0.79**	4.26	1.57	3.19	1.57
5	2.94	0.74	0.00	2.22	**0.49**	**1.48**
6	3.68	1.52	**3.68**	**0.76**	7.37	0.00
7	**11.27**	**5.22**	13.62	8.70	11.74	8.70
8	3.90	6.09	**3.90**	**5.22**	4.39	5.22
9	1.06	7.63	2.65	1.69	**1.59**	**2.54**
10	4.50	27.12	**2.00**	**11.02**	1.50	17.80
Ave.	5.04	5.83	5.05	4.13	5.46	4.72
Min.	1.06	0.74	0.00	0.76	0.49	0.00
Max.	11.27	27.12	13.62	11.02	14.63	17.8

† False reject rate.
‡ False accept rate.

Table 4: The error rates with the adaptation

No.	Extended 2-norm			
	Normal		Adaptation	
	FRR	FAR	FRR	FAR
1	2.45	3.51	**2.94**	**2.63**
2	14.63	3.51	**7.32**	**5.26**
3	**7.25**	**2.88**	5.70	7.69
4	3.19	1.57	**2.13**	**2.36**
5	**0.49**	**1.48**	1.47	8.89
6	7.37	0.00	**3.16**	**3.03**
7	11.74	8.70	**6.57**	**3.48**
8	4.39	5.22	**6.34**	**1.74**
9	**1.59**	**2.54**	3.70	5.08
10	1.50	17.80	**4.00**	**3.39**
Ave.	5.46	4.72	4.33	4.36
Min.	0.49	0.00	1.47	1.74
Max.	14.63	17.80	7.32	8.89

5.4 Improvements by the Adaptation

The proposed adaptation mechanism utilizes the results of the classification. The measured timing vector is used in updating a hypothesis space, if it is classified as the legitimate user. We tested the hypothesis spaces with the adaptation mechanism using the extended 2-norm. To avoid a heavy load, we executed the adaptation process once whenever 20 samples were collected. The experimental results are shown in Table 4. The adaptation mechanism improved the performance of the system on average. However, it is observed that the error rates for the passwords 3, 5, and 9 were increased slightly by the adaptation mechanism. We believe it was because the misclassification in the early stage misled the hypothesis space.

Table 2: Comparison of the hypothesis spaces

No.	Extended 1-norm				Extended 2-norm				Extended infinity-norm			
	$\gamma = 7.5$		$\gamma = 15$		$\gamma = 7.5$		$\gamma = 15$		$\gamma = 7.5$		$\gamma = 15$	
	FRR†	FAR‡	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR
1	49.02	0.00	3.92	4.39	38.73	0.00	**5.88**	**2.63**	80.39	0.00	48.04	0.00
2	**7.80**	**2.63**	3.41	50.00	9.76	0.88	6.34	38.60	45.85	0.00	23.90	0.88
3	12.44	0.96	2.07	20.19	26.94	0.00	**9.33**	**3.85**	15.54	4.81	9.33	19.23
4	1.06	3.15	0.53	53.54	**1.60**	**0.79**	1.06	41.73	57.98	0.00	25.53	0.00
5	12.75	0.00	**0.49**	**0.00**	14.22	0.00	2.94	1.48	33.82	0.00	14.22	0.00
6	**2.63**	**0.00**	0.53	5.30	10.00	0.00	3.68	2.27	68.42	0.00	29.47	0.00
7	8.92	18.26	1.88	42.61	8.92	22.61	2.82	51.30	**11.74**	**3.48**	7.04	26.96
8	5.85	0.00	**1.46**	**6.09**	25.85	0.00	11.71	0.87	62.93	0.00	43.90	0.00
9	0.53	44.92	0.00	69.49	1.06	35.59	1.06	62.71	**1.59**	**0.85**	1.06	3.39
10	4.50	32.20	0.00	74.58	**18.50**	**20.34**	0.00	57.63	65.50	1.69	26.00	15.25
Ave.	10.55	10.21	1.43	32.62	15.56	8.02	4.48	26.31	44.38	1.08	22.85	6.57
Min.	0.53	0.00	0.00	0.00	1.06	0.00	0.00	0.87	1.59	0.00	1.06	0.00
Max.	49.02	44.92	3.92	74.58	38.73	35.59	11.71	62.71	80.39	4.81	48.04	26.96

† False reject rate.
‡ False accept rate.

5.5 Comparison to Other Approaches

As mentioned, statistical methods estimate the distributions of the timing vectors statistically, and classify an input based on the distributions. Unfortunately, most literatures on the statistical methods do not provide the details for implementation, and thus we could not implement them precisely. Instead, we implemented a version of the statistical method reflecting the main ideas in [16, 14], and tested it with our data sets. The empirical results showed that the error rates of the method are higher than the error rates of ours. Although the implemented version may be different from the original version in the literatures, the results partially support that the proposed method is competitive with the statistical methods.

On the other hand, a method based on neural network was also implemented following [22, 4]. It showed comparable performance to our method. However, to achieve such performance, it has to use a large number of samples as well as the imposter's timing vectors which are not available in the training phase. Since it is impractical to assume a large number of training sets including the imposters' timing vectors, it is unfair and of little meaning to compare the neural network-based approach to our method.

6. CONCLUDING REMARKS

In this paper, we proposed a keystroke biometric method based on ellipsoidal hypothesis space. A hybrid genetic algorithm was used to find a proper hypothesis space according to the pseudo-volume. We obtained remarkable performance with the proposed method using an appropriate number of training sets. Empirical results for a number of test sets were given which support our claims.

Keystroke biometrics is a cheap biometric method because there is no need for any additional hardware except a keyboard. However, the error rates of the keystroke-based authentication is relatively high as compared to other biometrics. In spite of the high error rates, keystroke biometrics is very useful for securing passwords, because an intrusion into the system can be made only after the password secrecy is broken. Even a loose standard may greatly help the security.

It may be intrinsically not possible to perfectly screen imposters out primarily due to the instability in human typing and the limit in collecting data. However, since it is only a supportive authentication system, the proposed method can be used in practical situations. For example, when a suspicious connection is detected, the system may send a watch signal to the administrator and collects all the information about the connection.

7. ACKNOWLEDGMENTS

This work was supported by the Brain Korea 21 Project in 2007. The ICT at Seoul National University provided research facilities for this study. The authors wish to thank all the participants in our experiments.

8. REFERENCES

[1] L. C. F. Araújo, L. H. R. Sucupira, M. G. Lizárraga, L. L. Ling, and J. B. T. Yabu-uti. User authentication through typing biometrics features. In *ICBA*, pages 694–700, 2004.

[2] F. Bergadano, D. Gunetti, and C. Picardi. User authentication through keystroke dynamics. *ACM Transactions on Information and System Security*, 5(4):367–397, November 2002.

[3] M. Brown and S. J. Rogers. User identification via keystroke characteristics of typed names using neural networks. *International Journal of Man-Machine Studies*, 39(6):999–1014, 1993.

[4] S. Cho, C. Han, D. H. Han, and H.-I. Kim. Web-based keystroke dynamics identity verification using neural network. *Journal of Organizational Computing and Electronic Commerce*, 10(4):295–307, 2000.

[5] S. Cho and S. Hwang. Artificial rhythms and cues for keystroke dynamics based authentication. In *International Conference on Biometrics*, pages 626–632, 2006.

[6] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, second edition, 2000.

[7] S. T. de Magalhães, K. Revett, and H. M. D. Santos. Password secured sites – stepping forward with keystroke dynamics. *nwesp*, 0:293–298, 2005.

[8] W. G. de Ru and J. H. P. Eloff. Enhanced password authentication through fuzzy logic. *IEEE Expert*, 12(6):38–45, 1997.

[9] P. Dowland, S. Furnell, and M. Papadaki. Keystroke analysis as a method of advanced user authentication and response. In *SEC*, pages 215–226, 2002.

[10] S. Furnell, J. P. Morrissey, P. W. Sanders, and C. T. Stockel. Applications of keystroke analysis for improved login security and continuous user authentication. In *SEC*, pages 283–294, 1996.

[11] J. D. Garcia. Personal identification apparatus, 1986. United States Patent US 4,621,334, November 4, 1986.

[12] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley, 1989.

[13] D. Gunetti and C. Picardi. Keystroke analysis of free text. *ACM Transactions on Information and System Security*, 8(3):312–347, August 2005.

[14] S. Hocquet, J.-Y. Ramel, and H. Cardot. Fusion of methods for keystroke dynamic authentication. In *AutoID*, pages 224–229, 2005.

[15] J. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[16] R. Joyce and G. Gupta. Identity authentication based on keystroke latencies. *Communications of the ACM*, 33(2):168–176, February 1990.

[17] D. V. Klein. Foiling the cracker: A survey of, and improvements to, password security. In *Proceedings of the 2nd USENIX Security Workshop*, pages 5–14, August 1990.

[18] J. J. Leggett and G. Williams. Verifying identity via keystroke characteristics. *International Journal of Man-Machine Studies*, 28(1):67–76, 1988.

[19] D.-T. Lin. Computer-access authentication with neural network based keystrokeidentity verification. *International Conference on Neural Networks*, 1:174–178, June 1997.

[20] F. Monrose, M. K. Reiter, and S. Wetzel. Password hardening based on keystroke dynamics. *International Journal of Information Security*, 1(2):69–83, February 2002.

[21] F. Monrose and A. D. Rubin. Authentication via keystroke dynamics. In *ACM Conference on Computer and Communications Security*, pages 48–56, 1997.

[22] M. S. Obaidat and B. Sadoun. Verification of computer users using keystroke dynamics. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 27(2):261–269, 1997.

[23] J. W. Sammon Jr. A non-linear mapping for data structure analysis. *IEEE Transactions on Computers*, 18(5):401–409, 1969.

[24] R. Spillane. Keyboard apparatus for personal identification. *IBM Technical Disclosure Bulletin*, 17(3346), April 1975.

[25] D. A. Umphress and G. Williams. Identity verification through keyboard characteristics. *International journal of man-machine studies*, 23(3):263–274, 1985.

[26] J. R. Young and R. W. Hammon. Method and apparatus for verifying an individual's identity, 1989. United States Patent US 4,805,222, February 14, 1989.

[27] E. Yu and S. Cho. GA-SVM wrapper approach for feature subset selection in keystroke dynamics identity verification. In *Proceedings of the International Joint Conference on Neural Networks*, pages 2253–2257, July 2003.