Forgery Quality and its Implications for Behavioral Biometric Security

Lucas Ballard, Daniel Lopresti, *Senior Member, IEEE,* and Fabian Monrose

***Abstract*— Biometric security is a topic of rapidly growing importance in the areas of user authentication and cryptographic key generation. In this paper, we describe our steps toward developing evaluation methodologies for behavioral biometrics that take into account threat models which have been largely ignored. We argue that the pervasive assumption that forgers are minimally motivated (or, even worse, naïve) is too optimistic and even dangerous. Taking handwriting as a case in point, we show through a series of experiments that some users are significantly better forgers than others, that such forgers can be trained in a relatively straightforward fashion to pose an even greater threat, that certain users are easy targets for forgers, and that most humans are a relatively poor judge of handwriting authenticity and hence their unaided instincts cannot be trusted. Additionally, to overcome current labor-intensive hurdles in performing more accurate assessments of system security, we present a *generative attack* model based on concatenative synthesis that can provide a rapid indication of the security afforded by the system. We show that our generative attacks match or exceed the effectiveness of forgeries rendered by the skilled humans we have encountered.**

I. Introduction

The security of many systems relies on obtaining human input that is assumed to be not readily reproducible by an attacker. Passwords are a common example, though the assumption that these are not reproducible is suspect. Indeed, memorable passwords are generally easy for an adversary to guess [5]. Biometrics is an alternative form of input that is believed to address the contention between memorability and security. This belief has led to the incorporation of biometrics into security applications such as authentication [24] and cryptographic key generation [23], [32].

Biometrics may be divided into two broad categories. *Physiological* biometrics measure biological traits, for instance, characteristics of a fingerprint or iris. *Behavioral* biometrics measure how users perform certain actions, such as speaking or writing. Although physiological biometrics have enjoyed more attention than behavioral biometrics, and have consequently become more integrated into commercial products, behavioral biometrics exhibit several qualities that make them attractive for security applications. For instance, whereas an adversary can passively extract physiological biometrics (i.e., by lifting a fingerprint from a keyboard), behavioral biometrics do not lend themselves as easily to surreptitious capture as they require a user to consciously perform an action (i.e., speaking a specific phrase). Additionally, while physiological biometrics cannot change, behavioral biometrics naturally change with the action that is performed. This property is useful for security applications such as key generation, where key compromise necessitates the creation of a new key.

Regardless of the type of biometric, designers generally perform empirical evaluations to justify the assumption that a system will withstand attacks. The evaluation usually follows a standard model: enroll some number of users by collecting training samples. At a later time, test the rate at which users' attempts to recreate the biometric to within a predetermined tolerance fail. This failure rate is denoted as the False Reject Rate (FRR). Additionally, evaluation involves assessing the rate at which one user's input (i.e., an impostor) is able to fool the system when presented as coming from another user (i.e., the target). This yields the False Accept Rate (FAR) for the system under consideration. Typically, one uses the equal error rate (EER), or the point at which the FRR and the FAR are equal, to describe the accuracy of a biometric system.

Clearly the FAR and EER are a function of the quality of the collected forgeries. For an evaluation to be meaningful, the forgeries must be representative of those that the system would expect to see during actual operation. For physiological biometrics—which are not based on human actions—a reasonable approach is to use samples from one user as forgeries for another. As an example, one might try to match one user's fingerprint to another's template.

Providing a reasonable forgery for behavioral biometrics is not as straightforward. Researchers predominantly use two forgery styles to estimate the FAR of a behavioral biometric system. "Naïve" (also called "random," "zero-effort," or "accidental") forgeries are created by using one user's samples as forgeries for another user. Naïve forgery has roots in forgeries of physiological biometric systems (where its use is less suspect) and is easy to perform as it requires only enrollment samples. However, naïve forgeries may not provide an adequate estimate of security; in some instances, they are not even based on writing, speaking, or typing the same passphrase as the target user.

"Skilled" forgeries are created by users who use information about the targeted input to create a replica. Skilled forgeries are generally preferred to naïve forgeries as they provide a more realistic view of security. However, in this work, we provide what we believe to be the first analysis that concretely demonstrates that even so-called skilled forgeries are *not* indicative of realistic threats faced by biometric systems. Thus, the evaluation of behavioral biometrics under such weak security assumptions can be misleading.

Manuscript received May 19, 2006; revised February 22, 2007 and March 5, 2007.

Lucas Ballard and Fabian Monrose are with the Department of Computer Science, Johns Hopkins University, Baltimore, MD.

Daniel Lopresti is with the Department of Computer Science & Engineering, Lehigh University, Bethlehem, PA.

Misunderstanding forger capability is especially dangerous when behavioral biometrics are adopted for sensitive applications such as authentication or cryptographic key generation. Underestimating the ability of an adversary to bypass an authentication mechanism could lead users to feel a disproportionally high level of trust, and consequently to forgo extra steps that they may have otherwise taken to secure sensitive information. This problem is amplified when biometrics are used to create cryptographic keys, which might be used to encrypt sensitive data over extended periods of time. The long term nature of the encryption introduces the possibility for adversaries to search for the target key. For instance, adversaries could forge the biometric input to within some tolerance, and then perturb the resulting key to enumerate likely possibilities. In such a situation, any overestimate of forger capability could be devastating. Thus, it is essential for researchers to truly understand the effort that an adversary must exert to forge a behavioral biometric.

In what follows, we provide an in-depth study that emphasizes the disconnect between standard evaluation practices and realistic adversaries. While our belief is that our results are generally applicable, and our ideas may be extended to any behavioral biometric, the detail required by our study necessitates an comprehensive analysis of a specific modality. As a case in point, we focus on handwriting. Through a series of experiments, we show that some users are significantly better forgers than others (so-called “wolves” in the jargon for a hypothetical menagerie of users [2]), that such forgers can be trained in a relatively straightforward fashion to pose an even greater threat, that certain users are easy targets for forgers (i.e., “lambs”), and that most humans are a relatively poor judge of handwriting authenticity (hence assertions that “our forgers looked like they were doing a good job” are suspect). We conclude with our proposal for a new evaluation paradigm for biometric security based on the concept of generative models for the behavior in question.

II. Handwriting Biometrics

Research on user authentication via handwriting has had a long, rich history, with hundreds of papers written on the topic. The majority of this work to date has focused on the problem of signature verification [28]. Signatures have some well known advantages: they are a natural and familiar way of confirming identity, have already achieved acceptance for legal purposes, and their capture is less invasive than most other biometric schemes [4]. While each individual has only one true signature—a notable limitation—handwriting in general contains numerous idiosyncrasies that might allow a writer to be identified.

In considering the mathematical features that can be extracted from the incoming signal to perform authentication, it is important to distinguish between two different classes of inputs. Data captured by sampling the position of a stylus tip over time on a digitizing tablet or pen computer are referred to as *online* handwriting, whereas inputs presented in the form of a 2-D bitmap (e.g., scanned off of a piece of paper) are referred to as *offline* handwriting. To avoid confusion with the traditional attack models in the security community, later on in this paper we shall eschew that terminology and refer to the former as covering both temporal and spatial information, whereas the latter only covers spatial information. Features extracted from offline handwriting samples include bounding boxes and aspect ratios, stroke densities in a particular region, curvature measurements, etc. In the online case, these features are also available and, in addition, timing and stroke order information that allows the computation of pen-tip velocities, accelerations, etc. Studies on signature verification and the related topic of handwriting recognition often make use of 50 or more features and, indeed, feature selection is itself a topic for research. The features we use in our own work are representative of those commonly reported in the field [7], [16], [21], [34].

In the literature, performance figures (i.e., EER) typically range from 2% to 10% (or higher), but are difficult to compare directly as the sample sizes are often small and test conditions dissimilar [3]. Unfortunately, forgers are rarely employed in such studies and, when they are, there is usually no indication of their proficiency. Attempts to model attackers with a minimal degree of knowledge have involved showing a static image of the target signature and asking the impostor to try to recreate the dynamics [25]. The only concerted attempt we are aware of, previous to our own, to provide a tool for training forgers to explore the limits of their abilities is the work by Zoebisch and Vielhauer [33]. In a small preliminary study involving four users, they found that showing an image of the target signature increased false accepts, and showing a dynamic replay doubled the susceptibility to forgeries yet again. However, since the verification algorithm used was simplistic and they do not report false reject rates, it is difficult to draw more general conclusions.

To overcome the “one-signature-per-user” (and hence, one key) restriction, we employ more general passphrases in our research. While signatures are likely to be more user-specific than arbitrary handwriting, results from the field of forensic analysis demonstrate that writer identification from a relatively small sample set is feasible [9]. Indeed, since this field focuses on handwriting extracted from scanned page images, the problem we face is less challenging in some sense since we have access to dynamic features in addition to static. Another concern, user habituation [3], is addressed by giving each test subject enough time to become comfortable with the experimental set-up and requiring practice writing before the real samples are collected. Still, this is an issue and the repeatability of non-signature passphrases is a topic for future research.

III. Experimental Design

We collected data over a two month period to analyze six different forgery styles. We consider three standard evaluation metrics: *naïve*, *static*, and *dynamic* forgeries[1] [10], [15], [30], as well as three metrics that will provide a more realistic

[1] Although the biometric literature often refers to static or dynamic forgeries as skilled forgeries, here we make a distinction. In fact, only a subset of forgers who are presented with static or dynamic information may indeed be “skilled”.

definition of security: *naïve**, *trained*, and *generative*. Naïve forgeries are not really forgeries in the traditional sense; they are measured by authenticating one user's natural writing samples of a passphrase against another user's template for the same passphrase. Static forgeries are created by humans after seeing static renderings of a target user's passphrase. Dynamic forgeries are created by humans after seeing real-time renderings of a target user's passphrase. Naïve* forgeries are similar to naïve forgeries except that only writings from users of a similar style are authenticated against a target user's template. Trained forgeries are generated by humans under certain conditions, which will be described in greater detail later. Lastly, generative forgeries exploit information about a target user to algorithmically create forgeries. Such information may include samples of the user's writing from a different context or general population statistics.

A. Data Collection

Our results are based on 9,026 handwriting samples collected on digitized pen tablet computers from 47 users during several rounds. We grouped users into three categories according to their writing style: "block" writers tend to lift their pen between letters, "cursive" writers tend to connect every letter, and "mixed" writers connected some letters, but not others. The determination of whether or not users connected letters was made by the authors based on static writing samples. Our data set contains 10 block writers, 17 mixed writers, and 20 cursive writers.

We used NEC VersaLite Pad and HP Compaq TC1100 tablets as our writing platforms. To ensure that the participants were well motivated and provided writing samples reflective of their natural writing (as well as forgery attempts indicative of their innate abilities), several incentives were awarded for the most consistent writers, the best/most dedicated forgers, etc. Additionally, before any data collection, users were asked to write several phrases to become comfortable with the writing device [3]. To create a strong underlying representative system, users were given instructions to write as naturally (and consistently) as possible.

During enrollment users provided twenty renderings of five different phrases consisting of two-word oxymorons ("crisis management," "graphic language," "least favorite," "perfect misfit," and "solo concert"). We chose these phrases as they were easy to remember (and therefore, can be written naturally) and could be considered of reasonable length. Signatures were not used due to privacy concerns and strict restrictions on research involving human subjects. More importantly, in the context of key generation, signatures are not a good choice for a handwriting biometric as the compromise of keying material could prevent a user from using the system thereafter. This data was collected across two rounds, `round II` starting approximately two weeks after `round I` (see Table I for a breakdown of the number of enrollment and forgery samples collected in each round). Enrollment samples were used to create templates for authentication, as well as naïve and naïve* forgeries.

In `round I` users also provided 65 additional writing samples to create our "parallel corpus," which would later be used to create generative forgeries. This set was restricted so that it did not contain any of the five phrases from the enrollment data set, yet provided coverage of the phrases at the bigram level. Users were asked to write one instance of each phrase as naturally as possible.

We collected static and dynamic forgeries in `round II`. Users were asked to forge representative samples (based on writing style, handedness of the original writer, and gender) from `round I` to create two sets of 17 forgeries. First, users were required to forge samples after seeing *only* a static representation. Users were then asked to forge the same phrases again, after seeing a real-time rendering. Users were instructed to use the real-time presentation to improve their rendering of the spatial features (for example, to distinguish between one continuous stroke versus two overlapping strokes) and to replicate the temporal features of the writing.

Lastly, in `round III` we selected nine users from `round II` to provide our trained forgeries. These users exhibited a natural tendency to produce better forgeries than the average user in our study (although we did not include all of the best forgers). This group consisted of three "skilled" (but untrained) forgers for each writing style, when evaluated using the authentication system to be described in Section III-C and Section III-D. Each skilled forger was asked to forge writing from the style which they exhibited an innate ability to replicate and was provided with a general overview and examples of the types of features that handwriting systems typically capture. As we were trying to examine (and develop) truly skilled adversaries, our forgers were asked to forge 15 writing samples from their specified writing style, with 60% of the samples coming from the weakest 10 targets, and the other 40% chosen at random. (Interestingly, the accuracy of our trained forgers against this mix of targets and against the entire population differed only a statistically insignificantly amount.) From this point on, these forgers will be referred to as "trained" forgers. See Figure 1 for example trained forgeries. We believe that the selection of the naturally skilled forgers, the additional training, and the selection of specific targets produced adversaries who reflect realistic threats to biometric security.

The experimental setup for trained forgers was as follows. First, a real-time reproduction of the target sample is displayed (at the top half of the tablet) and the forger is allowed to attempt forgeries (on the bottom half) with the option of saving the attempts she liked. She can also select and replay her forgeries and compare them to the target. In this way, she is able to fine-tune her attempts by comparing the two writing samples. Next, she selects the forgery she believes to be her best attempt, and proceeds to the next target.

B. Accounting for Hardware Variability

Extra care was taken when preparing our tools for data collection. In particular, we encountered two difficulties: (1) the two platforms sampled stylus inputs at different rates and (2) replay of real-time forgeries could be inconsistent and slow. The first issue is problematic as it leads to extra errors if evaluation samples are not collected on the same tablet as

	Enrollment	Parallel Corpus	Naïve	Naïve*	Static	Dynamic	Trained
	round I/II	round I	round I/II		round II		round III
Block	1000	650	4600	180	122	122	47
Mixed	1700	1105	4600	360	218	219	46
Cursive	2000	1300	4600	380	227	227	43

TABLE I
NUMBER OF SAMPLES COLLECTED FOR ENROLLMENT, THE PARALLEL CORPUS, AND FORGERIES.

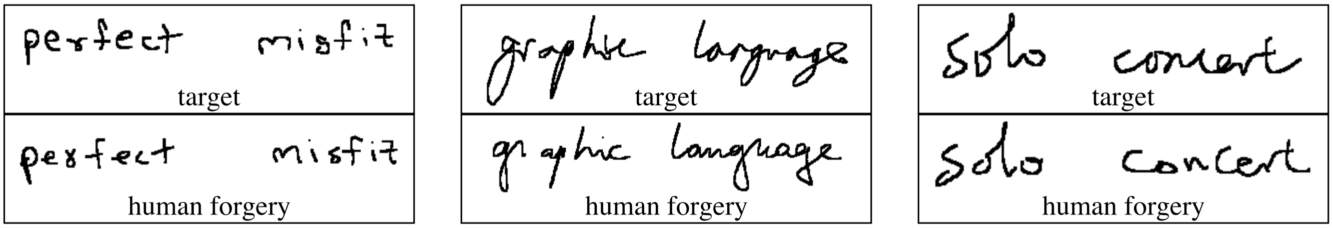

Fig. 1. Examples of block, mixed, and cursive forgeries provided by our trained forgers.

the enrollment samples. The second issue is problematic as forgers cannot be expected to accurately replicate temporal features if they are not presented with realistic representations of the handwriting. We address these two issues as follows.

Whereas the NEC tablet sampled 66.5% of the stylus inputs at a rate of 5-8 ms, the HP tablet sampled 88.5% of the input at 3-5 ms or 10-12 ms. To account for this variability, we re-sampled each sample at a rate of 8 ms before computing any feature. Our re-sampling approach is straightforward and closely related to widely accepted techniques [29]. The results presented in Section V indicate that the forging accuracy for inter-tablet experiments versus intra-tablet experiments were statistically insignificant.

We attributed slow playback of handwriting samples to underlying activities in the operating system. These unpredictable and bursty overheads increased the delays that naturally occur between rendering each point in the handwriting. To account for this, we designed a simple error correcting algorithm. Assume that the system has rendered point p_i and should wait d_i ms before rendering p_{i+1}. Let t_i be the amount of time that should have elapsed while rendering p_{i-2}, p_{i-1}, and p_i and t'_i be the actual elapsed time. If $t'_i > 2t_i$, we reduce the next delay to $\max(0, d_i - (t'_i - t_i))$. The success of our forgers in replicating temporal features (see Section V) shows that discrepancies that appeared in the rendering were indeed insignificant.

C. Authentication Algorithm

We loosely adapted the system presented in [34] for generation of "biometric hashes" to measure the FAR for each of the six forgery styles. We selected this technique as the basis for our evaluation since it does not use any additional cryptographic components (e.g., [14]), that might pose an additional hurdle to forgers. To create an accurate forgery for a biometric hash, one must only replicate the features, which were selected to be representative of the state of the art [7], [16], [21], [34].

For completeness, we briefly describe relevant aspects of the system; for a more detailed description see [34]. A user writes a passphrase on an electronic tablet to input a sample to the system. The tablet extracts a set of signals from the handwriting. The discrete signals $x(t)$ and $y(t)$ specify the location of the pen on the writing surface at time t, and the binary signal $p(t)$ specifies whether the pen is up or down at time t. The tablet then computes a set of n statistical features $(f_1, \ldots, f_n)$ over these signals. These features comprise the actual input to the biometric hash algorithm.

During an enrollment phase, each legitimate user writes a passphrase a pre-specified number (m) of times. Let $f_{i,1}, \ldots, f_{i,n}$ denote the feature values for sample i. Using the feature values from each user and passphrase, the system computes a global set of tolerance values ($T = \{\epsilon_1, \ldots, \epsilon_n\}$) to be used to account for natural human variation [34]. Once the m readings have been captured, a biometric template is generated for each user and passphrase as follows: Let $\ell'_j = \min_{i \in [1,m]} f_{i,j}$, $h'_j = \max_{i \in [1,m]} f_{i,j}$, and $\Delta_j = h'_j - \ell'_j + 1$. Set $\ell_j = \ell'_j - \Delta_j \epsilon_j$, and $h_j = h'_j + \Delta_j \epsilon_j$. The resulting template is an $n \times 2$ matrix of values $\{\{\ell_1, h_1\}, \ldots, \{\ell_n, h_n\}\}$.

Later, when a user provides a sample with feature values $f_1, \ldots, f_n$, the system checks whether $f_j \in [\ell_j, h_j]$ for each feature f_j. Each $f_j \notin [\ell_j, h_j]$ is deemed an error, and depending on the threshold of errors tolerated by the system, the attempt is either accepted or denied. We note that as defined here, templates are insecure because they leak information about a user's feature values. We omit discussion of securely representing biometric templates as this is not a primary concern of this research.

D. Feature Analysis

The security of any biometric system is directly related to the quality of the underlying features. A detailed analysis of proposed features for handwriting verification is presented in [34], although we argue that the security model of that work sufficiently differs from our own and so a new feature-evaluation metric was required for our examinations. In that work, the quality of a feature was measured by the deviation of the feature and entropy of the feature across the population. For our purposes, these evaluation metrics are not ideal: we

Feature (f)	$Q(f)$	**Feature (f)**	$Q(f)$	**Feature (f)**	$Q(f)$
Spatial Features					
Pen-down distance [7]	0.81	Writing height [7], [34]	0.65	Lower zone [21]	0.62
Median θ [21]	0.71	Pen-up distance	0.64	X-Area [34]	0.62
Vert. end dist. [16]	0.67	# of strokes [34]	0.63	Loop area [7]	0.61
Y-Area [34]	0.65	# of extrema [34]	0.62	Upper zone [21]	0.61
Writing width [7], [34]	0.65	Loop y centroid [7]	0.62	Horiz. end dist [16]	0.60
Temporal Features					
Writing time [16]	0.87	Time of max v_x [16]	0.78	Pen up/down ratio [16]	0.71
# of times $v_x = 0$ [16]	0.86	Inv. Mom. 21 [7]	0.76	Time of max θ	0.70
# of times $v_y = 0$ [16]	0.85	Inv. Mom. 12 [7]	0.75	Duration $v_y < 0$ [16]	0.70
Inv. Mom. 00 [7]	0.85	Median pen velocity [16]	0.74	Duration $v_x < 0$ [16]	0.69
Inv. Mom. 10 [7]	0.82	Duration $v_y > 0$ [16]	0.73	Time of min v_x [16]	0.69
Inv. Mom. 01 [7]	0.79	Duration $v_x > 0$ [16]	0.72	Time of min v_y [16]	0.68
Inv. Mom. 11 [7]	0.78	Time of max vel. [16]	0.72	Time of max v_y [16]	0.68

TABLE II

FEATURES USED TO EVALUATE FORGERS. θ IS THE ANGLE BETWEEN POINTS, v, v_x, v_y ARE OVERALL, HORIZONTAL, AND VERTICAL VELOCITIES.

are not only concerned with the entropy of each feature, but rather how difficult the feature is to *forge*[2].

As our main goal is to highlight limitations in current practices, it is imperative that we evaluate a robust and usable system based on a strong feature set. To this end, we implemented 144 state of the art features [7], [16], [26], [34] and evaluated each with a quality function. For each feature f, let r_f and a_f be the proportion of times that legitimate users and forgers with access to dynamic information fail to replicate f. Then, our quality function is defined as $Q(f) = (a_f - r_f + 1)/2$, and so the range of Q is $[0, 1]$. Intuitively, features with a quality score of 0 are completely useless—they are *never* reliably reproduced by original users ($r_f = 1$) and are *always* reproduced by forgers ($a_f = 0$). On the other hand, features with scores closer to 1 are highly desirable when implementing biometric authentication systems.

For our evaluation, we divided our feature set into two groups covering the temporal and spatial features, and ordered each according to the quality score. We then chose the top 40 from each group, and disregarded any with a FRR greater than 10%. Finally, we discounted any features that could be inferred from others. This analysis resulted in what we deem the 36 best features—15 spatial and 21 temporal—described in Table II.

IV. EVALUATION METHODOLOGY

This section presents the results for the five evaluation metrics that use forgeries generated by humans. Before computing the FRR and FAR, we removed outliers from the enrollment samples as follows. We assume that each feature is independently distributed. For each user, we removed all samples that have more than $\delta = 3$ features that fell outside $k = 2$ standard deviations from that user's mean feature value. The parameters δ and k were empirically derived; increasing δ or k beyond this point did not significantly affect which samples were classified as outliers. We also excluded users (the so-called "Goats" [2]) for which we removed more than 25% of the samples as outliers and classified such users as "Failing to Enroll" [20]. The FTE rate was $\approx 8.7\%$. After combining this with outlier removal, we still had access to 79.2% of the original data set.

To compute the FRR and FAR we use the system described in Section III-C with the 36 best features from Section III-D. The FRR is computed as follows: we repeatedly randomly partition a user's m samples into two groups and use the first group (of size $\frac{3m}{4}$) to build a template and authenticate the samples in the second group (of size $\frac{m}{4}$) against the template. Depending on how many outliers were removed for each user, $15 \leq m \leq 20$. To compute the FAR we use all of the user's samples to generate a template and then authenticate the forgeries against this template.

V. HUMAN EVALUATION

Our experiments were designed to illustrate the discrepancy in perceived security when considering traditional forgery paradigms and a more stringent, but more realistic, security model. In particular, we assume that at the very minimum, the adversary (1) tries to impersonate users who have a writing style that the forger has a natural ability to replicate, (2) has knowledge of how biometric authentication systems operate, and (3) has a vested interest in accessing the system, and therefore is willing to devote significant effort towards these ends.

Figure 2 presents ROC curves for forgeries from impersonators with varying levels of knowledge. The plot denoted FAR-naïve depicts results for the traditional case of naïve forgeries widely used in the literature [10], [15], [30]. Therefore, in addition to ignoring the target writer's attributes, this classification makes no differentiation based on the forger's or victim's style of writing, and so may include, for example, block writers "forging" cursive writers. Arguably, such forgeries may be inferior to a less standard (but more reasonable) type of naïve classification (FAR-naïve*) where one only attempts to authenticate samples from writers of similar styles.

The FAR-static plot shows the success rate of forgers who receive access to only a static rendering of the passphrase. By contrast, FAR-dynamic forgeries are produced by humans after seeing (possibly many) real-time renderings of the target phrase. One can easily consider this a realistic threat if we

[2] It is interesting to note that despite the different metrics, there was a high correlation between our stronger features and those in [34].

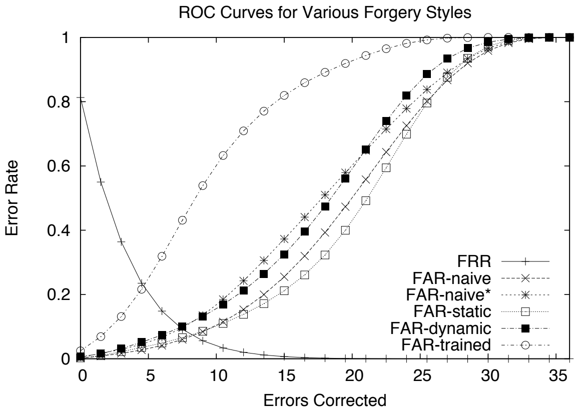

Fig. 2. ROC curves for human forgers. Naïve* and dynamic forgeries exhibit similar patterns, but both are inferior to trained forgeries.

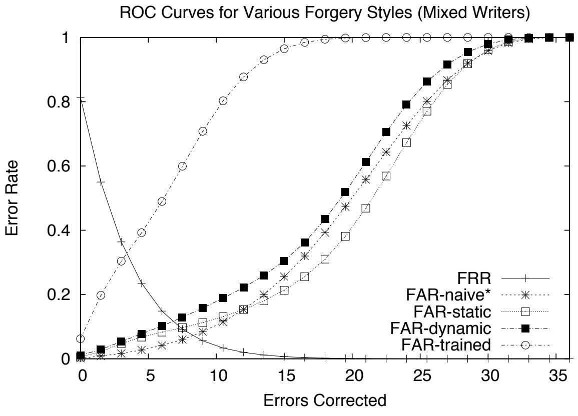

Fig. 3. ROC curves for mixed writers. This group appeared the easiest to forge by the users in our study.

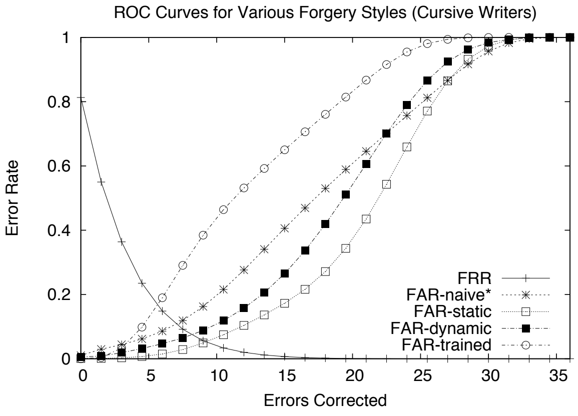

Fig. 4. ROC curves for cursive writers. This group appeared the most difficult to forge by users in our study.

assume that a motivated adversary may capture the writing on camera, or more likely, may have access to data written electronically in another context. Lastly, FAR-trained presents the resulting success rate of forgeries derived under our forgery model which captures a more worthy opponent—one who has natural skill and who has undergone some level of training. When classified by writing style, these trained forgers were very successful against block and mixed writers (see Figure 3), and had the most difficulty with cursive writers (Figure 4).

Intuitively, one would expect that forgers with access to dynamic and/or static representations of the target writing should be able to outperform naïve* forgeries. This is not necessarily the case, as we see in Figure 2 that at some points, the naïve* forgeries do better than the forgeries generated by forgers who have access to static and/or dynamic information. This is primarily due to the fact that the naïve* classification reflects users' normal writing (as there is really no forgery attempt here). The natural tendencies exhibited in such writings appear to produce better "forgeries" than that of static or dynamic forgers (beyond some point), who may suffer from unnatural writing characteristics as a result of focusing on the act of forging.

One of the most striking results depicted in the figures is the significant discrepancy in the FAR between standard evaluation methodologies and that of the trained forgers captured under our strengthened model. For instance, the EER for this system under FAR-trained forgeries is approximately 20.6% at four error corrections (see Table III)[3]. However, for the more traditional dynamic, static and naïve forgeries, one would arrive at EERs of 7.9%, 6.0%, and 5.5%. These results are indeed inline with the current state of the art [10], [15], [30]. Even worse, under the most widely used form of adversary considered in the literature (i.e., naïve) we see almost a four-fold over-estimate of equal error rate.

	Naïve*	Static	Dynamic	Trained
Block	7.6	7.2	8.5	21.3
Mixed	5.6	8.2	9.0	33.1
Cursive	8.3	3.6	5.5	13.5
Overall	7.4	6.0	7.9	20.6

TABLE III
EQUAL ERROR RATES FOR DIFFERENT FORGERY STYLES.

Figure 5 provides assurance that the increase in forgery quality is not simply a function of selecting naturally skilled individuals from our dynamic forgers to act as trained forgers. The graph shows the improvement of FAR for these forgers across rounds II and III. Improvement is significant, especially for those who focused on mixed and block writers. Notice that at the EER (at seven errors) induced by forgers with access to dynamic information from round II (Figure 2), our trained cursive, block, and mixed forgers improved their FAR by 0.18, 0.34, and 0.47, respectively. This change results from less than two hours of training and effort, which is likely much less than what would be exerted by a dedicated forger.

While it is interesting to note the drastic increase in forger improvement, the manner in which the forgers improved is also intriguing. While the trained forgers were able to better

[3] It is important to note that the trained forgers faced a different distribution of "easy" targets in Round II and in Round III. We did this to analyze the system at its weakest link. However, after normalizing the results so that both rounds had the same makeup of "easy" targets, the EER only changes marginally from 20.6% to 20.0% at four errors corrected.

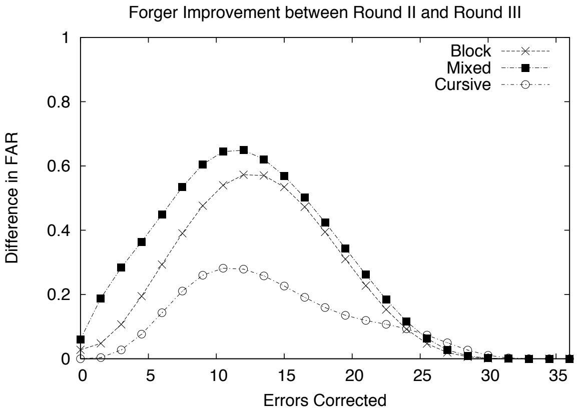

Fig. 5. The impact of training on the FAR exhibited by our trained forgers.

replicate each feature, the improvement of a particular subset (see Table IV) was especially noteworthy.

With the exception of writing width and pen-down distance, each of the features are intuitively related to being able to consistently replicate pen-tip velocity across the entire writing sample. A closer analysis of these features also provides insight as to why mixed forgers improved more dramatically than block or cursive forgers. For instance, the success of the mixed forgers in capturing Invariant Moments 12 and 21, writing width, and the number of times that $v_x = 0$, improved by an average of 78.3%. By contrast, cursive forgers only improved by an average of 45.0%.

A disturbing characteristic of these features is that they were ranked among the highest by our quality metric (see Table II). This correlation could be due to the fact that these were the features that were originally the most difficult to forge and therefore had the most room for improvement. Nonetheless, a closer examination of this impact is warranted, but is left for future work.

Feature (f)	a_f (rnd II)	a_f (rnd III)	% Improvement
# of times $v_x = 0$	.823	.284	65.4
Writing width	.386	.147	61.8
Inv. Mom. 21	.612	.240	60.8
Inv. Mom. 12	.596	.246	58.7
Time of min v_y	.437	.190	56.5
Inv. Mom. 11	.659	.289	56.1
Pen-down distance	.713	.314	56.0
# of extrema	.318	.149	53.2
Inv. Mom. 10	.744	.359	53.1

TABLE IV

FEATURES FOR WHICH FAR WAS MOST INCREASED BY TRAINING. a_f IS THE PROPORTION OF TIMES FEATURE f IS MISSED BY FORGERS.

VI. AN ALTERNATIVE PERSPECTIVE: FORGERY DETECTION BY HUMANS

The results from Section V demonstrate that our methodologies did indeed improve the talents of our forgers with respect to an authentication algorithm. To provide additional evidence that forgers can improve their performance through training in a more intuitive sense, we conducted an alternative evaluation aimed at understanding the proficiency of laypersons in our study (herein referred to as *human judges*) at detecting forgeries. To the best of our knowledge, there were no professional document examiners in our preliminary study.

Authenticity	Training	Style	% of Samples
Forgery	Trained	Dynamic	25
Forgery	Trained	Static	0
Forgery	Untrained	Dynamic	12.5
Forgery	Untrained	Static	12.5
Authentic	NA	Dynamic	25
Authentic	NA	Static	25

TABLE V

BREAKDOWN OF THE SAMPLES CLASSIFIED BY OUR HUMAN JUDGES.

We caution the reader that the proficiency of professional document examiners compared to that of laypersons in detection forgeries remains a controversial topic (see for example, [6], [31]). Indeed, while recent studies [11], [12] seem to indicate that a well-trained subset of the population can perform significantly better than chance at this task, these results are still being openly debated. For that reason, we primarily use our analysis of human judges as yet another indication of the importance of using strong forgers for evaluation purposes.

To measure the impact of forger training on a layperson's ability to distinguish forgeries, we performed the following experiment. In a given round, judges were presented with three writing samples of the same passphrase: the first two originated from the same user, and the third was selected as a forgery with 50% probability. The judge's task was to decide whether or not all of the samples originated from the same user. In all, 24 judges were asked to determine the authenticity of 20 samples. The tests were not timed.

Half of the forgeries originated from untrained forgers and half originated from trained forgers. For each round, the judge saw either static or dynamic renderings. If the forger was only presented with static information, then the judge saw a static rendering of the samples as well; otherwise, the judge saw dynamic renderings which she could replay at will. Thus, forgeries generated by trained forgers were always rendered in real-time. See Table V for the precise distribution of the samples classified by our human judges.

We note that since judges saw only two samples from the target writer we cannot directly compare the judges' accuracy to that of the authentication algorithm, which was trained on 15-20 samples. However, we can rely on these results as an indication of forger improvement, and as a preliminary indication of whether humans are good at detecting forgeries.

Figure 6 shows the overall accuracy of the judges. The error rate is simply the proportion of times a judge misclassifies a sample, whether it be a false accept or false reject. At first glance, it appears that different types of forgeries have little impact on the error rates of judges. However, as we show shortly, when we separate the errors as false rejects (Figure 7) and false accepts (Figure 8), this is not necessarily the case. The primary observation from Figure 6 is that untrained human judges seem to perform poorly at detecting forgeries. For instance, for each of the forgery styles, approximately 50% of the judges had a classification rate of less than 75%.

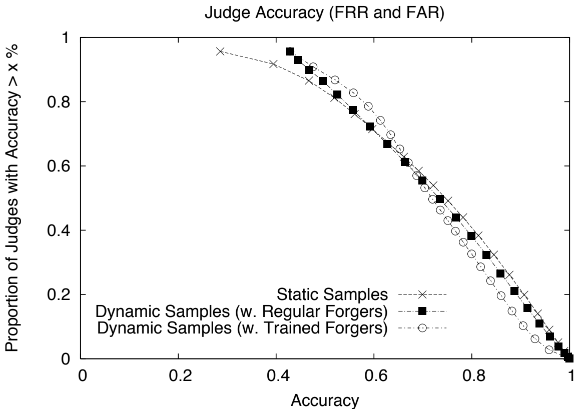

Fig. 6. Proportion of judges with overall accuracy greater than $x\%$.

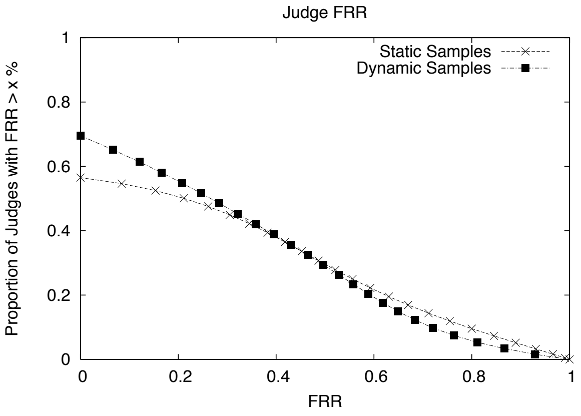

Fig. 7. Proportion of judges with a FRR greater than $x\%$.

The average FRR (Figure 7) was 35.5% and 35.4% when observing static and real-time renderings, respectively. Interestingly, this seems to imply that, for the most part, the addition of dynamic information did not improve a judge's ability to accurately identify true samples. However, the increase in the FAR for normal versus trained forgers is significant (Figure 8); the false-accept rate of the trained forgeries rose above 0 for almost 75% of the judges. By contrast, the FAR of the untrained forgers only was greater than 0 for 20% of the judges. Additionally, 50% of the judges exhibited a FAR $>$ 20% against the trained forgers, whereas only 17% of these judges had similar rates against untrained forgers. Clearly, our trained forgers were not only more talented in replicating the features from Section III-D, but in also replicating the idiosyncrasies that a normal human might use to differentiate writing from different users. Note that there is an anomaly in Figure 8 in that outliers at the tails of the distributions suggest that there were regular forgers who fooled judges more often than trained forgers. In particular, one judge was fooled by 80% of the untrained forgeries.

VII. Generative Evaluation

Finding and training "skilled" forgers is a time, and resource, consuming endeavor. To confront the obstacles posed

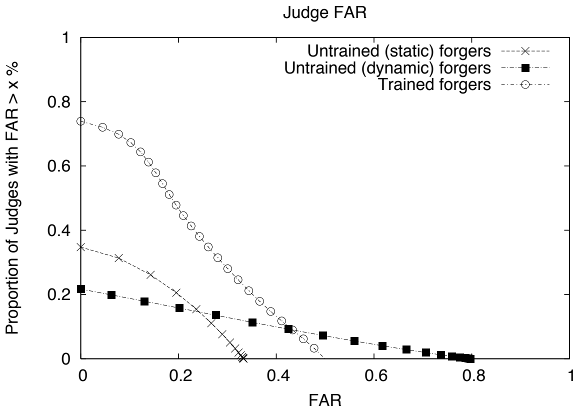

Fig. 8. Proportion of judges with a FAR greater than $x\%$.

by wide-scale data collection and training of good impersonators, we explore an automated approach using generative models as a supplementary technique for evaluating behavioral biometrics. We investigate whether an automated approach, using limited writing samples from the target, could match the false accept rates of our trained forgers in Section V.

Our proposed algorithm is *generative* in nature; it uses limited information from the target user, as well as general population statistics, and intuitive rules of thumb to create a best-effort guess at a target user's biometric. As such, it is designed to replicate only a target user's passphrase. However, as our results will demonstrate, this simple approach is surprisingly effective and underscores the importance of considering generative algorithms both as useful techniques to evaluate the security of a system, as well as a new threat model that researchers should consider when designing their systems.

For the remaining discussion we explore a set of threats that stem from generative attacks which assume knowledge that spans the following spectrum:

I. *General population statistics*: Gleaned, for example, via the open sharing of test data sets by the research community, or by recruiting colleagues to provide samples.

II. *Statistics specific to a demographic of the targeted user*: In the case of handwriting, we assume the attacker can extract statistics from a corpus collected from other users of a similar writing style.

III. *Data gathered from the targeted user*: Excluding direct capture of the secret itself, one can imagine the attacker capturing copies of a user's handwriting, either through discarded documents or by stealing a PDA.

To make this approach feasible, we also explore the impact of these varying threats. A key issue that we consider is the amount of recordings one needs to make these scenarios viable attack vectors. As we show later, the amount of data required may be surprisingly small for the case of authentication systems based on handwriting dynamics.

To synthesize handwriting we assemble a collection of basic units (n-grams) that can be combined in a concatenative fashion to mimic authentic handwriting. We do not make use

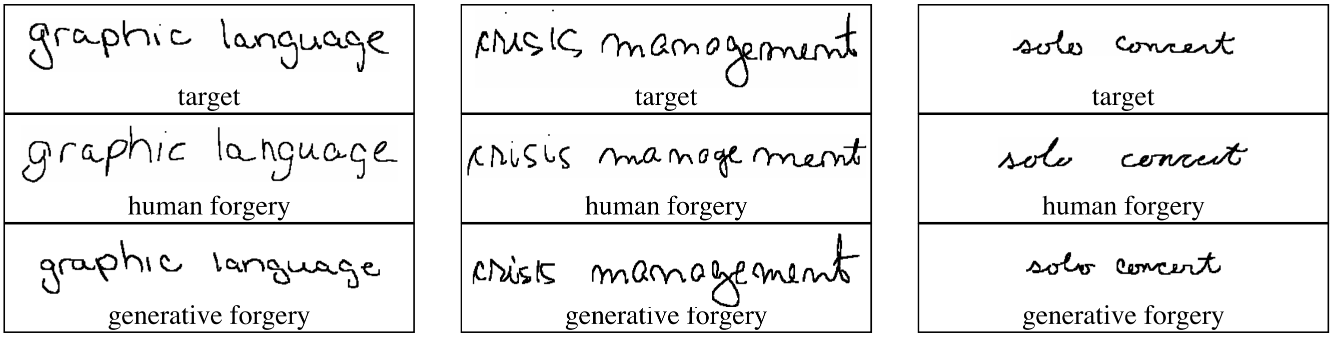

Fig. 9. Example generative forgeries against block, mixed and cursive writers. For each box, the second rendering is a human-generated forgery of the first, and the third is created by our generative algorithm.

of an underlying model of human physiology, rather, creation of the writing sample is accomplished by choosing appropriate n-grams from an inventory that may cover writing from the target user (scenario III above) as well as representative writings by other members of the population at large (scenarios I and II). Our technique expands upon earlier rudimentary work [19], and is similar in flavor to approaches taken to generate synthesized speech [23] and for text-to-handwriting conversion [8].

The first step of the forging process is to obtain a small set of samples from the user (the parallel corpus), and a set of samples from other writers of the same writing style (to derive population statistics). As is the case with traditional computations of EER, we assume that passphrases are known by the forger. Additionally, we assume that both corpora contain letters in the target user's passphrase. Given these corpora, our algorithm takes a set of n-grams $(g_1, \ldots, g_z)$ from a target user and replicates a passphrase. In particular, it shifts, transposes, and concatenates the t, $x(t)$, $y(t)$ and $p(t)$ signals of each n-gram to create a master set of signals that represents a forgery. This process can be classified into three high-level stages: (1) adjusting the spatial positions of each n-gram, (2) adjusting the ordering of strokes across each n-gram, and (3) adjusting the overall time signal.

Adjusting the spatial location of each point is relatively straightforward. We first shift each n-gram such that the baselines align on the same horizontal axis. We then use population statistics to determine the horizontal distance between each n-gram. In particular, given the last character of the first n-gram, and the first character of the second, we distance the two n-grams by the median distance between these two characters as they appear in the population (if these two letters generally overlap, then this value could be negative, which would cause a shift to the left). This process is applied iteratively over each n-gram to create the final horizontal alignment.

Explicitly, let $\omega(g_i, g_{i+1})$ be the median distance between the last character in g_i and the first in g_{i+1}, and X_i be the value of the x coordinate of the rightmost point in g_i (the leftmost value is always normalized to 0). Then our final forgery will incorporate g_i, but with all points shifted to the right by

$$\delta_x(i) = \delta_x(i-1) + X_i + \omega(g_{i-1}, g_i)$$

for $2 \leq i \leq z$ and $\delta_x(1) = 0$. After completing the vertical and horizontal shifts of the n-grams we have effectively created a static forgery. To modify this forgery to mimic temporal features we apply the second and third stages.

The second stage, determining the order of strokes across n-grams, is slightly more complicated. We start with the left-most letter of the left-most n-gram and proceed to the right, ordering the strokes according to their temporal order within each n-gram. There is one exception to this rule: we must delay strokes that occur after a stroke that is connected to the proceeding letter. For instance, cursive writers might render the dot of an ‘i’ that appears near the beginning of a word only after finishing all other letters. We exploit the simple observation that a stroke will (generally) only be delayed if a preceding stroke is connected to the following letter. The probability that a given stroke is connected to a given letter may be inferred from the population. So, we process each stroke of each letter, and connect it to the first stroke of the next letter with this inferred probability. If we decide to connect a stroke to the next letter, we push the proceedings strokes onto a stack and smooth the ends of the connected strokes using an iterative averaging algorithm. On the next pen-up event (either one that occurs naturally within an n-gram or the next time we decide to not connect letters) we empty the stack. This simulates a cursive writer who writes a phrase and then returns to dot ‘i’s and cross ‘t’s, starting with those closest to the end of the word.

The third stage, shifting time signals, involves determining the temporal delay between each n-gram, as well as the delay imposed on the deferred strokes. To determine the delay between the last character of one n-gram and the first character of the next we simply take the median delay between these two characters as they appear in the population. Mathematically, let $\tau(g_i, g_{i+1})$ be the median delay between the last character in g_i and the first in g_{i+1}. Let T_i be the time that the first delayed stroke in the last letter of g_i starts. If there are no delayed strokes in g_i, T_i is the maximum time (the time signal of g_i is always shifted to start at 0). Then our final forgery will incorporate g_i, but with the time shifted by

$$\delta_t(i) = \delta_t(i-1) + T_i + \tau(g_{i-1}, g_i)$$

for $2 \leq i \leq z$ and $\delta_t(1) = 0$.

To infer the elapsed time for delayed strokes we use the 75^{th} percentile pen-up velocity from the population and the

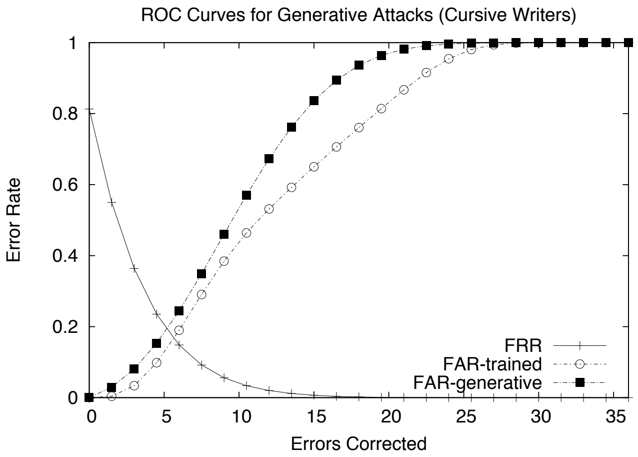

Fig. 10. ROC curves for generative forgeries against cursive writers. Even with access to only limited information, the algorithm outperforms our trained forgers, shifting the EER from 13.5% at four errors to 16.1% at three errors.

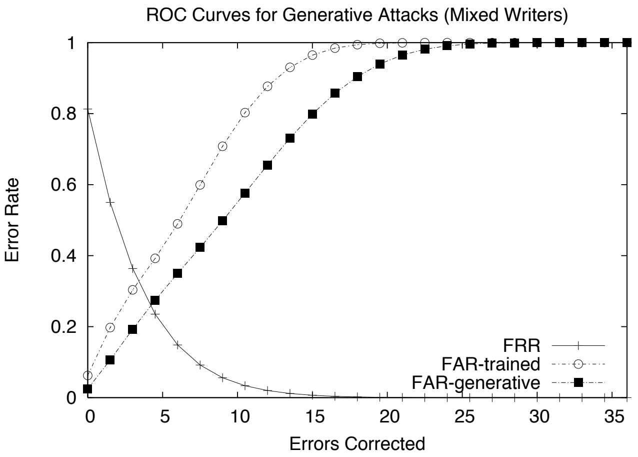

Fig. 11. ROC curves for generative forgeries against mixed writers. The generative algorithm does not perform as well as our trained forgers: the EER shifts from 33.4% at three errors to 24.5% at four errors.

distance between the beginning of a delayed stroke and the last rendered stroke. We choose to use the 75^{th} percentile as pen-up velocities tend to be dominated by velocities associated with spaces, which are intuitively slower than those associated with dotting ‘i’s and crossing ‘t’s.

Using these three high-level stages, we combine each of the n-grams to make a final set of signals to represent a forgery.

VIII. Generative Results

To evaluate this concatentative approach we analyzed the quality of the generated forgeries for each user and passphrase. However, rather than using all 65 of the available samples from the parallel corpus, we instead choose 15 samples at random from each target user’s parallel corpus—with the one restriction that there must exist at least one instance of each character in the passphrase among the 15 samples. The attacker’s choice of n-grams are selected from this restricted set. To explore the feasibility of our generative algorithm we ensure that adjacent n-grams do not originate from the same writing sample, but an actual adversary might benefit from using n-grams from the same writing sample.

Additionally, we limit the corpus from which we derive population statistics to contain only 15 randomly selected samples from each user with a similar writing style as the target user. We purposefully chose to use small (and arguably, easily obtainable) data sets to illustrate the power of this concatenative attack. Example forgeries derived by this process are shown in Figure 9.

We generated 25 forgery attempts for each user and passphrase. Figures 10 and 11 depict the average FAR across all 25 forgery attempts for cursive and mixed writers. As a baseline for comparison, we replot the FRR and FAR-trained plots from Section V. The FAR-generative plot shows the results of the generative algorithm against the entire population. Overall, under these forgeries there is an EER of 27.4% at three error correction compared to an EER of 20.6% at four error corrections when considering our trained forgers.

In general, the generative approach fares well against block writers (not shown), improving the EER over trained forgers from 20.4% at four errors corrected to 31.2% at three errors corrected. The improvement is less pronounced against cursive writers (Figure 10), only improving over the EER of trained forgers from 13.5% at six errors corrected to 16.1% at five errors corrected. Interestingly, the generative approach does not outperform our trained forgers against mixed writers. The EER for trained forgers of mixed writers was 33.4% at three errors, whereas our generative approach only achieved 24.5% at four errors (Figure 11). However, we argue that this approach highlights an adversarial threat that should be accounted for in analyzing the security of biometric systems.

Lastly, we note that on average each generative attempt only used information from 6.67 of the target user’s writing samples. Moreover, the average length of an n-gram was 1.64 characters (and was never greater than 4). More importantly, as we make no attempt to filter the output of the generative algorithm by rank-ordering the best forgeries, the results could be much improved. That said, we believe that given the limited information assumed here, the results of this generative attack only underscores its importance.

IX. Related Work

This paper expands upon previous work [1], where we discuss our data collection efforts, highlight the shortcomings associated with assuming weak adversaries, and provide our generative framework as an approach to facilitate more robust evaluation metrics. In the current work, we present a more thorough analysis of our testing methodologies and elaborate on the results of our trained forgers. Additionally, we present results from a study that show that the forgeries we collected were not only able to bypass an automated reference monitor, but were also able to mislead *human* judges.

To the best of our knowledge, there is relatively little work that encompass our goals and attack models described herein. However, there is a vast body of work on the topic of signature verification (see [10], [28]). Kholmatov and Yanikoglu [13] provide an authentication mechanism based on online signature verification. Their technique uses Dynamic Time Warping in conjunction with Principle Component Analysis and a linear classifier to decide whether or not signatures are genuine. This approach achieved an EER of approximately 1.3% and was used to win the first Signature Verification Competition (SVC) [35]. Additionally, Kholmatov and Yanikoglu employ a tool to provide realtime playback of writing samples to aid forgers during evaluation. However, it is unclear whether forger proficiency or training was considered when computing FAR.

Also germane are a series of recent papers that have started to examine the use of dynamic handwriting for the generation of cryptographic keys. Kuan et al. present a method based on block-cipher principles to yield cryptographic keys from signatures [14]. The authors test their algorithm on the SVC data set and report EERs of between 6% and 14% if the forger has access to a stolen token. The production of skilled forgeries in the SVC data set resembles part of the methodology used in round II of our studies and so does not account for motivation, training, or talent.

Finally, there have been a handful of works on using generative models to attack biometric authentication. However, we note there exists significant disagreement in the literature concerning the potential effectiveness of similar (but inherently simpler) attacks on speaker verification systems (e.g., [23], [27]). Lindberg and Blomberg, for example, determined that synthesized passphrases were not effective in their small-scale experiments [18], whereas Masuko et al. found that their system was defeated [22].

X. Conclusions

Some of the most fundamental computer security mechanisms—whether they are used to ensure access control, data privacy, or data integrity—rest on the ability of a legitimate user to generate an input that an attacker is unable to reproduce. The security of technologies that are based on behavioral biometrics is estimated using the perceived inability of forgers to replicate a target user's input. We caution that if legitimate adversaries are not considered, this practice may significantly underestimate the real risk of accepting forgeries. In fact, we demonstrate for a specific behavioral biometric, that even a small amount of training can drastically improve a forger's chances at success.

To address previous evaluation shortcomings and data collection obstacles, we present an automated technique for producing forgeries to assist in the evaluation of biometric systems. We show that our generative approach matches or exceeds the effectiveness of forgeries rendered by the trained humans in our study, and thus offers a viable alternative for enhancing the evaluation of biometric security. We argue that such an approach is imperative when weakest-link security assessment is important. As part of our future work, we intend to incorporate more sophisticated algorithms in our generative models (e.g., [17], [30]), and provide a broader spectrum of approaches for enhancing biometric performance evaluation.

We believe that the ideas and methodologies presented herein can be extended beyond handwriting to provide weakest-link type analyses of other behavioral biometric modalities. Specifically, we feel that the idea of trained and target-selected forgers should be examined further. For instance, one might take ethnicity and gender into account when creating forgeries for voice-based biometrics, or handedness and hand size into account for keystroke-based biometrics. Another worthwhile avenue of research could examine generative models to explore the threat space for other modalities.

Acknowledgments

The authors would like to thank Dishant Patel, Carolyn Buckley, Charles V. Wright and Jay Zarfoss for their help in our data collection efforts. We are also grateful to the anonymous reviewers from the 15^{th} Annual USENIX Security Symposium and the IEEE Transactions on Systems, Man, and Cybernetics for helpful comments. We especially thank the many people who devoted hours to providing us with handwriting samples. This work is supported by NSF grant CNS-0430338.

References

[1] L. Ballard, F. Monrose, and D. Lopresti, "Biometric authentication revisited: Understanding the impact of wolves in sheep's clothing," in *Proceedings of the 15^{th} Annual USENIX Security Symposium*, August 2006, pp. 29–41.

[2] G. R. Doddington, W. Liggett, A. F. Martin, M. Przybocki, and D. A. Reynolds, "Sheep, goats, lambs and wolves: A statistical analysis of speaker performance in the NIST 1998 speaker recognition evaluation," in *Proceedings of the Fifth International Conference on Spoken Language Processing*, November 1998.

[3] S. J. Elliott, "Development of a biometric testing protocol for dynamic signature verification," in *Proceedings of the International Conference on Automation, Robotics, and Computer Vision*, Singapore, 2002.

[4] M. C. Fairhurst, "Signature verification revisited: promoting practical exploitation of biometric technology," *Electronics & Communication Engineering Journal*, pp. 273–280, December 1997.

[5] D. Feldmeier and P. Karn, "UNIX password security – ten years later," in *Advances in Cryptology – CRYPTO '89 Proceedings*, ser. Lecture Notes in Computer Science, vol. 435. Berlin, Germany: Springer-Verlag, 1990, pp. 44–63.

[6] O. Galbraith, C. Galbraith, and N. Galbraith, "The principle of the 'drunkard's search' as a proxy for scientific analysis: The misuse of handwriting test data in a law journal article," *International Journal of Forensic Document Examiners*, vol. 1, pp. 7–17, 1995.

[7] R. M. Guest, "The repeatability of signatures," in *Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition*, October 2004, pp. 492–497.

[8] I. Guyon, "Handwriting synthesis from handwritten glyphs," in *Proceedings of the Fifth International Workshop on Frontiers of Handwriting Recognition*, Colchester, England, 1996.

[9] C. Hertel and H. Bunke, "A set of novel features for writer identification," in *Proceedings of the International Conference on Audio- and Video-based Biometric Person Authentication*, Guilford, UK, 2003, pp. 679–687.

[10] A. K. Jain, F. D. Griess, and S. D. Connell, "On-line signature verification," *Pattern Recognition*, vol. 35, no. 12, pp. 2963–2972, 2002.

[11] M. Kam, G. Fielding, and R. Conn., "Writer identification by professional document examiners," *Journal of Forensic Sciences*, no. 42, pp. 778–785, 1997.

[12] M. Kam, K. Gummadidala, and R. Conn, "Signature authentication by forensic document examiners," *Journal of Forensic Science*, vol. 46, 2001.

[13] A. Kholmatov and B. Yanikoglu, "Biometric authentication using online signatures," in *Computer and Information Sciences - ISCIS 2004, 19th International Symposium*, ser. LNCS. Springer, October 2004, pp. 373–380.

[14] Y. W. Kuan, A. Goh, D. Ngo, and A. Teoh, "Cryptographic keys from dynamic hand-signatures with biometric security preservation and replaceability," in *Proceedings of the Fourth IEEE Workshop on Automatic Identification Advanced Technologies*. Los Alamitos, CA: IEEE Computer Society, 2005, pp. 27–32.

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

[16] L. Lee, T. Berger, and E. Aviczer, "Reliable on-line human signature verification systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 6, pp. 643–647, June 1996.

[17] X. Li, M. Parizeau, and R. Plamondon, "Segmentation and reconstruction of on-line handwritten scripts," *Pattern Recognition*, vol. 31, no. 6, pp. 675–684, December 1998.

[18] J. Lindberg and M. Blomberg, "Vulnerability in speaker verification – a study of technical impostor techniques," in *Proceedings of the European Conference on Speech Communication and Technology*, vol. 3, Budapest, Hungary, September 1999, pp. 1211–1214.

[19] D. P. Lopresti and J. D. Raim, "The effectiveness of generative attacks on an online handwriting biometric," in *Proceedings of the International Conference on Audio- and Video-based Biometric Person Authentication*, Hilton Rye Town, NY, USA, 2005, pp. 1090–1099.

[20] A. J. Mansfield and J. L. Wayman, "Best practices in testing and reporting performance of biometric devices," Centre for Mathematics and Scientific Computing, National Physical Laboratory, Tech. Rep. NPL Report CMSC 14/02, August 2002.

[21] U.-V. Marti, R. Messerli, and H. Bunke, "Writer identification using text line based features," in *Proceedings of the Sixth International Conference on Document Analysis and Recognition*, September 2001, pp. 101–105.

[22] T. Masuko, K. Tokuda, and T. Kobayashi, "Imposture using synthetic speech against speaker verification based on spectrum and pitch," in *Proceedings of the International Conference on Spoken Language Processing*, vol. 3, Beijing, China, October 2000, pp. 302–305.

[23] F. Monrose, M. Reiter, Q. Li, D. Lopresti, and C. Shih, "Towards speech-generated cryptographic keys on resource-constrained devices," in *Proceedings of the Eleventh USENIX Security Symposium*, 2002, pp. 283–296.

[24] F. Monrose, M. K. Reiter, Q. Li, and S. Wetzel, "Cryptographic key generation from voice (extended abstract)," in *Proceeedings of the 2001 IEEE Symposium on Security and Privacy*, May 2001, pp. 12–25.

[25] I. Nakanishi, H. Sakamoto, Y. Itoh, and Y. Fukui, "Optimal user weighting fusion in DWT domain on-line signature verification," in *Proceedings of the International Conference on Audio- and Video-based Biometric Person Authentication*, Hilton Rye Town, NY, USA, 2005, pp. 758–766.

[26] W. Nelson and E. Kishon, "Use of dynamic features for signature verification," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, October 1991, pp. 1504–1510.

[27] B. L. Pellom and J. H. L. Hansen, "An experimental study of speaker verification sensitivity to computer voice altered imposters," in *Proceedings of the 1999 International Conference on Acoustics, Speech, and Signal Processing*, March 1999.

[28] R. Plamondon, Ed., *Progress in Automatic Signature Verification*. World Scientific, 1994.

[29] R. Plamondon, D. P. Lopresti, L. R. B. Schomaker, and R. Srihari, "Online handwriting recognition," in *Wiley Encyclopedia of Electrical and Electronics Engineering*. John Wiley & Sons, Inc., 1999, pp. 123–146.

[30] R. Plamondon and S. N. Srihari, "On-line and off-line handwriting recognition: a comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63–84, 2000.

[31] D. Risinger, M. Denbeaux, and M. Saks., "Exorcism of ignorance as a proxy for rational knowledge: the lessons of handwriting identification 'expertise'," *University of Pennsylvania Law Review*, vol. 137, pp. 731–787, 1989.

[32] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and B. V. Kumar, "Biometric encryption^{tm} using image processing," in *Optical Security and Counterfeit Deterrence Techniques II*, vol. 3314. IS&T/SPIE, 1998, pp. 178–188.

[33] C. Vielhauer and F. Zöbisch, "A test tool to support brute-force online and offline signature forgery tests on mobile devices," in *Proceedings of the International Conference on Multimedia and Expo*, vol. 3, 2003, pp. 225–228.

[34] C. Vielhauer and R. Steinmetz, "Handwriting: Feature correlation analysis for biometric hashes," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 542–558, 2004.

[35] D.-Y. Yeung, H. Chang, Y. Xiong, S. George, R. Kashi, T. Matsumoto, and G. Rigoll, "SVC2004: First international signature verification competition," in *Proceedings of the International Conference on Biometric Authentication (ICBA)*, Hong Kong, July 2004.

Lucas Ballard received his B.S. in computer science and mathematics from Brandeis University in 2003. He is currently pursuing his Ph.D. in computer science at Johns Hopkins University, where he received his M.S. in security informatics in 2004 and his M.S.E. in computer science in 2006. His research focuses on computer security and applied cryptography, particularly in the contexts of biometrics and networking.

Daniel Lopresti received his bachelor's degree from Dartmouth College in 1982 and his Ph.D. in computer science from Princeton University in 1987. He spent several years in the Computer Science Department at Brown University, and then went on to help found the Matsushita Information Technology Laboratory in Princeton, NJ. He later also served on the research staff at Bell Labs. In 2003, Dr. Lopresti joined the Computer Science and Engineering Department at Lehigh University where he leads research examining fundamental algorithmic and systems-related questions in pattern recognition, bioinformatics, and security. He has authored or co-authored over 80 publications in journals and refereed conference proceedings, and holds 21 U.S. patents. He is a Senior Member of IEEE.

Fabian Monrose received his Ph.D. in Computer Science from the Courant Institute of Mathematical Sciences, New York University, in 1999. For three years thereafter, he served as a member of technical staff in the Secure Systems Research group at Bell Labs, Lucent Technologies. In 2002, he joined the Computer Science Department at Johns Hopkins University, and holds a joint appointment in the Hopkins Information Security Institute. He holds 3 U.S. patents, has co-authored more than 30 refereed papers in computer and information security, and has served on over 20 program committees for security venues. He currently serves on the editorial board of ACM Transactions of Information and System Security.