SigningRing: Signature-based Authentication using Inertial Sensors on a Ring Form-factor

Yifeng Cao
ycao361@gatech.edu
Georgia Institute of Technology
USA

Ashutosh Dhekne
dhekne@gatech.edu
Georgia Institute of Technology
USA

Mostafa Ammar
ammar@cc.gatech.edu
Georgia Institute of Technology
USA

ABSTRACT

Commercial smart rings have demonstrated their utility in health applications such as sleep monitoring and fitness tracking. However, given the small size of a ring form-factor, the applicable scenarios of smart rings are still under-explored. This paper presents SigningRing, which proposes a novel functionality for the smart ring: employing the inertial sensor embedded in the smart ring for secure and fast user authentication. To access an account, the user wearing the ring will move their finger in air, tracing a signature-like pattern, as if signing their name. In our evaluation with 18 volunteers, SigningRing achieves 97.4% in balanced accuracy, with 99.8% true negative rate and 95.1% true positive rate.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Hardware** → *Sensor applications and deployments.*

KEYWORDS

IMU, Sign, Ring, Gestures, Authentication

ACM Reference Format:
Yifeng Cao, Ashutosh Dhekne, and Mostafa Ammar. 2024. SigningRing: Signature-based Authentication using Inertial Sensors on a Ring Form-factor. In *Workshop on Body-Centric Computing Systems (BodySys '24), June 3–7, 2024, Minato-ku, Tokyo, Japan.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3662009.3662019

1 INTRODUCTION

The ring form-factor is gaining popularity for health and fitness applications with both commercial products such as the Oura Ring [2] and academic research [21]. In this paper we explore a completely different application for the ring form-factor: authentication. We wonder: *Is it possible to use specific movements of the hand or a finger as recorded by a ring on the finger in lieu of passwords for authentication?* The specific hand movements could be predefined patterns or personal signatures and could provide either a password replacement solution, or a full two-factor-authentication (2FA) solution with the ring hardware representing a token possessed by the user. Our recently accepted work at WiSec, called UWB-Auth [5], describes the 2FA mechanisms in more detail, whereas this paper focuses on our explorations around the ring form factor for signing-based authentication.

In this paper, we develop a system, called SigningRing, that captures a user's hand or finger movements using inertial sensors installed on a ring worn by the user. These movements are compared with previously registered signature movements and if they match, authentication is considered to be successful. Of course, such a signature verification system must rely on inexact matching, and determining an acceptable error-margin forms an important challenge for our system. However, since inertial sensors capture both the position information (what is being written/drawn) and the timing of movements (how it is being written/drawn), SigningRing is expected to be quite robust. When the user wearing the ring performs authentication, they will virtually draw a pattern in the air. The motion features, including speed, pauses, intricate strokes, are captured by the inertial measurements (acceleration, angular velocities). Serving as raw data, these inertial measurements are fed into a Siamese Neural Network (SNN) for similarity calculation with pre-registered features. The authentication passes only when the submitted features matched the registered features.

In our evaluation we recruit 18 volunteers and ask them to draw signatures using our ring platform. It can achieve a low false positive rate (adversary gaining access to the online account, *after* obtaining the token) of 0.2% while maintaining false negative rate (a legitimate user is unable to access at the first attempt) as low as 4.9%. We hope our explorations will encourage further work in this direction.

BodySys '24, June 3–7, 2024, Minato-ku, Tokyo, Japan
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0666-0/24/06
https://doi.org/10.1145/3662009.3662019

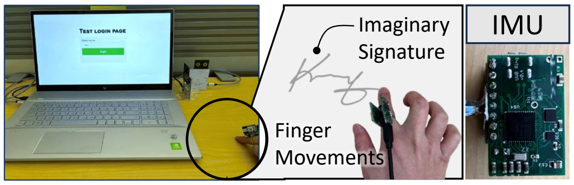

Figure 1: SigningRing prototype: IMU for authentication via finger movements drawing an imaginary signature.

2 RELATED WORK

User identification. In research, use of diverse modalities for user authentication on mobile devices, such as acoustic signal [6, 17], vibration signal [19], IMU [7, 14], RF signal [18], etc., have been explored. SigningRing leverages the uniqueness of each individual's signature for user identification. Traditional signature authentication is mainly computer vision based [12] or touchpad based [15], which have demonstrated the effectiveness of using signature in identification. SigningRing instead extracts motion features of handwritten signatures, using an IMU, for fast and accurate signature authentication.

Authentication with in-air drawing. User signatures have been widely used for authentication from financial transactions to employment contracts. Recently, the research community has started to explore the possibility of using virtual in-air drawing in the authentication system [3, 8]. However, these solutions rely on leap motion sensor [1] to reconstruct the drawing trajectory with infrared, which leads to high hardware cost and low usability. Also, merely matching trajectories disregards the biometric features that are correlated with the user writing habit, such as pause and the writing speed. In contrast, SigningRing exploits motion features collected by inertial sensors to match the user's virtual signature in multi-dimension, resulting in cheap and accurate authentication resilient to mimicking attack.

3 DESIGN

3.1 Promise in Personal Signatures

Signatures are usually a stylized way of writing a name, perfected by individuals through a significant amount of practice. It is replete with intricate strokes and pauses, and potentially curves, lines, and dots. In many cases, well designed signatures, while plainly visible on documents, are difficult to replicate by an attacker [10]. Inherently, the *knowledge of how a signature is drawn* is important in addition to knowing the word the signature represents.

We see an opportunity to incorporate the idea of signatures into the authentication know-factor. The wearable ring can capture the user's finger movements using an IMU and match those to the user's previously registered IMU readings. The movements will capture both *what was drawn* by the user as well as *how it was drawn*, thereby increasing the feature space for matching signatures. Fig. 2 shows how the same word, "MyPassword", written by two volunteers, differs visually, and as observed from the IMU sensor data.

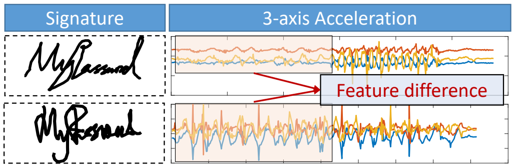

Figure 2: Two volunteers stylize the word "MyPassword". (Left) visualized movements; (Right) acceleration readings.

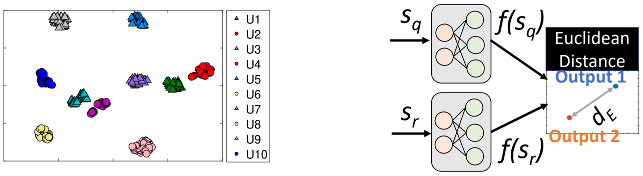

Figure 3: (Left) Feature visualization using t-SNE [16] of signatures written by 10 different users; (b) (Right) The forward process of siamese network to compute signature similarity.

Of course, in contrast to traditional authentication approaches like passwords, matching signatures is an inexact process, meaning, there will be instances when a legitimate user's signature may fail to match with their registered signature. Similarly, there might be instances when an attacker may be able to draw something that looks similar to the user's registered signature from the lens of the IMU readings. As a guiding principle, we wish to design a signature matching algorithm, based on IMU data, that will cluster together the signatures drawn by an individual, but separate them well when compared to signatures of others. Our algorithm does not use visible written signatures, but rather finger movements (captured by the ring) that mimic signature on a flat surface.

Next, we discuss the details of extracting features from the IMU sensor data, and signature comparison algorithms.

3.2 Feature Extraction from IMU Data

At a high level, we wish to create a model that matches a user-supplied finger movement data, captured using an IMU on the ring during usual login, with similar movement data registered by the user during signup. However, sensor data can be quite noisy, meaning comparing two sequences of sensor data is non-trivial. Therefore, we extract and store features of registered sensor data (when the user first signs up) and, at every login, match these features with the sensor data obtained from the user freshly drawing their signature.

We use hand-crafted features that closely resemble those qualitatively described in [11], modified for suitability on a ring worn on the finger:

- The time span of the handwriting sequence.

- The mean and the standard deviation of acceleration of every 0.5s time window in 8 seconds (represents the positional derivatives described in [11]).
- The zero-crossing rate of acceleration of every 0.5s time window in 8 seconds.
- The Fast Fourier Transform (FFT) of 3-axis acceleration.

These features collectively function as the fingerprint of one's signature. Fig. 2 (right) shows an example of the raw IMU data from which the above features are extracted. Since individual personalized signatures are also quite different from one another (and not the same word written differently), we expect a larger separation between signatures from different people. Fig. 3 (left) visualizes the separation between personalized signatures of 10 different users converted to 2D space using t-SNE [16], a method to visualize high-dimension data in a low-dimensional space. Signatures from different users are clearly separable, while different instances of the user's signatures are closely clustered, an essential property for effective signature matching, which we discuss next.

3.3 Signature Similarity Based Matching

The extracted features are then fed into a matching model to determine if a query signature sequence (during a login attempt) matches the signatures registered during signup. To perform this comparison of signature sequences, we adopt siamese neural network (SNN) [4], an architecture using the same parameters on two input vectors and producing a comparable output vector. The forward process of the SNN is depicted in Fig. 3 (right): The query signature s_q and a registered signature s_r are fed into the same neural network, producing $f(s_q)$ and $f(s_r)$ respectively, in the output space. If s_q is from the same user, outputs of the SNN, $f(s_q)$ and $f(s_r)$, should be close in the feature space. Otherwise, the outputs should be far from each other. We use Euclidean distance $d = d_E(f(s_q), f(s_r))$ to measure the proximity of two output vectors. We use a simple two-layer perceptron (128×16) network that avoids overfitting. The network is trained by back-propagating the contrastive loss [9]:

$$\mathcal{L}(s_q, s_r) = \ yd \ + (1-y)\max(0, m-d), \qquad (1)$$

where $y \in \{0, 1\}$ represents the true label and m is the margin for contrastive loss. In our design, we use $m = 0.5$.

Of course, signatures drawn by the same individual naturally have small variations. Having access to only a single registered signature can be too restrictive for usability, making it very difficult for the legitimate user to authenticate their own accounts, or if the matching criteria are kept loose, may make unauthorized access easier. To enhance SigningRing's robustness, the user will be asked to repeat the same signatures multiple times (n) in the registration phase (s_r), which forms a database of legitimate signatures $\mathcal{S} = \{s_r^{(i)}, i \in [1, n]\}$. To pass signature verification, the query signature s_q will be compared with each registered signature in the database. When s_q is "close" ($d_E(f(s_q), f(s_r))$ is small) to a majority of the registered signatures, SigningRing assumes s_q comes from a legitimate user. Mathematically, SigningRing checks:

$$|\{s_r^{(i)} : d_E(f(s_r^{(i)}), f(s_q)) < d_0)\}| \geq \rho|S|, \qquad (2)$$

where d_0 is the Euclidean distance threshold of SNN, and ρ is the "majority" percentage over which a query signature is treated as being legitimate.

4 IMPLEMENTATION

We have developed a ring form-factor hardware with two PCB layers one for compute and inertial sensors and another for communication. The overall dimension is $3cm \times 1.8cm \times 1.5cm$. The top layer consists of a DWM1000 chip and a UWB antenna which only performs communication between the login device and the ring. The bottom layer houses a LSM6DSO 6-DoF inertial sensor, along with a Cortex M0. The UWB chip and IMU are connected to a Cortex M0 microcontroller for data processing. The IMU collects data at 70Hz rate.

5 EVALUATION

5.1 Signature Authentication

We evaluate the performance of signature authentication phase (know-factor), by recruiting 18 volunteers[1] and collects 78 instances of their signatures, for a total of 1,559 signatures, covering various signature styles and levels of practice. All the volunteers are right-handed, with the ring worn on their index finger. Signature authentication in the SigningRing protocol provides a secure and usable way to ensure legitimacy of the user, *replacing passwords.*

5.1.1 Understanding the Space of Signatures. The matching of IMU sensor data is inexact. Due to inherent variations in signatures drawn by a person we must allow a *margin of error*. However, this margin of error could also allow an attacker's signature to be accepted by SigningRing. Therefore, we must evaluate SigningRing under the following metrics, calculated from true positives (TP), false negatives (FN), true negatives (TN), false positives (FP), and understand the tradeoffs between them:

(1) *True positive rate (TPR)*: $\frac{TP}{TP+FN}$. TPR describes the probability that a legitimate signature is accepted by the server, which results in a higher *speed of authentication* and a better *user experience* of the 2FA scheme.

(2) *True negative rate (TNR)*: $\frac{TN}{TN+FP}$. TNR measures the probability that a fake signature is rejected by the server (*security* of the signature verification algorithm).

(3) *Overall accuracy*: $\frac{TP+TN}{TP+TN+FP+FN}$. This balance-metric measures the overall performance of signature verification.

[1] This study has been approved by the IRB at our institution.

Table 1: Accuracy, TPR and TNR of different models.

Model	Accuracy (%)	TPR (%)	TNR (%)
CTW [20]	92.4	90.6	94.1
OC-SVM [13]	92.6	89.5	95.7
SVM	94.6	97.9	91.3
Siamese (SNN)	**97.4**	**95.1**	**99.8**

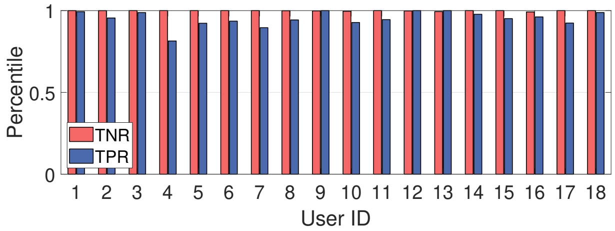

Figure 4: TNR and TPR of each volunteer.

As mentioned in Section 3, SigningRing extracts features from raw IMU data and employs SNN for signature authentication. A query signature is assumed legitimate if it is close to a majority of the registered signatures. In this experiment, we set $\rho = 0.8$. The SNN distance threshold d_0 is calculated by $d_0 = \alpha \cdot d_0^{tr}$, where d_0^{tr} is the distance maximizing overall accuracy in the training dataset, and α is a scale factor representing the preference on TNR or TPR. The model perfers higher TNR when $\alpha < 1$. Otherwise, the model perfers higher TPR. We set α to 0.75, since rejecting fake signatures is typically more important. The evaluation is performed via $k-$fold cross-validation (k is the number of volunteers): we iteratively train the model with $k - 1$ users' data, and evaluate on the remaining users.

The primary question we ask is: *what model of comparing the sensor data will keep an appropriate tight bound to maximize TPR and TNR?* We compare the following models with SNN (which we described in Section 3.3):
(1) *Canonical time warping (CTW)*: CTW [20] is an extension of dynamic time warping which calculates the similarity between two time-series sequences by performing spatial-temporal alignment. It rejects a signature if the similarity is less than a threshold, otherwise it accepts the signature.
(2) *Support vector machine (SVM)*: SVM takes the difference of the query signature and one pre-registered signature, and performs binary classification to determine signature match.
(3) *One-class SVM (OC-SVM)* [13]: OC-SVM trains only with positive signatures and looks for a hyperplane that maximizes the distance to the origin in the feature space. The signature that lies between the origin and hyperplane will be rejected, otherwise it will be accepted.

Overall performance. Table. 1 presents the results in term of overall accuracy, TPR, and TNR, of the same k-fold cross-validation set over all the above models. Overall, SNN achieves the best performance across all the models, with 95.1% accuracy in approving a legitimate query and 99.8% accuracy in rejecting a fake query. This demonstrates that our feature extraction and SNN-based similarity comparison effectively encode raw IMU data into the feature space that signatures are comparable with the Euclidean distance.

Of course, as we set higher preference on rejecting fake queries, TNR is slightly higher than TPR. Based on the application use-case and user preferences, the weight of TNR and TPR is adjustable by tuning the scale factor α. For instance, a user who is confident that the ring cannot be lost or stolen, may prefer higher TPR. Fig. 5 shows the trade-off between TPR and TNR when different values of $alpha$ are used. As α increases from 0.7 to 1.1, TNR/TPR varies from 99.9%/91.9% to 95.2%/99.9%. Observe that TPR and TNR are balanced when $\alpha = 0.91$ rather than 1, which results from differences between the training and test datasets.

Per-individual accuracy. An important question to ask, despite the overall TNR/TPR accuracy of 99.8%/95.1%, is: *do some specific users fail to login into their own accounts more than others?* Fig. 4 shows the TNR and TPR of each individual. We indeed observe that some users perform poorly compared to most others. For example, TPR of User 4 drops to 80% to maintain higher than 99% TNR. We summarize two underlying factors accounting for the low TPR of these individuals: (i) The individual fails to draw consistent signatures, resulting in a large margin of error to account for the larger intra-class variance; (ii) The individual uses a signature which is trivial to mimic. This indicates the need for a metric quantifying the *quality* of a signature.

To judge the improvement in TPR when low-quality signatures are not accepted, we plot the receiver operating characteristic curve (ROC) when users 4 and 7 are included or excluded (see Fig. 6). After removing user 4 and 7's data, the false positive rate (FPR), defined as the probability that a fake signature gets identified as a legitimate one, decreases for the same TPR. The process of determining the quality of a signature is analogous to password strength meters on websites: if a signature is too weak, the system can require redrawing a better, stronger signature for improved security. Of course, quantifying the strength of a signature is difficult. We explore the effect of signature duration on its strength in microbenchmarks, based on our collected dataset.

5.1.2 Resilience to Knowledge-Based Attacks. While the previous section shows a reasonable resilience (99.8%) to random signatures, we would also like to explore how SigningRing stands up against knowledge based attacks. Note that a personalized signature entails knowledge of *what* is drawn as well as *how* it is drawn. Therefore, we must evaluate both these aspects separately as well as jointly as follows:
(1) *Known-text (KT) attack*: In this attack, we assume that the adversary knows *what* is drawn but the adversary *does not observe how* the user draws the signature.

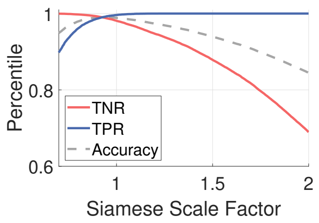

Figure 5: The tradeoff between TPR and TNR.

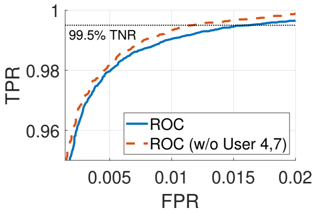

Figure 6: The ROC curve of signature authentication.

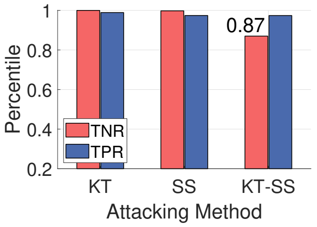

Figure 7: Performance of the model under attacks.

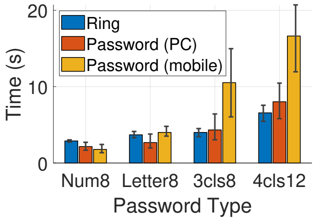

Figure 8: SigningRing input time versus password typing.

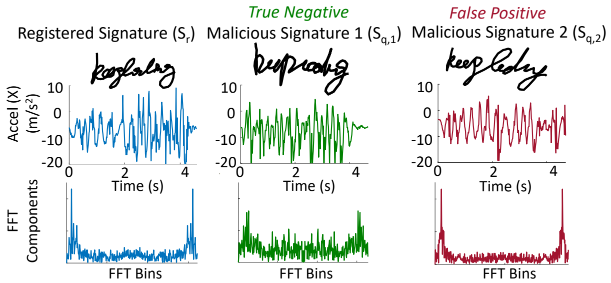

Figure 9: An example of a rejected KT-SS attack (true negative) and an accepted KT-SS attack (false positive).

(2) *Shoulder surfing (SS) attack*: In this common attack scenario, the adversary observes *how the user draws* their signature, but *does not know what* (the characters) the user draws. (3) *Known-text and shoulder-surfing (KT-SS) attack*: The adversary knows *what* is drawn (the characters that make up the signature are known to the adversary) and observes *how the user writes*. The adversary may video-record and watch the user’s signing style and do skilled practice to mimic the drawing style of the user.

To evaluate the performance of each kind of attack, one volunteer is asked to design their signature and register it. Other volunteers, acting as adversaries, will try to trick the model into accepting the signatures drawn by them. In KT-SS attacks, the adversaries are allowed to watch the video which captures the entire signing process and practice unlimited number of times. Note that knowledge is additive in this evaluation, therefore, different sets of volunteers and signatures are involved in each of the KT, SS, and KT-SS attack evaluations. Fig. 7 shows the TNR under each attack (each instance has its own TPR since a different volunteer set is involved). Fig. 7 shows SigningRing’s resilience to KT attack and SS attacks, achieving nearly 100% performance in rejecting attackers’ signatures. Such high TNR results from the biometric nature of signatures: even if an adversary knows the signature content, or has seen the hand movements, it is difficult to mimic the signature’s subtleties, such as pause, speed, strength, etc. In contrast, password-based authentication would break under either of these attacks.

Of course, SigningRing does not bring absolute security: in the KT-SS attacks, the TNR decreases to 87%. Fig. 9 gives an example of a failed and a successful KT-SS attack, in which the signature word is “keepcoding”. We can see while malicious query signature 1 ($S_{q,1}$) attempts to mimic the registered signature (e.g., letter connection and pause), it is still distinguishable with IMU features. In comparison, malicious signature 2 resembles the subtleties of the legitimate user, resulting in similar features which tricks the model. However, we would like to point out that $s_{q,2}$ is written by a volunteer who practiced mimicking the victim’s signature for tens of minutes with a pre-recorded slow-motion video. A potential solution might be to model with adversarial samples, which we leave to future work.

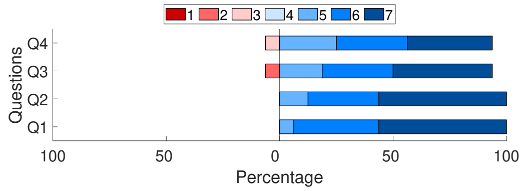

Figure 10: Likert scale chart of SigningRing user study.

5.1.3 *Signature Input Time.*

In addition to authentication accuracy, the time taken to input the signature is also a crucial consideration in real-world 2FA solutions. Long and complicated authentication process degrades user experience. Here we take an analytical view and compare SigningRing’s signature input time with password typing (1) on a physical keyboard on PC; and (2) on a virtual keyboard of mobile devices (we use iPhone13 in the experiment). We generate random passwords of four different levels of complexity for test: (1) Pure number with a length ≥ 8 (Num8); (2) Pure letters with a length ≥ 8 (Letter8); (3) Combination of three of {number, lower-case letters, upper-case letters, symbols} with a length ≥ 8; (4) Combination of all of {number, lower-case letters, upper-case letters, symbols} a length ≥ 8. For SigningRing, volunteers write the signature of the same complexity. For all input methods, volunteers first practice for several times before being ready for timed repetitions.

Fig. 8 shows the input time for SigningRing, typing password on keyboard, and typing password on virtual keyboard. SigningRing takes 2.89*s*, 3.69s, 4*s*, 6.57*s* for the four password types. For simple passwords, the input time of SigningRing is comparable to traditional password typing in PC and mobile devices. However, SigningRing outperforms others in complicated passwords, especially when compared to typing in mobile devices. Specifically, compared to typing on PC and mobile devices, SigningRing is 1.08× and 2.63× faster (4.33s on PC and 10.52s on mobile devices) in 3Class8 password, and 1.22× and 2.53× faster (8.02s on PC and 16.62s on mobile devices) in 4Class12 password. Given how password-strength requirements are strictly enforced today, we believe SigningRing significantly improves usability.

5.2 Usability: User Study

To understand the experience of the volunteers, we asked the volunteers to fill out a short questionnaire at the end of the study. Specifically, we ask the following questions: (Q1) I can repeat drawing my pattern naturally and quite precisely. (Q2) I can quickly draw my pattern for authentication. (Q3) Overall, I think wearing the ring will not interfere with my daily life activities (particularly if the ring form-factor is converted into something like the oura ring [2]). (Q4) Overall, I think I can draw the same pattern over and over again with low cognitive load. We use a 7-point likert-scale describing the agreement levels from strongly disagree (1) to strongly agree(7) in our questionnaire. Likert scale chart of the answers from our volunteers (shown in Fig. 10) shows that volunteers overall agree that 2FA of SigningRing has high repeatability (Q1), short authentication time (Q2), low overhead of wearing the ring (Q3), and low cognitive load (Q4). While a larger user-base is required to provide conclusive evidence, we see significant promise in the SigningRing idea.

6 CONCLUDING REMARKS

We envision a few potential extensions of SigningRing. First, we believe quantifying signature quality will significantly improve the authentication security. Overly simplistic signatures make accounts vulnerable if the ring is stolen. Further, We have not explored robotic replication of signature drawing which could potentially mimic a user's finger movements almost exactly. In summary, SigningRing provides an option for authentication using wearable rings. It shuns passwords in favor of personalized signatures captured via inertial sensors, enabling fast and secure authentication.

REFERENCES

[1] Leap motion controller 2. https://www.ultraleap.com/.

[2] Oura. https://www.ouraring.com.

[3] Santosh Kumar Behera, Ajaya Kumar Dash, Debi Prosad Dogra, and Partha Pratim Roy. Air signature recognition using deep convolutional neural network-based sequential model. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3525–3530. IEEE, 2018.

[4] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, 6, 1993.

[5] Yifeng Cao, Ashutosh Dhekne, and Mostafa Ammar. Uwb-auth: A uwb-based two factor authentication platform. https://faculty.cc.gatech.edu/~dhekne/UWB_Auth_WiSec2024.pdf.

[6] Yongliang Chen, Tao Ni, Weitao Xu, and Tao Gu. Swipepass: Acoustic-based second-factor user authentication for smartphones. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(3):1–25, 2022.

[7] Andrea Ferlini, Dong Ma, Robert Harle, and Cecilia Mascolo. Eargate: gait-based user identification with in-ear microphones. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, pages 337–349, 2021.

[8] Elyoenai Guerra-Segura, Aysse Ortega-Pérez, and Carlos M Travieso. In-air signature verification system using leap motion. *Expert Systems with Applications*, 165:113797, 2021.

[9] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[10] Anil Jain, Lin Hong, and Sharath Pankanti. Biometric identification. *Communications of the ACM*, 43(2):90–98, 2000.

[11] Jacques Linden, Raymond Marquis, Silvia Bozza, and Franco Taroni. Dynamic signatures: A review of dynamic feature variation and forensic methodology. *Forensic science international*, 291:216–229, 2018.

[12] Muhammad Imran Malik, Sheraz Ahmed, Marcus Liwicki, and Andreas Dengel. Freak for real time forensic signature verification. In *2013 12th International Conference on Document Analysis and Recognition*, pages 971–975. IEEE, 2013.

[13] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[14] Dai Shi, Dan Tao, Jiangtao Wang, Muyan Yao, Zhibo Wang, Houjin Chen, and Sumi Helal. Fine-grained and context-aware behavioral biometrics for pattern lock on smartphones. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(1):1–30, 2021.

[15] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, and Javier Ortega-Garcia. Deepsign: Deep on-line signature verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(2):229–239, 2021.

[16] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[17] Cong Wu, Jing Chen, Kun He, Ziming Zhao, Ruiying Du, and Chen Zhang. Echohand: High accuracy and presentation attack resistant hand authentication on commodity mobile devices. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2931–2945, 2022.

[18] Weiye Xu, Wenfan Song, Jianwei Liu, Yajie Liu, Xin Cui, Yuanqing Zheng, Jinsong Han, Xinhuai Wang, and Kui Ren. Mask does not matter: anti-spoofing face authentication using mmwave without on-site registration. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, pages 310–323, 2022.

[19] Xiangyu Xu, Jiadi Yu, Yingying Chen, Qin Hua, Yanmin Zhu, Yi-Chao Chen, and Minglu Li. Touchpass: towards behavior-irrelevant on-touch user authentication on smartphones leveraging vibrations. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pages 1–13, 2020.

[20] Feng Zhou and Fernando Torre. Canonical time warping for alignment of human behavior. *Advances in neural information processing systems*, 22, 2009.

[21] Hao Zhou, Taiting Lu, Yilin Liu, Shijia Zhang, Runze Liu, and Mahanth Gowda. One ring to rule them all: An open source smartring platform for finger motion analytics and healthcare applications. In *Proceedings of the 8th ACM/IEEE Conference on Internet of Things Design and Implementation*, pages 27–38, 2023.