Investigating the Discriminative Power of Keystroke Sound

Joseph Roth *Student Member, IEEE,*, Xiaoming Liu, *Member, IEEE,* Arun Ross, *Senior Member, IEEE,* and Dimitris Metaxas, *Member, IEEE*

***Abstract*—The goal of this paper is to determine if keystroke sound can be used to recognize a user. In this regard, we analyze the discriminative power of keystroke sound in the context of a continuous user authentication application. Motivated by the concept of digraphs used in modeling keystroke dynamics, a virtual alphabet is first learned from keystroke sound segments. Next, the digraph latency within pairs of virtual letters, along with other statistical features, are used to generate match scores. The resultant scores are indicative of the similarities between two sound streams, and are fused to make a final authentication decision. Experiments on both static and free text-based authentication on a database of 50 subjects demonstrate the potential as well as limitations of keystroke sound.**

***Index Terms*—Keystroke sound, keystroke dynamics, keyboard typing, continuous authentication.**

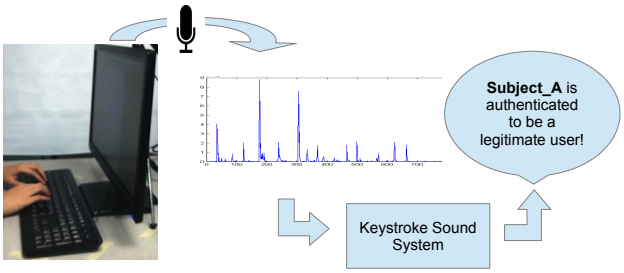


Fig. 1. Studying the discriminative power of *keystroke sound*. The sound of a user typing on the keyboard is captured by a simple microphone attached to the PC and is the input to the proposed system, which matches the characteristic of the acoustic signals to that of the claimed identity.

I. Introduction

Given the role of the keyboard in contemporary society, a number of research directions have been developed around it. First, motivated by the telegraph in the 19^{th} century, researchers discovered that the keystroke timing information varies across users. This led to the development of *keystroke dynamics*, which utilizes the keystroke timing information for user authentication [2], [12], [28]. Second, computer security researchers have used the keystroke sound for acoustic cryptanalysis. For example, Asonov and Agarwal presented a learning-based approach to identify the pressed keys using the keystroke sound [1] which was furthered by Zhuang *et al.* [29].

In this work, we consider another pertinent question: What is the discriminative capability of keystroke sound? Besides academic curiosity, an answer to this question can result in incorporating keystroke sound as an additional *biometric* cue in an active authentication framework. Furthermore, *forensic* applications can be developed based on preliminary analysis of keystroke sound.

However, in order to answer the aforementioned question, we need to first address the following issues: (a) How do we design an automated approach to extract discriminative information from keystroke sound? and (b) How do we utilize this approach to verify the identity of a subject using a keyboard?

J. Roth, X. Liu, and A. Ross are with the Department of Computer Science and Engineering, Michigan State University East Lansing, MI 48824.

D. Metaxas is with the Computer Science Department, Rutgers University, Piscataway, NJ 08854.

Corresponding author: Xiaoming Liu, liuxm@cse.msu.edu

Therefore, motivated by both scientific curiosity and potential applications, we present a systematic study on the discriminative power of keystroke sound. A basic overview is shown in Figure 1. Given the sound of the keys as a user types recorded by a microphone, our proposed system performs feature extraction and matching, and verifies the identity of the user. The subject of our study, keystroke sound, has a number of benefits. First, while it does require an external sensor, microphones are inexpensive and standard peripheral devices readily available in many PCs, laptops, monitors, and webcams. Second, the capture and analysis of keystroke sound does not interfere with a user's normal computer operation. Third, unlike keystroke dynamics, keystroke sound avoids the explicit logging of keys and hence the text being typed cannot be easily divulged. Finally, in our experiments, we demonstrate that in the unconstrained typing scenario keystroke sound has a shorter verification time, *i.e.*, the time required to make an authentication decision [10], than keystroke dynamics. Keystroke sound can be confounded by environmental noise, but the use of appropriate audio filtering or a directed microphone can mitigate this problem.

Our technical approach to match two keystroke sound signals is inspired by a combination of prior work in keystroke dynamics and acoustic emanations [1]. One of the most popular features in keystroke dynamics is digraph latency [3], [14], [15], which calculates the time difference between pressing the keys of two letters in succession. It has been shown that word-specific digraphs are more discriminative than a generic digraph, which is computed without regard to which letters were typed [22]. Assuming that the acoustic signal from a keystroke does not explicitly carry the information of what letter is typed, we propose a novel approach to employ the

digraph feature by constructing a *virtual alphabet*. Given the acoustic signals from all training samples, we first detect segments of keystrokes, whose Mel-Frequency Cepstral Coefficients (MFCC) [4] are fed into a K-means clustering routine. Each resultant cluster centroid is considered as a virtual letter and their collection is considered as a virtual alphabet, which enables us to compute the most frequent digraphs (a pair of cluster centroids) and their statistical attributes for each subject. Based upon the virtual alphabet, we can also compute the histogram of keystrokes within an acoustic stream, which is very similar to the popular Bag-of-Words (BOW) approach in the computer vision community [6], [23]. In addition, we consider a number of other feature representation and scoring schemes. Eventually a score level fusion scheme is employed to determine whether a probe stream matches with the gallery stream. We collect a keystroke sound database of 50 subjects in a static text session where subjects type a fixed collection of text four times, and a free text session where subjects type an impromptu mail letter. Although most prior work on keystroke dynamics focus on static text, we study the matching of keystroke sound signals in both the static and free text sessions. A preliminary result of 11% Equal Error Rate (EER) on a test set of 35 subjects, where the remaining 15 subjects are used for training, indicates the potential to conduct future research to study this novel aspect of the keyboard.

A preliminary version of this work was published in the International Conference on Biometrics 2013 [21]. We have extended it in a number of ways: (i) focused on the discriminative analysis of the keystroke sound signal; (ii) proposed a new score function (histogram of virtual letters) that performs best among all four functions; (iii) performed sound matching using the free text session of our database; (iv) substantially reduced the EER of matching performance from 25% to around 11% on our database, despite the increased number of subjects.

In summary, this paper has three main contributions:

$\diamond$ We investigate the discriminative power of keystroke sound, which has potential applications in forensics and biometric authentication.

$\diamond$ We collect a first-of-its-kind sound database of users typing on a keyboard. The database and the experimental results are made publicly available so as to facilitate future research and performance comparison on this research topic.

$\diamond$ We propose a novel virtual alphabet-based approach to learn various score functions from acoustic signals, and a score-fusion approach to match two sound streams.

II. Prior Work

In this section, we present a brief survey of keystroke dynamics as well as other applications of keystroke sound.

Keystroke dynamics, the habitual patterns and rhythms a user exhibits while typing on a keyboard, has a long history dating back to the use of telegraphs in the 19^{th} century and Morse Code in World War II, but most of the prior work still focus on static text [2], [12], [28], *i.e.*, all subjects type the same text. Only a few recent efforts have addressed the scenario of free text, *i.e.*, a subject can type arbitrary text, which is necessary for continuous authentication [16], [24]. However, free text keystroke dynamics still has a number of drawbacks. First, it requires *long* probe sequences to make a decision since the limited information from its digraph features requires a large number of pairs common to both the gallery and probe. For example, the work of Xi *et al.* [26] requires at least 700 characters which corresponds to more than three minutes of typing. This long verification period poses a security risk to continuous authentication since during this period the system is unsure of the identity of the typist. Second, everything the user types is explicitly recorded via key-logging. These limitations motivate us to explore other *complementary* means of user authentication based on interaction with the keyboard.

Typing behavior, the distinctive hand movements made by a user while typing, has been recently explored [20]. This work utilized a webcam pointed at the keyboard while the user types, and extracts dynamic shape information from the hands over time. While keystroke dynamics studies the temporal aspects, typing behavior studies the visual aspects; in this paper, we study the acoustic aspects of keyboard usage.

To the best of our knowledge, there are only two prior publications from one research group exploring the discriminative power of keystroke sound [5], [17]. They used a *combination* of keystroke dynamics with sound information to authenticate users typing the password "kirakira". The only feature extracted was the maximum sound level occurring for each key press. Our work differs from this in that we automatically estimate the key press timing without key logging and we do not impose any constraints on the keyboard, where users may type any text freely.

There has been a series of work on acoustic cryptanalysis in the computer security community. In their seminal work, Asonov and Agarwal [1] exploited keystroke sound to eavesdrop on a subject typing. They identified key presses and used a Fast Fourier Transform (FFT) feature-based classifier to recognize new key presses. Their system required extensive training of 100 presses per key, but still failed to identify the correct keys when trained and tested on *different* subjects. This type of error suggests that keyboard sound can potentially *differentiate* between subjects. Zhuang *et al.* [29] used an unsupervised method that clusters the keystroke sounds and uses English orthography and word frequency rules to recover the text in a *10-minute* audio recording. They demonstrated the superiority of MFCC features over FFT features. Kelley [11] re-implemented the aforementioned technique and also focused on the effects of errors made while typing. He noted two sources of errors in recovering text: the predominance of typing errors, which requires the usage of the *backspace* key to correct, and extraneous sounds produced by keyboard interactions that do not result in physical key presses. While these errors may present difficulties for recovering the typed text, they provide additional useful information for subject discrimination, which is not present in keystroke dynamics alone.

The focus of previous keystroke sound research has been on designing signal processing and machine learning algorithms to recover the typed letters. One of their main assumptions is that, when pressed, each key will emit a slightly different acoustic signal dependent upon the user. This motivates us to

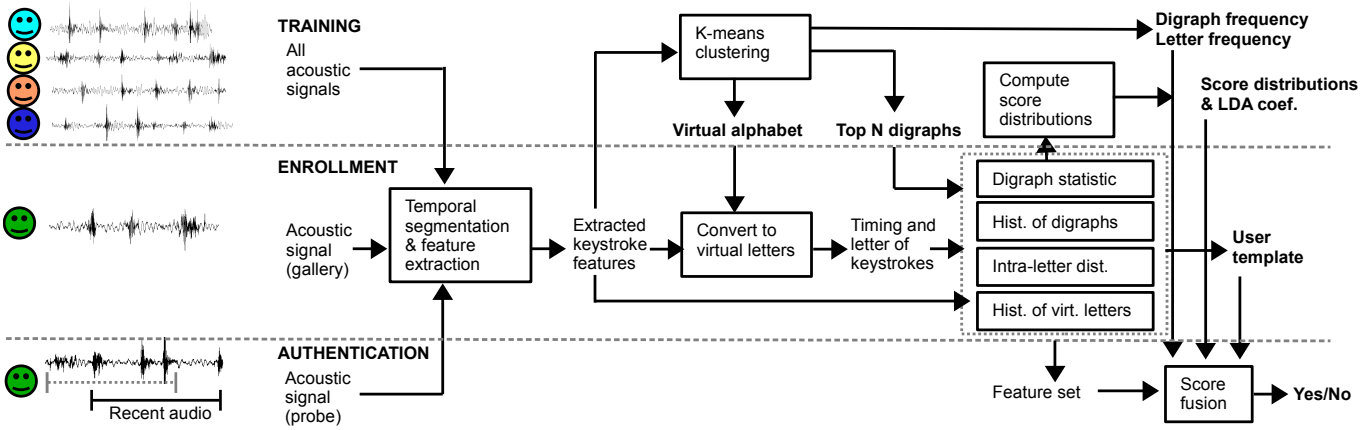


Fig. 2. Architecture of a biometric authentication system based on keystroke sound, where the boldface indicates the output of each one of the three stages: training, enrollment, and authentication.

cluster sample keystroke sounds to learn a *virtual alphabet* for the proposed approach. Assuming the continued success of this line of work in the future, it can be leveraged to combine the best of both worlds: keystroke dynamics-based authentication using recovered keys, and enhanced discrimination due to additional 1-D acoustic signals that are not present in key-logging.

III. Keystroke Sound Algorithm

To examine the discriminative ability of keystroke sound, we propose a system to match two sound streams in a continuous authentication application. Here we present a high level overview of the proposed system, ranging from audio recording to the authentication decision. Then we present in detail the various algorithmic components, which are used in the three stages of the system. We discuss the motivation and techniques used, as well as the input and output from each component.

A. System Overview

We formulate our algorithm as a pattern matching problem that takes gallery and probe sound streams as inputs and returns a similarity score between them. The gallery sound stream has its features pre-computed and stored as a user template during the enrollment stage. The probe sound stream is produced by the current user of the system who has to be recognized. Both sound streams are recorded with the subject typing in the same environment, which is described in detail in Section IV.

We briefly summarize the process in Figure 2. From [1], we know that keys produce unique sounds when pressed by the same user, but that different users produce slightly different sounds. In order to process the audio stream from typing, we must first identify the key presses from the background noise and extract frequency-based audio features describing each key press. To suppress some of the random noise effects we assign each press a *virtual letter*, which is simply the closest representative cluster of key press sounds. Then, all features are passed to a set of classifiers that jointly make a decision on whether the typing sounds are from the same user.

There are three different stages of operation for the system: training, enrollment, and authentication. During the *training* stage, a set of pre-recorded sound streams from multiple subjects is used to learn the various parameters of the algorithm suitable for the given environment. During the *enrollment* stage, a new subject types a pre-defined text while the system records the sound stream. It then estimates when keys were pressed, extracts the features from the stream, and creates and stores a user template for the subject. During the *authentication* stage, a subject claims his or her identity (*e.g.*, with a simple password) and the system then proceeds to continuously record the sound stream. The system extracts features from the sound stream in real-time and, after a sufficient length of time, compares them with the user template of the claimed identity to output a similarity score. If the computed similarity score is high enough, the subject is accepted and can continue operating the computer. Otherwise the subject is deemed an impostor and logged out of the system. In the following subsections, we present each component of this architecture in detail.

B. Temporal Segmentation & Feature Extraction

Let a raw acoustic typing signal, $g(t)$, be composed of keystroke acoustics interspersed with silent periods, which has muted non-deterministic background noise occurring throughout. $g(t)$ is recorded via a microphone at a specified sampling rate of f_s, where $f_s = 48\,\text{kHz}$ in this work. It is generally assumed that the keystroke acoustics and timing information carry all of the discriminative information about the typist, while the silent periods contain only background noises. Hence, before we extract features from the acoustic signal, we must first perform temporal segmentation of the keystroke, *i.e.*, estimating the times at which a keystroke occurs.

A keystroke is defined as the entire activity corresponding to a user pressing a key down, holding it, and releasing it to the upright position. A key press refers to the action of moving the finger down, striking the key, and the key striking

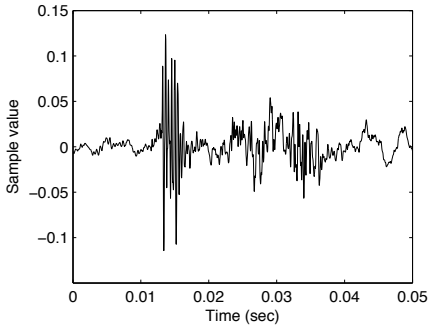


Fig. 3. The raw acoustic signal of a keystroke including key press and key release.

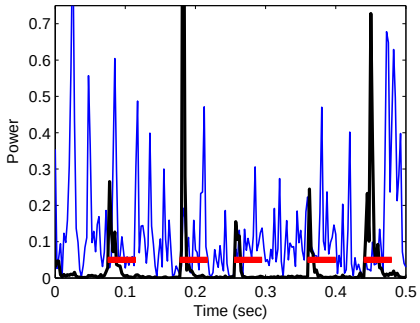


Fig. 4. Temporal segmentation by thresholding the FFT power. Black line is the power between 400 Hz and 12 kHz. The red lines indicate the duration L of the detected key press. Blue line is the power of the background noise at the lower frequencies (< 400 Hz).

the plate in the keyboard. A key release refers to the moving of the finger back into the upright position and the key snapping into its default location. Figure 3 shows a sample audio wave of a single keystroke. We see a clear peak at the key press, and a smaller, rougher peak at the key release. The sound of the key press is composed of the sound of the finger striking the key and the key striking the plate, but in practice, these sounds overlap each other in most cases. We denote a keystroke as $\mathbf{k}_i$, with a start time of t_i and a duration of L. Similar to prior work [29], we assume the keystroke duration is *constant* for all keys and subjects, because of the difficulty in precisely estimating the specific duration for each keystroke. Specifically, we set L to be 40 ms, since it covers the full length of most observed keystrokes.

Motivated by the work of Zhuang *et al.* [29], we conduct the temporal segmentation based on the observation that the energy of a keystroke is concentrated in the frequencies between 400 Hz and 12 kHz, while the consistent background noises (*e.g.*, hum of lights, computers, and HVAC systems) occupy mainly the lower frequency ranges. We compute the 5-ms windowed FFT of the acoustic signal $g(t)$ using a sliding window of a displacement of 2.5 ms, where the magnitudes of outputs in the range of 400 Hz and 12 kHz are summarized to produce an aggregate curve of the FFT power $p(u)$. By setting a threshold θ for $p(u)$, we can find the times u_i where $p(u_i - 1) < \theta \wedge p(u_i) > \theta$, as shown in Figure 4. Thus, we identify the start of keystrokes as $t_i = 2.5u_i$, where 2.5 is the sliding window displacement. In this work, we do not have the ground truth locations of when key presses occur to help guide the value of θ. We instead determine θ based on the number of key presses required to enter the static text without errors such that on average we recognize the correct number of key presses in the training data while rejecting superfluous background noise. In the future, an adaptive thresholding scheme could be employed to improve the segmentation for a given audio stream. Ideally this temporal segmentation should detect all keystrokes, instead of the background noise.

Once the start of a keystroke t_i is determined, we convert the acoustic signal within a keystroke segment, $g(t_i, ..., t_i + L)$, to a feature representation $\mathbf{f}_i$ for future processing. The standard MFCC features have demonstrated effectiveness in key recovery [29], even though they were initially developed for speech applications. Our MFCC implementation uses the same parameter setup as the work of Zhuang *et al.* [29]. That is, we have 32 channels of the Mel-Scale Filter Bank and use the first 16 resultant coefficients with 10-ms windows that are shifted by 2.5 ms. The resultant feature of a keystroke $\mathbf{k}_i$ is a 256-dimensional vector $\mathbf{f}_i$.

C. Virtual Alphabet via Clustering

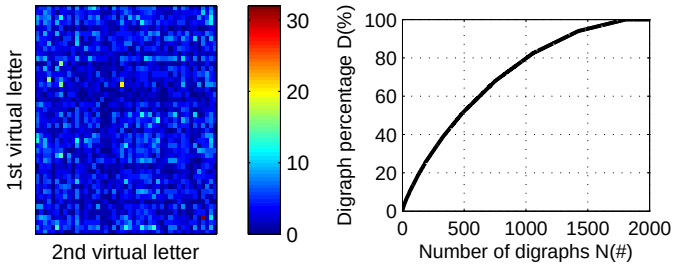
Most prior work of keystroke dynamics use digraph statistics - the time delay between two individual keys or two groups of keys, or trigraphs - the delay across three keys. In keystroke dynamics, such key information is readily available since key logging records the letter associated with each keystroke. However, this is not the case with the keystroke acoustic signal. We have estimated the timing of each keystroke, and now we need to estimate the label or the letter pressed at each keystroke. However, as shown in acoustic cryptanalysis [11], precisely recognizing the typed key from acoustic signals itself is an ongoing research topic.

Hence, we take a different approach by aiming to associate each keystroke with a unique *label*, with the hope that different physical keys will correspond to different labels, but also allowing different typists to generate different labels when pressing the same key which incorporates differing sound information. We call each label a *virtual letter*, the collection of which is called a *virtual alphabet*. Learning the virtual alphabet is accomplished by applying K-means clustering to the MFCC features of all keystroke segments in the training set. An acoustic signal is represented as a collection of key presses $\mathbf{K} = \{\mathbf{k}_i\}$, where each key press is a triplet $\mathbf{k}_i = \{t_i, \mathbf{f}_i, l_i\}$ and l_i is the corresponding virtual letter.

D. Score Functions

Given the aforementioned feature representation scheme, we next investigate a set of score functions to compute the similarity measure between two sets of features from the gallery and probe streams, as follows:

1) Digraph statistic: Our first score function is based on early work on keystroke dynamics. We use statistical features from only the digraph information, t_i and l_i. Since the virtual



(a) (b)

Fig. 5. (a) Heatmap of digraph occurrences in the training data. (b) Percentage of digraphs contained within the top N digraphs.

alphabet bears resemblance to the real letters within the same user, we expect high scores for genuine users and low scores for impostors. A digraph refers to the latency between presses of a pair of letters. There are two types of digraphs: word-specific digraphs and generic digraph [22]. Each word-specific digraph depends on one particular pair of letters, whereas the generic digraph is computed from all possible pairs of letters. We study both types of digraphs in this work.

With a virtual alphabet of K letters, we may generate up to K^2 unique word-specific digraphs and a single generic digraph. But as there are certain pairs of letters that do not follow each other in English, we expect the digraphs of virtual letters to follow an uneven distribution as well. During the training stage, we count the frequency of each digraph by passing through adjacent keystrokes, l_{i-1}^j and l_i^j, in the entire training set. Figure 5 (a) illustrates the occurrences of all possible digraphs in a set of training data. Then we generate a list of the top N most frequent digraphs, each denoted as $\mathbf{d}_n = \{k_{n1}, k_{n2}\}$, with corresponding digraph frequency as w_1^n. We set N based on a pre-defined constant D such that $\sum_{n=1}^{N} w_1^n \geq D$, *i.e.*, we incorporate the top D percent of most frequent digraphs in the score function. The relationship between N and D is displayed in Figure 5 (b). Section V will present the influence of D on the authentication performance.

Given the $\mathbf{K}$ representation of an acoustic signal, for each word-specific digraph $\mathbf{d}_n$, we compute the mean, m_n, and the standard deviation, σ_n, of the time difference variable $\Delta t = t_i - t_{i-1}$ where $l_i = k_{n2}$ and $l_{i-1} = k_{n1}$. Finally, the word-specific digraph similarity score between two arbitrary length signals, $\mathbf{K}$ and $\mathbf{K}'$, is computed using the following equation:

$$S_{1d}(\mathbf{K}, \mathbf{K}') = \sum_{n=1}^{N} w_1^n \left[\sum_{\Delta t} \sqrt{\mathcal{N}(\Delta t; m_n, \sigma_n^2)\mathcal{N}(\Delta t; m_n', {\sigma_n'}^2)} \right], \quad (1)$$

which basically sums up the overlapping region between two Gaussian distributions of the same digraph, weighted by w_1^n. The overlapping region is computed via the Bhattacharyya coefficient.

The generic digraph score function, on the other hand, is much simpler to compute. We compute the mean, m, and the standard deviation, σ, of the time difference variable $\Delta t = t_i - t_{i-1}, \forall i$, and the corresponding score function is,

$$S_{1i}(\mathbf{K}, \mathbf{K}') = \sum_{\Delta t} \sqrt{\mathcal{N}(\Delta t; m, \sigma^2)\mathcal{N}(\Delta t; m', {\sigma'}^2)}. \quad (2)$$

Algorithm 1: Feature extraction algorithm.

Input: A stream $g(t)$, top digraphs $\mathbf{d}_n$, cluster centroids $\mathbf{m}_f(k)$.
Output: A feature set $\mathbf{F}$.
Locate keystrokes $\mathbf{t} = [t_1, ..., t_i, ...]$ at times of high energy $p(u)$,
foreach *keystroke time* t_i **do**
  $\mathbf{f}_i$ = MFCC($g(t_i, ..., t_i + L)$),
  $l_i = \arg\min_k \|\mathbf{m}_f(k) - \mathbf{f}_i\|_2$,
m = mean($\{t_i - t_{i-1}\}$),
σ = std($\{t_i - t_{i-1}\}$),
foreach *digraph* $\mathbf{d}_n$ **do**
  Compute histogram of digraphs h_n via Eqn. (3),
foreach *letter* k **do**
  Compute histogram of virtual letters η_k via Eqn. (5),
  Compute $\bar{\mathbf{f}}_k$ via Eqn. (7),
return $\mathbf{F} = \{m, \sigma, \boldsymbol{\eta}, \mathbf{h}, \bar{\mathbf{f}}_k\}$.

Using the same experimental setup as Figure 9 on the database presented in Section IV, we find that S_{1d} is significantly slower to compute and also performs worse with an EER of 45%, compared to the EER of 30% based on S_{1i}. Therefore we choose to use the generic digraph statistic and denote its score as S_1. Note that this is different than what is observed in keystroke dynamics where the word-specific digraphs demonstrate superior performance than the generic digraph [22]. We hypothesize that keystroke segmentation errors and a greater possible number of unique digraphs than keystroke dynamics result in the better performance of the generic digraph.

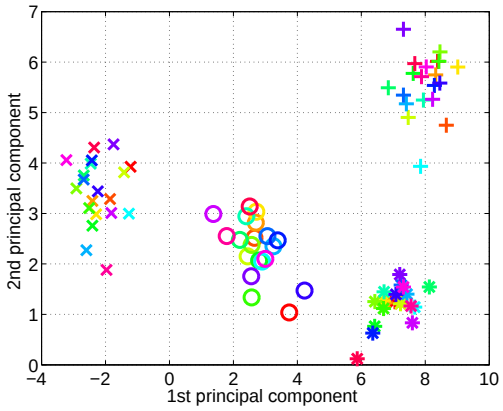
2) Histogram of digraphs: If subjects produce different virtual letters when typing the same text, they are also likely to generate different digraphs. Hence, we can use the frequencies of popular digraphs as the score function. In the previous subsection we describe the approach to compute the top N digraphs based on the frequency of occurrence within the training data. Given that, we compute the histogram, $\mathbf{h} = [h_1, h_2, ..., h_N]^\intercal$, of the top N digraphs, for each acoustic signal. That is,

$$h_n = \frac{\sum_i \delta(l_i = k_{n2})\delta(l_{i-1} = k_{n1})}{|\mathbf{K}| - 1}, \quad (3)$$

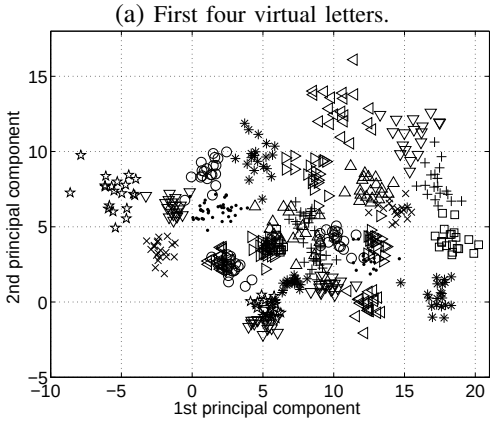
where δ is the indicator function and the numerator is the number of digraphs $\mathbf{d}_n = \{k_{n1}, k_{n2}\}$ within a sequence of length $|\mathbf{K}|$. The similarity score between two signals is simply the inner product between two histograms of digraphs,

$$S_2(\mathbf{K}, \mathbf{K}') = \mathbf{h}^\intercal \mathbf{h}'. \quad (4)$$

3) Histogram of virtual letters: When different subjects press the same key, different sounds may be produced. While trying to predict the text typed from the acoustic emanations, Asonov and Agrawal identified lower recognition rates when comparing between subjects [1]. This means that subjects produce different sounds while typing the same text and, therefore, examining the distribution of these sounds could discriminate among subjects. Motivated by this observation,



(a) First four virtual letters.



(b) All 30 virtual letters.

Fig. 6. The mean MFCC features $\bar{\mathbf{f}}_k$ of each of 20 subjects within virtual letters, plotted along the top-2 principle components reduced from the original 256-dimensional space. The symbol represents a virtual letter and the color in (a) indicates a subject (best viewed in color).

we compute the histogram, $\boldsymbol{\eta} = [\eta_1, \eta_2, ..., \eta_K]^\intercal$, of the K virtual letters as observed in each acoustic signal. That is,

$$\eta_n = \frac{\sum_i \delta(l_i = k_n)}{|\mathbf{K}|}, \tag{5}$$

where the numerator is the number of keystrokes assigned to virtual letter k_n. For this score function, the similarity score between two signals is the inner product between the two histograms,

$$S_3(\mathbf{K}, \mathbf{K}') = \boldsymbol{\eta}^\intercal \boldsymbol{\eta}'. \tag{6}$$

4) Intra-letter distance: We use the virtual letter to represent similar keystrokes emerging from different keys pressed by different subjects. Hence, within a virtual letter, it is very likely that different subjects will have different distributions. Figure 6 provides evidences for this observation by showing the mean MFCC features of 20 training subjects within virtual letters. It can be seen that 1) there is distinct inter-letter separation among virtual letters; 2) within each virtual letter, there is intra-letter separation due to individuality. Hence, we would like to utilize this intra-letter separation in our score function. For an acoustic signal, we compute the mean of $\mathbf{f}_i$ associated with each virtual letter, which results in K mean

Algorithm 2: Authentication algorithm.

Input: A probe stream $g'(t)$, a user template $\mathbf{F}$, top digraphs $\mathbf{d}_n$, cluster centroids $\mathbf{m}_f(k)$, score distributions m_{sv}, σ_{sv}, a threshold τ.
Output: An authentication decision d.
Compute feature set $\mathbf{F}'$ for probe $g'(t)$ via Alg. 1,
Compute digraph statistic score S_1 via Eqn. (2),
Compute histogram of digraphs score S_2 via Eqn. (4),
Compute histogram of virtual letters score S_3 via Eqn. (6),
Compute intra-letter distance score S_4 via Eqn. (8),
Compute normalized score S via Eqn. (9),
if $S > \tau$ **then**
  return d = genuine.
else
  return d = impostor.

MFCC features, as follows:

$$\bar{\mathbf{f}}_k = \frac{1}{|l_i = k|} \sum_{l_i = k} \mathbf{f}_i. \tag{7}$$

Given two acoustic signals $\mathbf{K}$ and $\mathbf{K}'$, we use Equation (8) to compute the Euclidean distance between the corresponding mean MFCC features and sum using a weight w_3^n, which is the overall frequency of each virtual letter among all keystroke segments and is pre-computed from the training set. The sign -1 is to ensure that, on average, the genuine probe has a larger score than the impostor probe.

$$S_4(\mathbf{K}, \mathbf{K}') = -\sum_{k=1}^{K} w_3^n ||\bar{\mathbf{f}}_k - \bar{\mathbf{f}}'_k||_2. \tag{8}$$

So far we have constructed a feature set for one acoustic signal, denoted as $\mathbf{F} = \{m, \sigma, \boldsymbol{\eta}, \mathbf{h}, \bar{\mathbf{f}}_k\}$, where $k \in [1, K]$. We summarize the feature extraction algorithm in Algorithm 1. If the acoustic signal is from the gallery stream, we call the resultant feature set as a *user template* of the gallery subject, which is computed during enrollment and stored for matching with a probe stream.

E. Score Fusion & Authentication Decision

Once the four scores are computed, we fuse them to generate one value that indicates the similarity between two sound streams. In this paper, we only consider fusion across multiple scores, since we focus on the ability of an individual probe to be matched with the correct user. In the future, a more rigorous continuous authentication score fusion will take temporal information into consideration, by integrating the previous score functions from the same computer session. In our system, we use a simple score-level fusion where the four normalized scores are reduced to a single score function through linear discriminate analysis (LDA) [13]. The optimal LDA projection vector $[c_1, c_2, c_3, c_4]^\intercal$ is learned on the scores of probes in the training set, such that the between-class scatter is maximized while minimizing the within-class scatter. The

final score is computed as follows:

$$S = \sum_{v=1}^{4} c_v \frac{S_v - m_{sv}}{\sigma_{sv}}. \quad (9)$$

To normalize the score functions, we use the mean m_{sv} and standard deviation σ_{sv} of the score distribution learned from the impostor examples in the training data, such that the normalized scores for the impostors will fall in a standard normal distribution and the genuine scores should be outliers on the positive side. We chose to only normalize based on the impostor scores because they follow a clear Gaussian-like distribution and according to [9], the z-score normalization can be used only when the data is Gaussian distributed. To make an authentication decision, a simple threshold τ is used to classify the user as genuine when $S > \tau$. We summarize the algorithm for the authentication stage in Algorithm 2.

TABLE I
AGE DISTRIBUTION OF SUBJECTS.

Age	10–19	20–29	30–39
Number of subjects	11	34	5

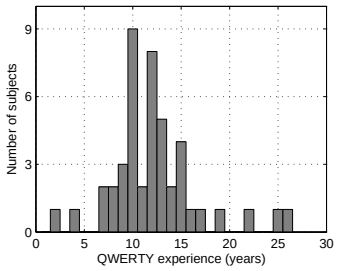


Fig. 7. Distribution of subjects' experience with keyboard.

IV. DATABASE

In this section, we present an overview of the database that we collected for this work, which is designed to help with other typing-based research as well. We present both the technical setup as well as the motivation behind the protocols for data collection. We have three main considerations when developing our protocol: 1) the text the subjects type, 2) the equipment on which they type, and 3) the environment in which they type.

Type of Text: When developing the protocol, our first goal is to be able to study the differences and dependencies of keystroke sound on static text and free text. Static text refers to typing of the exact same text during enrollment and authentication, which models typing of a password or a commonly repeated phrase such as an e-mail signature. Free text refers to allowing the subject freedom to choose the words and topics for typing, which models generic computer usage. For continuous authentication, the ability to work on free text is essential, but it could be more challenging due to the inherent differences in characters typed and keyboard activity. The question of static versus free text is pertinent to keystroke dynamics as well, where its research started with static text and substantially more efforts have been made on static text over free text in the past few years [2].

In order to answer this question with keystroke sound, we design our protocol to include two sessions. In the first session, we have the subject type static text by copying the first paragraph from "A Tale of Two Cities" by Charles Dickens, which is displayed on the monitor directly above the input text area. We further break the first session into four sub-sessions by asking the subject to repeat this typing exercise four times with a 2-3 second break between trials. Subjects are requested to remove their hands from the keyboard between sub-sessions in order to reset their position as well as frame of mind. Multiple typing instances of the same paragraph enable the study of static text-based authentication. In the second session, the subject is requested to type a half-page email to their family with no instructions on the content of the letter. We observe in this session, that subjects make spontaneous pauses during typing while they think of material to write, adjust their hands without pressing keys, and exhibit other real-world typing anomalies. This second session allows for research on free text-based authentication. Most subjects take between 5-8 minutes to type each session, depending on their typing ability and speed.

Equipment: Our second consideration is the equipment and setup. While we do not have complete control over the background environment, we could maintain the same physical equipment across all data collections. As demonstrated in the work of keyboard acoustic emanation [1], training on one keyboard and recognizing on a different keyboard, with the same brand and model, has adverse effects on the accuracy. For this reason, we use the same U.S. standard QWERTY keyboard for data collection. Although there are many available options for microphones, we decide to utilize an inexpensive webcam with an embedded microphone, which is centered on top of the monitor and pointed toward the keyboard. This setup uses commodity equipment and allows us to capture both the video of hand movement and the audio recording of the keyboard typing. Thus, a multi-modal (visual and acoustic) typing behavior analysis system can be developed in the future based on this database. The sound is captured at 48 kHz in dual channel, but based on our observations, these channels are almost identical and hence we combine them into a single channel, by simply averaging the two channels.

Environment: Our third consideration while collecting the database is the recording environment. The background noises present in the audio recording play an important role in the usability of the stream. Background noises can refer to voices, low frequency pitches from heaters, lights, or computers, other people typing, and any other sound not originating from the subject typing on the keyboard. These noises can both affect the sound of normal key presses, when they occur in sync with the subject typing, as well as pose difficulties in distinguishing between key press and background noise, when they are louder than the key presses themselves.

To mitigate background noises during collection, we direct the camera and microphone at the keyboard so the sounds from key presses are made prominent; we also communicate instructions to the subjects fully ahead of time and use non-

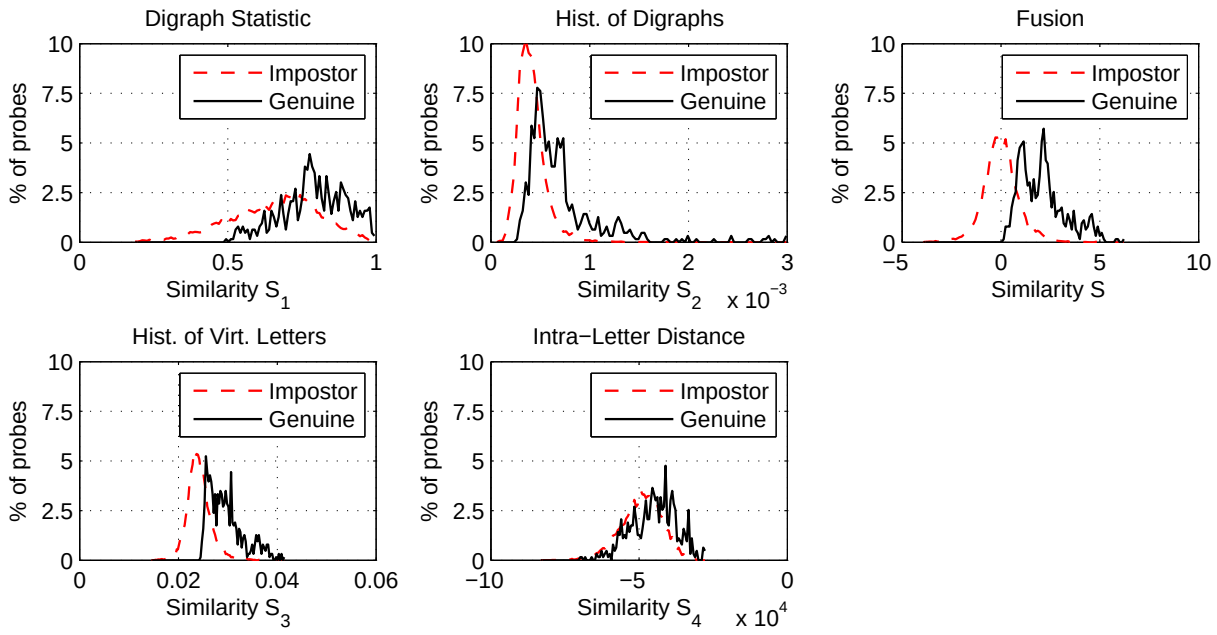


Fig. 8. Distribution of the individual score functions, S_1, S_2, S_3, and S_4, and the fused score S, for genuine and impostor probes.

verbal communication during the session to reduce the interference of our voices on the audio. In the algorithm design, we filter out the constant low frequency pitches when performing temporal segmentation to further remove background noise. We also attempt to maintain consistency in the background noise present in the recording by using the same table, chair, and position of keyboard, monitor, and webcam for all subjects. Nevertheless, some standard workplace noises exist in the background, *e.g.*, doors opening, people walking, and chairs rolling across the floor.

Subjects: Our subject pool consists of 50 individuals from different backgrounds. All subjects are either students or faculty members of Michigan State University, and were recruited through a broad announcement to engineering students or through personal referrals by other participants. Although the number of subjects (50) is preferred to be larger, it is on par with the number of subjects (51) in the well-known CMU benchmark keystroke database [12], which has been extensively tested on various keystroke dynamics studies. To study the various factors that may affect distinctiveness of keystroke sound, each subject is asked to finish a survey with four questions, *viz.*, the age group, years of experience in using keyboard, major type of keyboard, and years of experience in using QWERTY keyboards. The distribution of typing experience is reported in Figure 7, and the age distribution is summarized in Table I.

In order to facilitate further research on typing-based biometrics or to permit performance comparison between various approaches, we have released this database[1] for research purposes. This includes the four sub-sessions of the first session along with the training and testing set divisions as used in the experimental results.

[1]http://www.cse.msu.edu/~liuxm/typing

V. Experimental Results

The goal of this section is to provide a comprehensive analysis of the discriminative power of keystroke sound, using extensive experiments. The paper expands upon the experimental results presented in [21] by examining the effects of the new score function, searching the parameter space, and studying the unconstrained setting, *i.e.*, keystroke sound matching with free text. When a new biometric modality is introduced, it is a common practice to first evaluate its efficacy in constrained environments and then, as the technology matures, to consider operational unconstrained scenarios. For example, face recognition algorithms were initially tested on highly constrained databases such as FERET [19] and FIA [7], but are more recently being evaluated on unconstrained datasets such as LFW [8] and YouTube Faces [25]. From the acoustic realm, there is also text-dependent and text-independent speaker recognition, which is analogous to static and free text typing. Similarly, research on keystroke dynamic has mostly focused on static text for the past few decades and is progressing to free text in recent years. Following this research methodology, we mainly use the static text portion of our dataset, but also include experiments on the free text portion, which allows for true continuous authentication.

A. Setup

We refer to the four sub-sessions from the static text of the database as $S11$, $S12$, $S13$, and $S14$ and the free text session as $S2$. The proposed algorithm requires a separate training dataset for the purpose of learning a virtual alphabet, top digraphs, and the score distribution statistics. Hence, we randomly partition the database into 15 subjects for training and the remaining 35 subjects for testing our algorithm. We repeat this partitioning process 5 times to validate our results.

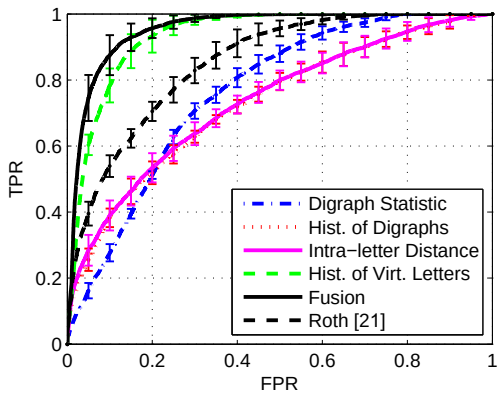


Fig. 9. The ROC curves of individual score functions as well as final fused score with error bars for $K = 60$ and $D = 20\%$.

Gallery and Probe: For each subject, we use the first typing trial, $S11$, as the gallery stream and portions of $S12$, $S13$, and $S14$ as the probe streams. For the probe streams, we need to balance two considerations. Firstly, we want to use as many probes as possible to enhance the statistical significance of our experiments, which requires that we use a partial sequence. Shorter probe sequences also allow for faster verification time to identify impostors quicker. Secondly, we want to use longer probe streams to allow accurate calculations of features for a given subject. We decide to form 7 continuous probe streams from each sub-session for each subject by using 70% of the paragraph starting at the 0%, 5%, 10%, 15%, 20%, 25%, and 30% mark of the paragraph. This overlap of text streams allows us to balance both considerations, while also simulating a continuous environment where the algorithm works with an incoming data stream and the results of the current probe build on the prior results. Although such overlapping creates dependency among testing probes, this fulfills both the requirement of periodic authentication, and the need to use a window of past observations to make an authentication decision. Note that the same gallery and probe partition is applied to both the training and testing set.

The average gallery and probe length is 94 and 62 seconds respectively. The total number of probe streams for training is 315 $(= 15 \times 3 \times 7)$ with 4725 $(= 315 \times 15)$ different cases. The total number of probe streams for testing is 735 $(= 35 \times 3 \times 7)$ with 25725 $(= 735 \times 35)$ different cases.

Evaluation Metrics: We use the standard Receiver Operating Characteristic (ROC) curve, as the main performance metric. The ROC curve has two axes: False Positive Rate (FPR), the fraction of impostor pairs incorrectly deemed genuine, and True Positive Rate (TPR), the fraction of genuine pairs correctly deemed genuine. A good biometric produces a low FPR at high values of TPR. To succinctly summarize the ROC curve, we use the Equal Error Rate (EER) which is the FPR when it equals $1-$TPR. In order to compare the performance of the score functions, we also plot the probability distributions of the genuine and impostor scores.

TABLE II
EER OF PARAMETER SEARCH FOR K AND D.

D \ K	**20**	**30**	**45**	**60**	**75**
10%	16.3	13.7	12.8	12.7	12.3
20%	17.7	14.3	12.5	**11.0**	14.9
50%	16.3	15.7	14.0	13.4	14.4
70%	16.9	15.8	12.9	12.3	14.1
90%	17.2	15.4	14.3	12.9	15.0

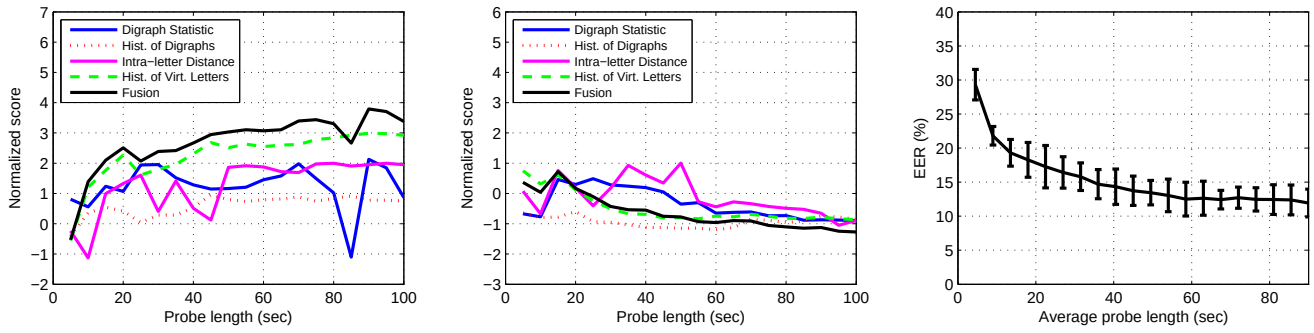
B. Score Function Comparison

Figure 8 presents the distributions of the four score functions and the overall fused score on one partition of the testing data. Figure 9 displays the authentication performance with tuned parameters after evaluating the algorithm on all five partitions, with each one of the score functions, the approach in our earlier work [21], and the fused score proposed in this paper. We can make a number of observations. Firstly, the individual score function distributions all display significant overlap between the genuine and impostor pairs. The task of identifying a single feature representation to discriminate users via keystroke sounds is challenging. Intra-letter distance, digraph statistic, histogram of digraphs, and histogram of virtual letters provide 34%, 33%, 30%, and 13% EER, respectively. Secondly, despite the overlap, there is still some separation between the genuine and impostor probes. Furthermore, by using fusion, we create a new fused score, which produces the best result and indicates that the individual score functions capture different aspects of the subject's typing sound. The result with the fused score has an EER of 11.0%. Finally, we have achieved substantially better performances compared to our earlier approach presented in [21], which has an EER of 24.2%.

C. Parameter Tuning

There are two different parameters for our algorithm, which are not deduced from the training set. First, the number of virtual letters, K, which has implications on the mapping of real keys on the keyboard. If K is less than the number of keys on the keyboard, it forces multiple real keys to be mapped onto the same virtual letter which can make the virtual digraphs meaningless. If K is greater than the number of keys, it forces different users pressing the same key to map to different virtual letters, which increases the total number of digraphs and could require longer probe sequences to make a reasonable decision. We seek to find a good balance for K by looking at 20, 30, 45, and 60 with the realization that about 30 keys are used on a keyboard in normal typing. Second, the number of top digraphs N is changeable. We set N based on the top $D = \{10\%, 20\%, 50\%, 70\%, 90\%\}$ of all digraphs that are included in the calculation. For example, when $D = 70\%$, we use the top $N = 797$ digraphs.

From Table II, it can be seen that as the number of virtual letters K increases, the authentication performance improves with decreasing EER, which is consistent with our intuition that a virtual letter represents the sound of a unique key pressed by a subset of subjects. This performance, however, saturates after $K = 60$ at which point we include enough



(a) (b) (c)

Fig. 10. (a-b) The authentication score over time for one specific genuine user (a) and impostor user (b). As time passes, the fused score stabilizes to the correct decision. We can observe the fused score rectifies the various fluctuations in individual scores. (c) EER of keystroke sound authentication with $S11$ gallery and differing percents of $S12$, $S13$, and $S14$ as probes. Performance stabilizes at 70% or ∼62 seconds of typing.

sounds to handle the different keys and means of pressing them. In comparison, as the percentage of selected digraphs D increases, the performance improvement is not as obvious as that of K, but it does improve slightly before declining when unimportant or unused digraphs are included. K contributes more to authentication due to the superiority of the histogram of virtual letters, which only depends on K, rather than D. Finally, the best performance (EER $= 11.0\%$) is achieved when $K = 60$ and $D = 20\%$.

D. Feature Correlation

To ensure good score-level fusion, it is desirable that scores are uncorrelated [18]. In Table III we examine the Pearson correlation coefficient, $p = \frac{cov(x,y)}{\sigma_x \sigma_y}$, of all combinations of score functions to identify the linear dependence of each score with each other. In doing so, we discover high correlation between the histogram of virtual letters and the histogram of digraphs scores. This may help explain why in Table II we see that increasing the number of digraph features does not improve the overall performance. The remaining features all exhibit weak linear correlation, which does not guarantee independence, but still contributes to the increased performance of the fused score.

E. Verification Time

One important question with continuous authentication is the time taken to either authenticate a genuine user or detect an impostor. Ideally this verification time [10] should be as short as possible in practical applications. To answer this question, we design an experiment to determine what length of probes is necessary to reach a reasonable decision. Using $S11$ as gallery, we vary the length of probes in $S12$, $S13$, and $S14$ by 5% across the entire length of the probes. Figures 10 (a-b) demonstrates how the score functions change over time for one specific genuine and impostor probe. In this example, we see that both fluctuate in uncertainty near the beginning when limited information is present, and they eventually stabilize to the correct decisions as time passes. Figure 10 (c) shows the EER for this experiment over the length of the probe. We see large errors using short probes with a rapid improvement from 20% of the probe length or ∼18 seconds to 70% of the probe length or ∼62 seconds.

TABLE III
PEARSON CORRELATION COEFFICIENTS p OF FOUR SCORES: DIGRAPH STATISTIC (DS), HISTOGRAM OF DIGRAPHS (HD), INTRA-LETTER DISTANCE (ILD), AND HISTOGRAM OF VIRTUAL LETTERS (HVL).

	DS	**HD**	**ILD**	**HVL**
DS	1.000			
HD	0.026	1.000		
ILD	0.148	0.103	1.000	
HVL	0.155	**0.760**	0.081	1.000

F. Number of Enrollment Samples

We hypothesize that repeating the enrollment session to create a set of user templates for each subject can capture more of the intra-subject variation and therefore improve the performance. If M gallery streams exist for each subject, the fused score S^i can be computed against each of the M user templates and the final score for the user can simply be the mean of fused scores, $S = \frac{1}{M}\sum_{i=1}^{M} S^i$. To validate our hypothesis about multiple gallery sequences, we use the seven 70% partitions of $S14$ as the probe and use all combinations of $S11$, $S12$, and $S13$, for 1, 2, and 3 gallery sequences. Furthermore, we perform the experiments 5 times for cross-validation. The EERs are 12.2%, 10.6%, and 10.2%, for 1, 2, and 3 gallery sequences respectively, which indicates that multiple galleries have a positive impact on the performance. Note that in this experiment, multiple gallery sequences are keystroke sounds when typing the same static text multiple times. In the future, when one subject has multiple gallery sequences with free texts, we would better capture the intra-subject variation and expect a larger margin of performance improvement for free text-based authentication.

G. Computational Efficiency

Since a probe stream is an one-dimensional signal, our algorithm can operate comfortably in real time, with very minimal CPU load, which is a very favorable property for continuous authentication. Our experiments were run on a commodity

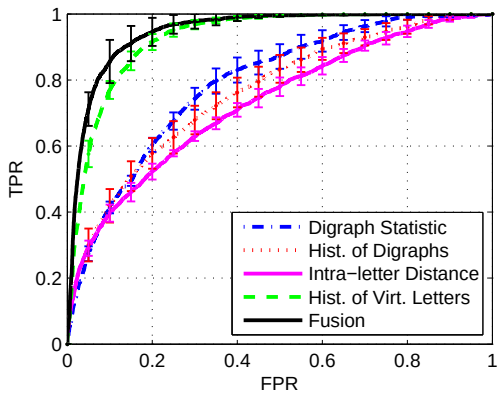


Fig. 11. ROC curves of 60-second probes in the unconstrained setting.

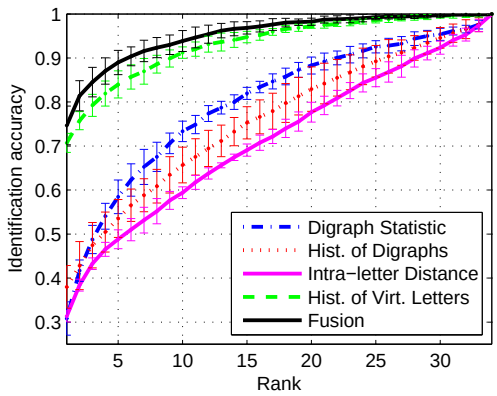


Fig. 12. CMC for closed set identification in the unconstrained setting.

desktop computer with 8 GB RAM and $3.7G$ Hz AMD Radeon processor. We implement our system in Matlab™ so the time reported is conservative, and an efficient C++ implementation would exhibit further improvements. For a 60-second probe stream, it takes approximately 20 seconds to create the feature representation with more than 98% of the time spent on keystroke segmentation. Once the features have been extracted, it takes less than 0.1 seconds to compute the score functions against a user template. Since the keystroke segmentation can be processed whenever the sound stream arrives, our system can comfortably execute in real time. Note that because of the negligible computational cost of matching to a user template, the computational efficiency of using multiple templates is almost the same as that of one template. A future work is to design an incremental way of computing the score function, similar to the online activity recognition work [27]. This is important since we would like to perform authentication in the online mode, as the sound stream is continuously received.

H. Unconstrained Free Text Setting

For keystroke sound to be used for continuous authentication, it needs to perform well during the unconstrained free text typing, which is captured in the $S2$ session of our database. In this subsection, we use the same parameters as previously tuned and evaluate the performance of using $S1$1 as the gallery and $S2$ as the probe. To create multiple probes for each user, we split the $S2$ session into 60-second probes with half overlap, which gives us in total 378 genuine samples and $12,852$ impostor samples. Figure 11 demonstrates the results for this unconstrained free text setting. We see the performance of the fused score is nearly as high as the constrained static text case with an EER of 11.7% only slightly less than 11.0% from the static text case, which is a very encouraging news considering the typical performance drop of conventional keystroke dynamics approaches when moving from static text to free text [2]. This demonstrates the potential effectiveness of keystroke sound for continuous authentication.

We attribute this minimal performance degradation to the formulation of our score functions with unconstrained free text typing in mind. The histogram of virtual letter score relies only on the *discriminative* sound produced from a collection of single keypresses. Hence, it can work well with free text typing, as long as common keypresses are observed even with limited typing duration ($\sim$60-second probe in both static and free text). In contrast, the conventional keystroke dynamics depend on the statistics of the time delay between common *pairs* of letters. Therefore, due to the variability of typed text, it takes a substantial amount of time to observe sufficiently common pairs for computing the statistics, which might cause a performance drop when the free text has the same probe length as the static text.

a) Closed Set Identification: In addition to user authentication, another application scenario of keystroke sound is the user identification in *forensic* applications. For example, closed set identification can be performed by computing and ranking the similarities between a probe stream and a set of gallery streams. Using the same data in the unconstrained free text setting, we conduct the closed set identification experiment and present the Cumulative Match Curve (CMC) in Figure 12. Keystroke sound demonstrates positive results for identification.

VI. Conclusions

In this paper, we explored the discriminative power of keystroke sound through a continuous authentication application, but there are other potential applications in forensics, security, and personalization. The proposed keystroke sound-based authentication does not interfere with normal computer use and requires minimal computational overhead. We collected a database of 50 individuals typing in both a constrained static text and unconstrained free text setting. We designed multiple approaches to compute match scores between a gallery and probe keystroke acoustic stream. Furthermore, we proposed a fusion of digraph statistics, histogram of digraphs, intra-letter distances, and histogram of virtual letters to authenticate a user. We obtained an EER of $\sim 11\%$ on a database of 50 subjects. This shows that there is promising discriminative information in the keystroke sound to be further explored. We wish to emphasize that the intent of this research study is to open up a new line of exciting research on typing-based analysis and authentication. We anticipate other interested researchers to commence applying keystroke acoustics to

various applications, ranging from continuous authentication, forensics to personalization.

There are a few limitations of the current approach which present interesting avenues for future work on this topic. First, the current database is constrained in the number of subjects, single keyboard, consistent typing environment, and single day of collection. Having a longitudinal study with many users will help identify the limitations of audio and understand the inter- and intra-class variability of keystroke sound. Second, the raw processing of the sound stream presents many opportunities for improvement. There may be better means of identifying keystrokes through supervised learning or context-sensitive thresholding. This will allow for more robust authentication in the presence of background noise typical of a normal work environment. Third, further exploration of discriminating features and classification algorithms can help improve performance. Fourth, there is no understanding of the susceptibility of the current system to attacks. Fifth, keystroke sound fits into the broader topics of keystroke dynamics and typing behavior. A real world application should integrate all available cues into a common framework to help make an authentication decision. We hope that other researchers will join us in pursuit of these research topics.

ACKNOWLEDGMENT

The authors thank the volunteers who participated in the collection of the keyboard typing database at Michigan State University. The authors also thank the associated editor and reviewers for their efforts and constructive comments.

REFERENCES

[1] D. Asonov and R. Agrawal. Keyboard acoustic emanations. In *Proc. of IEEE Symp. on Security and Privacy*, pages 3–11, May 2004.

[2] S. P. Banerjee and D. Woodard. Biometric authentication and identification using keystroke dynamics: A survey. *J. Pattern Recognition Research*, 7(1):116–139, 2012.

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

[4] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust., Speech, Signal Process.*, 28(4):357–366, Aug. 1980.

[5] H. Dozono, S. Itou, and M. Nakakuni. Comparison of the adaptive authentication systems for behavior biometrics using the variations of self organizing maps. *Int. J. Comput. Commun.*, 1(4):108–116, 2007.

[6] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 524–531, San Diego, CA, June 2005.

[7] R. Goh, L. Liu, X. Liu, and T. Chen. The CMU Face In Action (FIA) database. In *Proc. IEEE Intl. Workshop on Anal. and Modeling of Faces and Gestures held in conjunction with ICCV 2005*, pages 255–263, Beijing, China, Oct. 2005.

[8] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, Oct. 2007.

[9] A. K. Jain, K. Nandakumar, and A. Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12):2270–2285, Dec. 2005.

[10] Z. Jorgensen and T. Yu. On mouse dynamics as a behavioral biometric for authentication. In *Proc. ACM Symp. Information, Computer and Communications Security (ASIACCS)*, pages 476–482, Hong Kong, China, Mar. 2011.

[11] A. Kelly. *Cracking Passwords using Keyboard Acoustics and Language Modeling*. Master thesis, University of Edinburgh. 2010.

[12] K. S. Killourhy and R. A. Maxion. Comparing anomaly detectors for keystroke dynamics. In *Proc. of the Int. Conf. on Dependable Systems and Networks (DSN)*, pages 125–134, Lisbon, Portugal, July 2009.

[13] A. M. Martinez and A. C. Kak. PCA versus LDA. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(2):228–233, Feb. 2001.

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

[15] F. Monrose and A. Rubin. Authentication via keystroke dynamics. In *Proceedings of the 4th ACM conference on Computer and communications security*, pages 48–56, 1997.

[16] T. Mustafic, S. Camtepe, and S. Albayrak. Continuous and non-intrusive identity verification in real-time environments based on free-text keystroke dynamics. In *Proc. Int. Joint Conf. Biometrics (IJCB)*, Washington, DC, Oct. 2011.

[17] M. Nakakuni, H. Dozono, and S. Itou. Adaptive authentication system for behavior biometrics using supervised pareto self organizing maps. In *Proc. of the 10th WSEAS International Conference on Mathematical Methods, Computational Techniques and Intelligent Systems*, MAMECTIS'08, pages 277–282, Stevens Point, Wisconsin, USA, 2008.

[18] K. Nandakumar, A. Ross, and A. K. Jain. Biometric fusion: Does modeling correlation really matter? In *Proc. IEEE Conf. Biometrics: Theory, Applications, and Systems (BTAS)*, pages 271–276, Washington, DC, Sept. 2009.

[19] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss. The FERET database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing*, 16(5):295–306, Apr. 1998.

[20] J. Roth, X. Liu, and D. Metaxas. On continuous user authentication via typing behavior. *IEEE Trans. Image Process.*, 10:4611–4624, Oct. 2014.

[21] J. Roth, X. Liu, A. Ross, and D. Metaxas. Biometric authentication via keystroke sound. In *Proc. Int. Conf. Biometrics (ICB)*, Madrid, Spain, June 2013.

[22] T. Sim and R. Janakiraman. Are digraphs good for free-text keystroke dynamics? In *CVPR, Workshop on Biometrics*, 2007.

[23] J. Sivic and A. Zisserman. Video Google: a text retrieval approach to object matching in videos. In *Proc. Int. Conf. Computer Vision (ICCV)*, volume 2, pages 1470–1477, Nice, France, Oct. 2003.

[24] C. C. Tappert, S.-H. Cha, M. Villani, and R. S. Zack. A keystroke biometric system for long-text input. *Int. J. Inf. Security and Privacy*, 4(1):32–60, 2010.

[25] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 529–534, Colorado Springs, CO, June 2011.

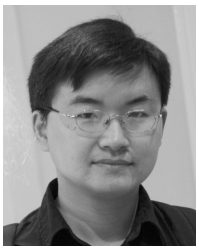
[26] K. Xi, Y. Tang, and J. Hu. Correlation keystroke verification scheme for user access control in cloud computing environment. *The Computer Journal*, 54(10):1632–1644, July 2011.

[27] Y. Zhang, X. Liu, M.-C. Chang, W. Ge, and T. Chen. Spatio-temporal phrases for activity recognition. In *Proc. European Conf. Computer Vision (ECCV)*, pages 707–721, Florence, Italy, Oct. 2012. Springer.

[28] Y. Zhong, Y. Deng, and A. K. Jain. Keystroke dynamics for user authentication. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW)*, Providence, RI, June 2012.

[29] L. Zhuang, F. Zhou, and J. D. Tygar. Keyboard acoustic emanations revisited. In *Proc. ACM Conf. Computer and Communications Security (CCS)*, pages 373–382, Alexandria, VA, Nov. 2005.

Joseph Roth is currently pursuing a Ph.D. degree with the Computer Vision Lab from the Department of Computer Science and Engineering at Michigan State University, East Lansing, MI. He received his B.S. in Computer Science from Grand Valley State University, Allendale, MI, in 2010. His research interests are computer vision and biometrics.



Xiaoming Liu is an Assistant Professor in the Department of Computer Science and Engineering at Michigan State University (MSU). He received the B.E. degree from Beijing Information Technology Institute, China and the M.E. degree from Zhejiang University, China, in 1997 and 2000 respectively, both in Computer Science, and the Ph.D. degree in Electrical and Computer Engineering from Carnegie Mellon University in 2004. Before joining MSU in Fall 2012, he was a research scientist at General Electric Global Research Center. His research areas are face recognition, biometrics, image alignment, video surveillance, computer vision and pattern recognition. He has authored more than 80 scientific publications, and has filed 22 U.S. patents. He is a member of the IEEE.

Arun Ross Arun Ross is an Associate Professor in the Department of Computer Science and Engineering at Michigan State University (MSU) and the Director of the i-PRoBe Lab. Prior to joining MSU in 2013, he was in the faculty of West Virginia University (WVU) from 2003 to 2012. He also served as the Assistant Site Director of the NSF Center for Identification Technology and Research (CITeR) between 2010 and 2012. Arun received the B.E. (Hons.) degree in Computer Science from the Birla Institute of Technology and Science, Pilani, India, and the M.S. and Ph.D. degrees in Computer Science and Engineering from Michigan State University. He is the coauthor of the textbook "Introduction to Biometrics" and the monograph "Handbook of Multibiometrics," and the co-editor of "Handbook of Biometrics". He is a recipient of the JK Aggarwal Prize, IAPR Young Biometrics Investigator Award (YBIA), the NSF CAREER Award, and was designated a Kavli Frontier Fellow by the National Academy of Sciences in 2006. He was an Associate Editor of IEEE Transactions on Information Forensics and Security (2009 - 2013), and IEEE Transactions on Image Processing (2008 - 2013). He currently serves as Area Editor of the Computer Vision and Image Understanding Journal, Associate Editor of the Image and Vision Computing Journal, Vice President of Education of the IEEE Biometrics Council, and Chair of the IAPR TC4 on Biometrics.

Dimitris Metaxas (M'93-SM'98) received the B.E. degree from the National Technical University of Athens Greece, Athens, Greece, in 1986; the M.S. degree from the University of Maryland, College Park, in 1988; and the Ph.D. degree from the University of Toronto, Toronto, ON, Canada, in 1992.

He is a Professor with the Department of Computer Science, Rutgers University, New Brunswick, NF. He is directing the Computational Biomedicine Imaging and Modeling Center. His research interests include the development of formal methods upon which computer vision, computer graphics, and medical imaging can advance synergistically.